Digital Media Software Engineering (DMSE) Program Review

2012-2013

Table of Contents

Contents

A. PROGRAM GOALS

Program Overview

The Bachelor of Science in Digital Media Software Engineering (DMSE) program at Ferris State University Grand Rapids offers students a well-rounded education as they earn their Bachelors of Science, while giving them the depth of technical skills needed to compete in the fast-paced world of software engineering. Graduates of the DMSE program are able to enter into a growing field of software engineering with a high level of competence and confidence.

The DMSE program is designed as a traditional engineering degree focused on the technologies, procedures, and methodologies involved in the software development process. Graduates will have experience successfully designing, developing, and delivering quality software applications, from concept to completion.

Graduates will have a mastery of the entire software engineering process. The DMSE program is committed to innovation and relevance, while covering traditional Software Engineering and Emerging technologies. Graduates complete projects that require employing various Software Engineering development Methodologies and industry best practices. Many projects are selected by the student, and they have the opportunity to develop software applications for the PC desktop, Internet, mobile devices, Smart Phones, PC Tablets, robotics, Wi-Fi networks, game consoles, and next-generation gesture control devices. Each project is designed to prepare them to seamlessly integrate into the workforce and begin a productive a career.

Mission

To provide a quality education to students seeking to gain technical expertise in software engineering, and develop skills that are valued and relevant to the workplace.

Core Values

- Quality instruction by teachers who have a proven track record of industry success.
- First rate response and customer service.
- Leading hardware and software for lab use and instruction.
- Relevant industry involvement in classroom projects.
- Progressive strategy for implementing evolving curriculum and classroom content.
- Expanding base of opportunities for students to showcase abilities to and interact with industry.
- Growth and diversification of student population and industry opportunities in Digital Media Software Engineering for South Western Michigan.
- Outreach programs to high school and junior high schools designed to stimulate interest and awareness for Digital Media Software Engineering.

Vision

Recognition as one of the leading educational institutions in Digital Media Software Engineering in the nation by industry and students. Students will develop outstanding software and leadership skills, while gaining the confidence necessary to apply their formal training to improve the quality of software locally, nationally, and globally.

Customer Value Proposition

Students are the primary customers and will seek out FSU/GR DMSE because they will recognize the caliber of education they will receive without having to leave Michigan. As a state institution, it's reasonable to expect (at least initially) that the majority of the students will come from in state. Students will come to this program at first because it is the only program of its kind in Michigan. Additionally, students will see the emerging list of employers of DMSE graduates grow and increase their confidence in the validity and capability that a DMSE degree will bring them. In summary:

- The program is unique in the delivery of traditional and emerging technology course material.
- The program offers a quality Digital Media Software Engineering education
- The program has a 2+2 relationship with Grand Rapids Community College
- The program successfully prepares students for a strong and growing industry
- The program incorporates traditional Software Engineering and Emerging Technologies

Organizational Environment

The DMSE Department interfaces with a growing list of individuals, groups and organizations. It is through this networking that DMSE will ensure the success of the program and its students. The primary workforce includes a Program Director Provided that DMSE continues to prove successful, additional faculty will likely be added. DMSE also works in conjunction with a supervising administrative department (Dean, Assistant Dean) that includes a marketing staff, and an administrative assistant support staff and an Information Technology (IT) staff. DMSE resides within FSU/GR's office space as leased from Grand Rapids Community College, maintains faculty offices and equipment, and utilizes computer labs/classrooms equipped with leading-edge computers, software, and digital projection. DMSE has access to additional non-computer classrooms as needed and often spills over into two more computer labs.

Organizational Relationships

DMSE has a growing list of individuals and organizations with which it interacts. On the front end, DMSE works with high schools and junior high schools to promote the program and industry. Students entering the program provide the foundation for the program's existence and are the primary focus. DMSE needs to deliver results based on expectations set by the Administrative Department. DMSE also continues to develop industry relationships for professional portfolio reviews, developing and screening/judging class projects and program review offering internships, and hopefully employing graduates.

Competitive Environment & Strategic Challenges

Currently, DMSE is still the one of only 3 programs that has an entire curriculum focused solely on the development of quality software through the discipline of Software Engineering in Michigan. This type of program has begun to gain popularity nationally and globally. With many schools recognizing the potential enrollment benefits that a program like this brings, more will want to start their own programs. DMSE will need to stay ahead of the competition, constantly cultivating and recruiting new students, while continuing to grow opportunities for its graduates. DMSE works with community colleges to setting up 2+2 programs encouraging articulation of transfer students.

Explanation of How and By Whom the Goals Were Established

The original goals for DMSE were established by founders, Don Green, Tracy Powers and Frederick Baker. Goals were determined from real time experiences of Frederick Baker and curricula from established programs such as from Rochester Institute of Technology, Grand Valley State University (Graduate Program), and the Milwaukee School of Engineering.

DMSE: A Brief History

The genesis of the Digital Media Software Engineering program was born from the insight and entrepreneurial spirit of Dr. Don Green. He recognized the global impact that software development was having on the world. He also realized that our nation was not creating enough qualified software engineers to meet the local, national, or global demands.

The Digital Media Software Engineering program was created to meet in response to the growing demands of the 2006 economies, which according to the Department of Labor, indicated that America would have a shortage of over 300K Software Engineers in the coming decade. It is important to note that this demand is nearly 1 full year prior to the emergence of smartphones, tablets and smart TVs which have increased the demand for software engineers significantly worldwide.

The DMSE program was created with one main focus, to produce Software Engineers that were competent, capable, and confident employing industry grade software tools, best practices, and project management skills to create quality software applications. To reach this goal at an undergraduate level required substantial research into the software industry current and future needs and an in-depth review of software engineering curriculums nationally and globally. Undergraduate, Graduate, and Doctorate programs were reviewed, and content relevant to the DMSE program were analyzed in great detail, and many aspects of various programs were modeled.

October 1st, 2007 Dr. Don Green hired Frederick Baker as the Coordinator of the DMSE program. Frederick Baker's vision for the program was the same as Dr. Don Green. Don and Frederick both understand the importance of well-prepared students for the corporate workforce, and the necessity to inspire and encourage students to focus on their own entrepreneurial enterprise.

There is only one degree in the DMSE program. The DMSE course material provides the students with traditional engineering best practices and problem solving skills, however there is significant space allocated for enterprising innovations as students create their own application masterpieces.

How do the goals apply to preparing students for careers in and meeting employer needs in the community/region/marketplace?

The DMSE program is designed to prepare students for careers in the Software Design and Development fields. The DMSE faculty actively develops strong relationships with professionals and companies in the software engineering industry.

DMSE immerses the students in 'real world' challenges. While building skills to meet these challenges, students learn how to build software applications using tools, techniques and best practices that utilized by industry professionals. The goal is to have students that are competent and confident to enter the workforce with the skills of a 2 year software engineer.

Additionally the faculty works with students to develop the people skills that keep them working.

DMSE 491 Internship is one of the tools used to achieve the program goals. Students are required to secure an internship. This immersive 400 hour experience is in place to help students make the transition from student to professional. Internship feedback helps provide data on how well the students are prepared for the workforce. Students are permitted to take DMSE 491 after their capstone course to potentially secure a position after graduation.

DMSE 499 Capstone is the final class a student would ultimately take. In it, students self-direct a portfolio level project. The goal is for the student to take all the skills they have learned and apply them to a project that can support their career start-up. Seniors participate in 'Capstone Presentation' that requires them to fully describe the entire Software Application Lifecycle they undertook to complete their application. This is designed as a way to have students rehearse their own 'pitches' as they job hunt. Additionally, the Capstone presentation provides an excellent showcase of the students' abilities which they can leverage for industry employment.

Have the goals changed since the last program review? If so, why and how? If not, why not?

This is the first program review for DMSE

Describe the relationship of the program goals to the University's mission, and the departmental, college and divisional strategic plans.

The mission of Ferris State University:

> *Ferris State University prepares students for successful careers, responsible citizenship, and lifelong learning. Through its many partnerships and its career-oriented, broad-based education, Ferris serves our rapidly changing global economy and society.*

The DMSE program is aligned with the mission of Ferris State University. The DMSE faculty focuses on the growth of students in career skills, civil involvement and lifelong learning. The faculty teaches the students how to solve complex problems in a professional context that demands traditional engineering rigor. Emphasis is placed on teamwork in many of the classes by requiring projects to be produced with a group of likeminded designers.

B. PROGRAM VISIBILITY AND DISTINCTIVENESS

Ferris GR's Digital Media Software Engineering (DMSE) is a high profile program that is creative, distinctive and rigorous. The program is focused on developing the next generation of software engineers.

DMSE is designed as a multidisciplinary degree. Disciplines covered include Project Management, Programming, Design, business, entrepreneurism, liberal studies, and emerging technologies. This format includes a required concentration for depth, an elective section that offers greater breadth in the discipline area, and general education requirements targeted toward courses serving employees in software development.

Using state of the art technology, professional grade software and innovative teaching approaches, Digital Media Software Engineering delivers the education that students will need to succeed in a highly competitive business.

Describe any unique features or components of the program.

DMSE summer camp

The Digital Media Software Engineering Summer Camp is a one week day camp that introduces high school students to a career in software design and development. Students are trained and mentored on leading edge professional grade software engineering tools.

During the summer camp week, students have the opportunity to learn from instructors and students in the field Software Engineering. Learners are exposed to the latest software tools used by industry professionals. They are challenged to develop and increase their technical skills.

Students that enjoy and excel at the introductory Digital Software Academy Course are given the opportunity to continue advancing to the 'Intermediate' and finally to the 'advanced' course. All student payments from DMSE summer camp are added to the DMSE scholarship.

Dual Enrollment with Grand Rapids Community College

Part of the appeal is the partnership with community colleges. All courses are located on the campus of GRCC and share in the Applied Technology Center that was built in 1991. The classrooms are modern, accessible and located close to amenities such as the GRCC field house, cafeteria and parking ramps.

Students who start out at GRCC can transfer over with a strong transfer plan or start their career at DMSE and blend classes seamlessly with their DMSE curricula. The faculty is familiar with many of the classes at GRCC and can advise appropriately according to student interests. GRCC's class offerings are very broad and the diverse population allows the students to fully experience a university-like setting.

One of the most appealing aspects of a GRCC / FSU collaboration is the tuition advantage a student enjoys. With up to 60 credit hours transferable to FSU, Kent county resident students essentially receive a 40 credit hour scholarship to FSU. This is significant for parents who are often helping to finance their student's education and students who are often holding a repayable student load at the end of their time with us.

Ferris DMSE is an amazing value in the educational world partly in thanks to the taxpayers of Kent County. Additionally, the consortium arrangement with GRCC's financial aid department allows us to help students their financial aid effectively.

Describe and assess the program's ability to attract quality students.

When DMSE was first initiated, the entry requirements of a 2.5 GPA and a combined ACT score of 15 needed. While the modest entry level requirements served students who were looking to get into college, it allowed students who were unprepared for the rigors of an engineering program.

In 2008, the decision to increase the entry requirements to 2.7 GPA from high school or Community College, a 21 ACT composite from high school. This goal was to eliminate academically challenged students from the freshman class. Those who had the ambition to improve their overall GPA could attend community college. When the candidate has completed 48 credit hours with a 2.7 or above they could successfully apply to the program with little or no loss of transfer credit.

DMSE has been steadily getting more attention in the Grand Rapids area. However, there is a need for a more aggressive marketing stance that includes billboards, newspaper ads and community activism on a periodic basis.

National Advertising

DMSE has significant growth potential. Consideration should be placed on directed advertising in 5 or more national trade magazines. Just as effective is placement on strategic websites. This will attract the attention of potential students and industry professionals. This will help establish the Ferris DMSE brand.

Summer Camp

DMSE summer camp is a (3) one week experiences for high school students that is designed to give them insights into the software engineering career. Students prepare applications for demonstration at the end of the week. Advertising is done through the college and departmental websites and local newspapers such as the Grand Rapids Press and the Advance. Most students enjoy the experience and many come back for the third week of summer camp that covers more advanced topics. The summer camps are used educate the students on software engineering and to provide a pipeline into the DMSE program. A side effect of this offering is community awareness of software engineering as a career

KCTC Relationship

One of the strongest educational relationships is with Kent County Technical Center also known as KCTC. They offer a 1 year – 2 course sequences that are articulated in Ferris as 6credit hours of college work. The courses cover C# programming and software design. Remarkably, KCTC students tend to be exceptional with good technical, creative and business skills already in place. We have found that upon completion of their courses they are excited about the prospect of attending Ferris to complete their degree in software engineering.

Website

In 2010, the DMSE website was created and could be found independently by search engines. This optimization has greatly improved communication between us and potential students. With the addition of Google analytics the effect of events and other promotional efforts from day to day were traceable. Additionally, the website now lets us create a 'corporate image' that enhances the image of a world class software engineering program.

FSU/GR recruitment team

The marketing director for Ferris has been informed of the needs and benefits of the DMSE program. The team is supported by a graphic designer. The marketing team holds recruiting events and open houses, visits high schools and community colleges daily and pursues new strategies for delivering the message that everyone can improve their lives through education.

High School Visitations

DMSE is involved with many of the area high schools in promoting high tech careers and other new media pursuits. DMSE Faculty work with high school event coordinators and attend career days every semester. With career days, students attend sessions they are interested in. During a typical career day, we can present three sessions of twenty to thirty people per session. Multiply this by 20 to 30 visits a year, and there is an opportunity to promote Ferris as the school of choice for software engineering. The schools that are visited annually are: Union, East Kentwood, West Catholic, Catholic Central, Creston, Forest Hills Northern, Forest Hills Central, College Prep Academy, Hopkins, Byron Center, Coopersville.

Community College Relationships

Throughout Michigan the faculty has established relationships and transfer plans for Digital Media Software Engineering. These two plus two arrangements allow for the efficient transfer of credit from one institution to another. The Colleges that have created the transfer plans include Grand Rapids Community College, Lansing Community College, Wayne County Community College, and Southwestern Michigan College. There are plans of adding to this list of colleges in the coming years.

Identify the institutions that are the main competitors for prospective students in this program.

Competition in Michigan

University of Michigan Dearborn

Grand Valley State University (Masters)

Nearby Michigan

Milwaukee School of Engineering

Rochester Institute of Technology

How are these programs similar and different from the FSU program?

Similarities shared by all the colleges are a respect for a liberal concept of education. All give heed to language arts, science, math, cultural enrichments, business, game design, and social awareness concepts. This commonality underscores the efficacy of a broad educational base. Media arts are by definition art intent to communicate. Each program has a large focus on traditional engineering practices.

The majority of the programs create a broad base of design understanding based in programming. Many high school students have not been trained to program as other educational demands are put on them. The discipline of programming allows for students to study a subject and investigate the basic nature of

something with observation. It is notable that most of the institutions remediate this deficit and provide training in this area.

The difference between the FSU offerings is found in the early introduction and implementation of emerging technologies. FSU also encourages students to pursue entrepreneurial applications that contain course constraints as project selections.

The DMSE program uniqueness is realized in several areas. The first is the relationship that was established with the Grand Rapids Community College. This relationship allows students to effectively take 60+ credits toward their degree at community college costs, which is a substantial cost savings.

In terms of agility, the DMSE program is able to quickly adjust to industry demands without the bureaucratic limitations placed on institutions that require three or more department disciplines to create their software engineering program. This allows the faculty to provide students with traditional software tools and the ability to introduce emerging technologies.

What can be learned from them that would improve the program at Ferris?

In Michigan

University of Michigan

- Reputation built from years of continuity and experience attracting top talent
- Articulations and transfer programs with community colleges
- Active research funding

Grand Valley State University

GVSU has an emphasis on Software Engineering in the Graduate program. While not a direct competitor, GVSU has deep industry connections and donations for project funding.

Outside Michigan

Milwaukee School of Engineering

The MSoE provides an extremely rigorous and extended program that focuses in great depth on mathematics and the sciences, including core engineering classes.

Rochester Institute of Technology

RIT was the first accredited software engineering program in the United States and is considered one of the top software engineering schools in the world. They have consistently reviewed and improved their curricula and have a well-connected advisory board.

Reviewing each institution and the offerings they present to the students indicates that the most important differences are found in their ABET accreditation and funding. It is the goal of the DMSE program to achieve accreditation and a budget that will allow for funding of current and emerging technologies.

C. PROGRAM RELEVANCE.

Provide a labor market demand analysis: This activity is designed to assess the marketability of future graduates. Reports from the Department of Labor and from industry are excellent sources for forecasting demand on graduates.

Labor Market Analysis – Digital Media Software Engineering

Job Outlook – Software Engineers

Jobs: Software engineer rated No. 2 on CareerCast.com's recent listing of the top 200 jobs for 2010, ranked on factors such as physical demands, work environment, income, etc.

CareerCast.com listed these salaries for the above positions:

- Software engineer: $85,139
- Web developer: $60,090
- Computer programmer: $70,178

Dice.com reports current average salaries of:

- Software engineer: $90,031
- Database administrator: $89,742
- Database developer: $84,176

CareerBuilder's salary site reports current average salaries of:

- Software engineer: $90,530
- Computer programmer: $72,661
- Database developer: $95,951

Redmond 2009 Salary Survey, in a survey of Microsoft IT compensation, reported salaries of:

- Programming project lead (Non-Supervisory): $100,635
- Database administrator/developer: $80,894
- Programmer/analyst: $78,818
- Average base salary of respondents: $83,113

Employment projections data for software developers, 2010-20

Occupational Title	SOC Code	Employment, 2010	Projected Employment, 2020	Change, 2010-20	
				Percent	Numeric
Software Developers	—	913,100	1,184,000	30	270,900
Software Developers, Applications	15-1132	520,800	664,500	28	143,800
Software Developers, Systems Software	15-1133	392,300	519,400	32	127,200

SOURCE: U.S. Bureau of Labor Statistics, Employment Projections program

Quick Facts: Software Developers	
2010 Median Pay	$90,530 per year $43.52 per hour
Entry-Level Education	Bachelor's degree
Work Experience in a Related Occupation	None
On-the-job Training	None
Number of Jobs, 2010	913,100
Job Outlook, 2010-20	30% (Much faster than average)
Employment Change, 2010-20	270,900

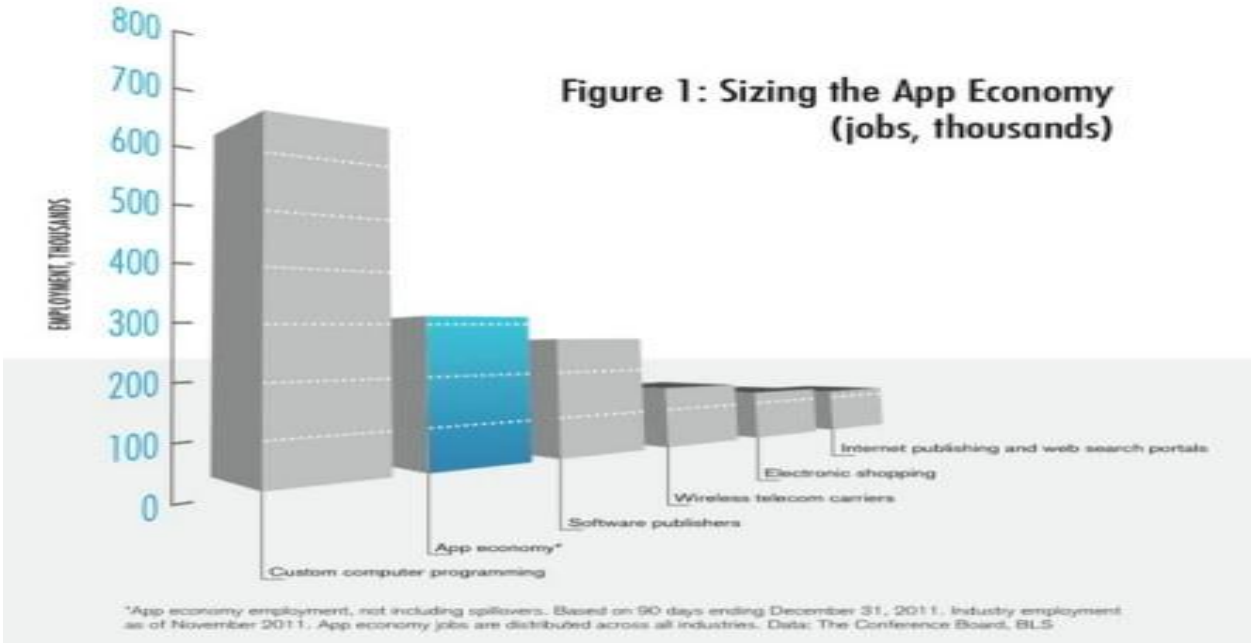
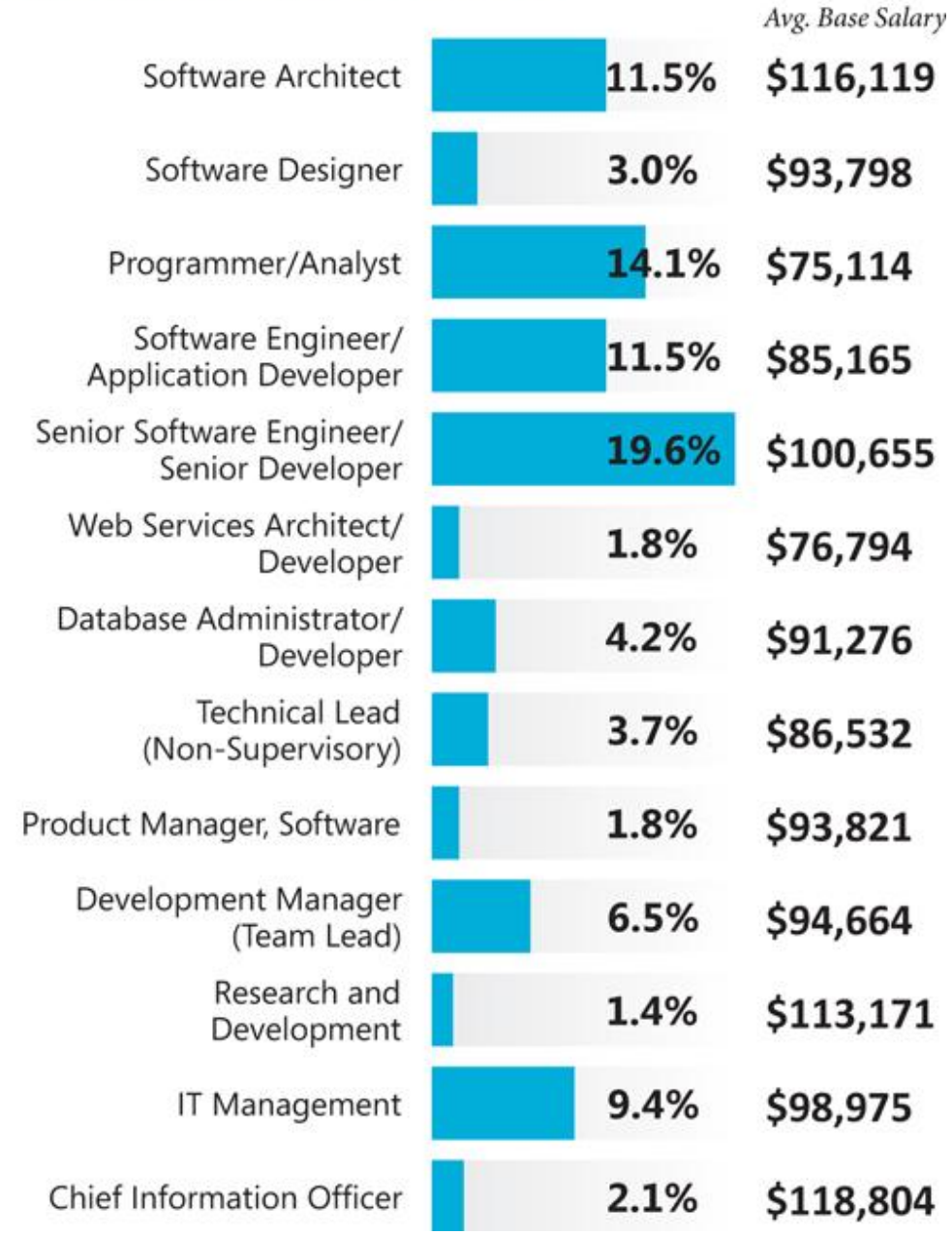


Figure 1: Sizing the App Economy
(jobs, thousands)
EMPLOYMENT, THOUSANDS
800
700
600
500
400
300
200
100
0
Custom computer programming
App economy*
Software publishers
Wireless telecom carriers
Electronic shopping
Internet publishing and web search portals
*App economy employment, not including spillovers. Based on 90 days ending December 31, 2011. Industry employment as of November 2011. App economy jobs are distributed across all industries. Data: The Conference Board, BLS

Salary by Job Title

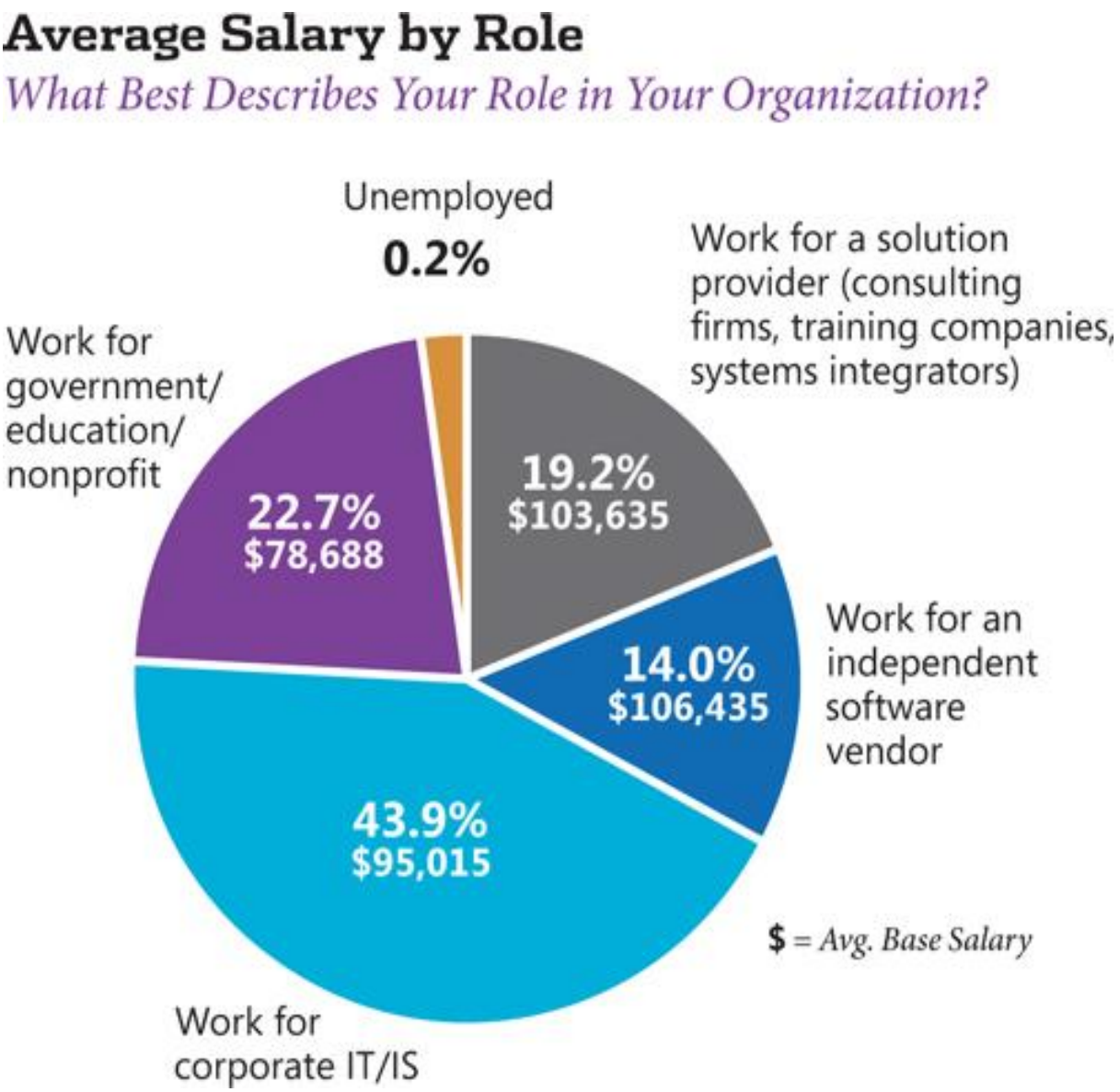
What's Your Primary Job Function?



Average Salary by Industry

What Is Your Organization's Primary Business Activity?

		Avg. Base Salary
Advertising	**0.3%**	**$74,167**
Aerospace	**1.5%**	**$102,996**
Business services/consultancy (non-computer)	**2.5%**	**$99,920**
Communications carrier	**0.4%**	**$72,187**
Computer/IT services (includes programming, ISV, Web dev)	**23.1%**	**$103,940**
Construction/refineries/petroleum	**1.3%**	**$85,966**
Defense/military	**2.9%**	**$98,599**
Education	**8.1%**	**$73,236**
Engineering	**3.8%**	**$106,977**
Finance/banking/accounting	**7.3%**	**$109,836**
Food/restaurant	**0.5%**	**$86,800**
Government	**13.8%**	**$80,448**
Insurance/real estate/law	**3.3%**	**$92,173**
ISP/ASP	**1.0%**	**$108,644**
Manufacturing (non-computer related)	**7.4%**	**$90,788**
Manufacturing (computer related)	**1.6%**	**$91,887**
Marketing/entertainment/ advertising/media	**2.2%**	**$90,325**
Medical/dental/healthcare	**8.0%**	**$90,025**
Research and development	**2.4%**	**$101,100**
Retail and distribution (including online)	**2.4%**	**$85,513**
Telecommunications	**1.6%**	**$82,529**
Transportation/utilities	**4.6%**	**$92,823**



Need for Baccalaureate Education

With this positive job outlook, predicted nationally and within Michigan, it is important that Ferris State University provide educational opportunities to contribute a well-prepared and available workforce to this area of employment.

Over 40 percent of multi-media artists and animators aged 25 to 44 have at least a bachelor's degree, according to data found on the Career One Stop's occupational profile of Multi-Media Artists and Animators (U.S. Department of Labor, Career, 2010).

According to the salary survey (with a 95% confidence level) published by Game Developer in July 2009, there is a clear wage advantage to having a bachelor's degree over an Associate's degree and no college at all.

	Bachelor's Degree	Associate's Degree	High School Diploma
Programming	$82, 867	$76,196	N/A
Art	$66,516	$74,521	$67,500
Design	$67,387	$55,500	$52,045
Production	$83,598	$59,643	N/A
Audio	$73,214	$59,643	N/A
Quality Assurance	$34,737	N/A	N/A

(Duffy, The entry-level, 2009)

A formal education is a very clear path to a career in Digital Media Software Engineering. Students who hold a degree have "clear ways of showing whether they meet the required skills in a job ad" and "earning a degree in computer science or game programming is often a very good investment toward getting a job in the game industry." (Duffy, Beat, 2009)

A formal art education provides artists and animators with an opportunity to collaborate with non-artists and prepares them with a flexible career applicable "not only to games, but film, television, advertising, and other commercial ventures, too." *(Beat the Catch-22)*

References

Duffy, J. (2009). Beat the catch-22. *Game Developer, 16*(7), 9. Retrieved from Academic OneFile

Duffy, J. (2009). The entry-level game developer salary survey. In G*ame Developer Career Guide Fall 2009* (pp. 32-38).

Retrieved from http://gamedeveloper.texterity.com/gamedeveloper/2009fall/#pg1

Michigan Department of Labor and Economic Growth. Bureau of Labor Market Information and Strategic Initiatives. (n.d.). *Arts, Design, Entertainment, Sports, and Media Occupation, Employment Forecast, 2006-2016.* Retrieved from http://www.milmi.org/admin/uploadedPublications/710_occ_g27.htm

Remo, C. (2009). Land of opportunity. *Game Developer, 16*(11), 7. Retrieved from Academic OneFile.

Describe and assess how the program responds to emerging issues in the discipline, changes in the labor force, changes in employer needs, changes in student needs, and other forces of change.

Emerging Issues: The faculty in DMSE researches daily the development in the software and digital industries. A significant aspect of the DMSE program is to remain current and relevant in the software and emerging technologies.

The DMSE program avoids being reactionary in issues regarding emerging practices and technologies. However, a proactive stance is taken in regards to responding to industry demands for entry level

positions for the students. Research is performed on industry job trends by searching job posting boards and listening to industry partners. DMSE faculties regularly attend career trade fairs and discuss with industry what their needs are. Faculty members are encouraged to travel once a year to an event to research and report on the needs of the industry.

The DMSE relies on an Advisory board to remain relevant and synchronized with the industry. The Software Engineering Industry Advisory Committee advises our program faculty on industry trends and needs and helps to keep the curriculum up-to-date and to maintain program goals. The program faculty report to the committee on the program, focusing on recent and planned changes, enrollment, and events.

Most industry members are engineering managers or senior engineers and have a close familiarity with what their company needs from a software engineering graduate. Student members of the committee broaden its perspective by raising issues of importance to their class and participating in committee discussions.

Professional Activities by Staff

2008	Game Developer's Conference	Frederick Baker
2009	Visual Studio Live Developers Conference	Frederick Baker
2010	Visual Studio Live Developers Conference	Frederick Baker
2011	Visual Studio Live Developers Conference	Frederick Baker
2012	Game Developer's Conference	Frederick Baker

How do you discover emerging issues?

Emerging issues are discovered through the advisory board, developer conferences, trade journals, main stream periodicals, companies, consulting, and daily research. Job growth and recommendations from the advisory panel are researched. Emerging issues are received from interns and graduates. With feedback from many varied sources, we are able to meet the immediate and near future needs of the workforce. Additionally, faculty is encouraged to maintain outside projects to keep them in tune with current trends.

How do you see the changing labor force?

The labor force is becoming increasingly more dependent on technology, more specifically, software. Nearly, every physical product today was created using software, manufactured using a machine that is "running" software, or is monitored and maintained using software.

The consumer demand for "Hi-tech" simply translates to quality software controlling intelligently hardware. The most valuable and notable companies in the world to date, Apple, Google, Facebook are all software companies.

The labor force in nearly every industry from Architecture to Zoology is dependent on the efficiency and effectiveness of software. Therefore, the demand for software engineering is increasing as the needs of the world are increasing, with no indication of decrease.

How do you react to changes in employer, student and other needs?

The faculty regularly discusses curriculum improvements with students and employers. The DMSE advisory board is helps determine what skills are needed for their company in the future. However, careful attention is used in order to minimize adoption of fads.

Changes in Employer Needs

With guidance from the advisory council and feedback from alumni, the Faculty examines how the curricula impact the hire-ability of the students.

Each year faculty members attend either the Game Developer's conference or Visual Live Developers conference to assess what is currently needed for employees in the industry. Each conference has 'career fairs' designed to attract the best and the brightest talent in the industry. At GDC, over 40 companies secure trade show career booths and speak to thousands of potential employees. Faculty informally interviews the principles of these companies and gathers data on the types of positions they have available.

Changes in Student Needs

Students often enter DMSE with one goal and work exploring the available career paths. After some exploration, they may discover that they have a particular talent that was heretofore unrecognized. Faculty help students identify this talent and direct them in ways that would be most beneficial to them. Student advisors look at their body of work and point out positive directions for their lives. Assigned advisors craft individual plans with students to help them achieve their goals.

One initiative underdevelopment is online delivery of the curricula. Currently every class has been designed and is prepared to be taught online or delivered in a blended format. The program plan is to have a total of four to six 300 and higher classes available for online or mixed delivery by spring semester 2013. This will allow many non-traditional students to work toward a full four year degree, while potentially working full-time.

The DMSE program has partnered with several Community Colleges across the state as they work on software engineering curricula. Students from these colleges are encouraged to decide early what their career goals are and take appropriate actions. Transfer plans have been created to help students make a smooth transition if they decide to apply to Ferris.

Assess why students come to FSU for the program. Summarize the results of the graduate exit survey and the student program evaluation.

Students come to the program for mainly two reasons. First, they are looking to get into the Computer Animation industry and work on film projects such as Star Wars, Avatar and Shrek. However, the soon realize that they really want to develop software and transfer into the DMSE program.

Secondly, they may have head a presentation from the DMSE Program director, attended the DMSE summer camp, or was fortunate to have seen or heard an advertisement regarding the DMSE program.

As of spring 2012 the Kent County Technical Center (KCTC) has become an additional source of qualified and interested students.

The graduate exit survey indicates that the students believe that they have received a quality education in software engineering that more than adequately prepared them for the workforce. However, they have indicated that they would have appreciated a larger offering of classes each semester. The students also indicate that the Software Engineering certification class helped them solidify what they had learned, but also felt that it would have been of greater benefit if they would have taken the course two semesters prior.

In addition, to the formal survey questions, each student has communicated through email their informal evaluation:

Student1 (www.Manatron.com)
Hey there old friend. Hope all your classes and everything is going good. I just got my first job finally after two or three interviews so things are good right now for me. Manatron is the name of the place. Thank you for all your hard work being my professor. I couldn't have had a better one. Take care.

Stundent2 (www.salespad.net)
I just thought I would give you the heads up about my new and improved software development job! I was hired for a salaried C# position at SalesPad. I just finished my first week there and it is an awesome gig. I even get my own office :) I Love my Job!! How's everything going with the program? Are more people starting to sign up? Well I hope your summer is going as splendidly as mine!

Student3

I just wanted you to be one of the first to know that I received an offer from Springthrough this morning. I have sent my acceptance and am slated to start on Monday June 18th. I'll be salaried as a full time employee and the job offers health care benefits for me and my family. Thank you so much for all your help, belief in me, your time and words. I don't think I would be here without you.

Student4 (www.Avanade.com)
Anyways, I'm over in Seattle right now
on a training project. We are doing some asp.net with SharePoint 2010 integration. Btw, SharePoint is BIG in this company. I've been keeping really busy for the past week here, and I'm going to New York on the 25th. So far everybody has been really impressed with the experience I gained with Ferris (mainly the fact that it's a college campus that emphasizes .net) and the database experience I've gotten with Broadway, and I've been proposed for three projects so far. I really want one of them which would take me to Toronto Canada for a few months.

My training project has been with a small team of about 6 people, only 3 of which (including myself) are software developers. Two others are systems engineers, and another is a business analyst. Everything we did in Ferris has really helped me and has directly applied to my experiences here. So far a lot of the training has been simply rehashing on concepts we covered in detail during the program.

Graduate Survey - Exit Interview

You are now (or soon to be) an alumnus of Ferris State University! Congratulations and we hope your hard work and dedication pay personal and professional dividends in the years to come. We'd like to ask you for some of your opinions that speak about the overall DMSE program and your experience at Ferris. Not only does this type of feedback offer us an opportunity to improve the program, Ferris State University require this type of information, and it is also required by accreditation bodies to demonstrate efforts for continuous improvement and striving to deliver quality and worthwhile education. Please take some time to provide us with information as to how you see the program now, as well as suggestions for possible improvement in the future.

1. **Name the three most valuable items you learned in the DMSE program**
   - Software Engineering concepts
   - Documentation, as far as design docs, tech docs, and the like is hugely important, although it may be an odd choice.
   - Programming
   - Collaboration Skills
   - Software Engineering Tools

2. **Name at least two items that were covered extensively in the program that you wish were not covered, because you either:**
   a. **Had previous extensive knowledge**
   b. **Had no need for coverage of this topic**
   -

3. **Were there any particular topics that you wish were covered that were not, or those that needed to be covered more extensively? (name these and comment)**
4. **On a scale from 1 (poor) to 10 (excellent), rate the overall quality of your education in the Ferris DMSE program**

(Poor)		(Undesirable)			(Useful)				(Excellent)
1	2	3	4	5	6	7	8	9	10

(8 5 7 6 7 9)

5. **On a scale from 1 (poor) to 10 (excellent), rate the overall quality of instructors from the DMSE program**

(Poor)		(Undesirable)			(Useful)				(Excellent)
1	2	3	4	5	6	7	8	9	10

(6 6 7 4 7 9)

6. **From question 4 (instruction), are there any specific instructors that stand out as either excellent or poor? (You can comment on any number of instructors and use the back of this page as necessary)**

7. **Below is a table which contains key items identified as useful and critical for graduates of the DMSE program. The list is presented in alphabetical order. On a scale from 1 (poor) to 10 (outstanding), please rank each of these skills as their importance AND how effective the DMSE program was in delivering content and skill building in this area of emphasis.**

8. **In your opinion, what should change in the coverage of topics in the current program? Why?**
9. **In your opinion, what are the strongest features of the Ferris DMSE program?**
10. **In your opinion, what are the areas that require the most improvement in the Ferris DMSE program? Why?**
11. **Is there anything else you'd like to comment on that we didn't ask about?**

How well does the program meet student expectations?

The student surveys indicate that students are reasonably pleased with the quality of education that they received at Ferris.

How is student sentiment measured?

Student Assessments of Instruction: We measure student sentiment by the Student Assessment of Instruction every semester. Part of the assessment allows students to feed back with comments. Classes have benefitted greatly from this information and continually improve. All classes are assessed and students are encouraged to provide comments to improve the class. This provides semester by semester feedback that lets us respond quickly and affirmatively to improve class content or fix problems.

Advising: Students are provided informal questions regarding courses they have taken to provide feedback on how successful they believe the program is at helping them achieve their goals Advising appointments allow for feedback on general education experiences at GRCC as well.

Gaming together: We stay close to our students by playing games with them on Xbox Live and the PlayStation Network. We use 'in-game' chat with students which provide them with an opportunity to discuss their concerns regarding the program in a relaxed and familiar setting.

D. PROGRAM VALUE.

Please refer to the faculty survey.

Describe the benefit of the program, facilities, and personnel to the University.

The computers we have are a cut above the standard classroom computer. They feature dual core Pentiums with 4 gigabytes of ram and high end graphics cards. Dual monitors show students what they are doing and they are packed with state of the art software. Quality classroom projectors and full stereo sound systems make the students' work shine during critiques.

The faculty of DMSE is unique. Each faculty member has 20+years of professional experience prior to teaching in the classroom. This lends credibility to the program. When parents and potential students talk to us, we can speak to them from professional experiences.

Describe the benefit of the program facilities, and personnel to the students enrolled in the program.

Facility: The Applied Technology Center at Ferris/DMSE is a versatile building that serves students adequately. Shared with GRCC, we have two restaurants/classrooms, a large atrium, patio, students lounges and classrooms all within walking distance of a vibrant urban landscape. Administration is convenient and help is just a few steps away for students. Faculty offices are in direct proximity and the auditorium space near the classrooms. Students have short walks to GRCC classes that make up their general education requirements. Two frequent complaints from students is that there is no 'communal' space that is 'Ferris' and that the classrooms can get overheated as the computers tend to use significant power.

Personnel: Besides having exemplary faculty, DMSE has the privilege to be served by a skilled and dedicated administrative staff. The Ferris Online Appointment software helps student's book appointments, direct traffic. Financial aid councilors are available by appointment or by 'drop in' subject to availability. Students need not go any farther than their own halls to get advice on financing their education. Registration also is immediately available as is student service councilor and career service councilor. Students are cared for and helped quickly and politely by the first rate staff.

What is the assessment of program personnel of the value of the program to employers?

DMSE teaching faculty had industry careers before becoming faculty at Ferris. Having the unique opportunity to start 'from the ground up', the DMSE program was crafted to provide each student with the resources necessary to succeed in software engineering.

Explain how is this value is determined.

This value is determined on street level. Many of the students stay in contact with us via social networking sites such as Facebook and Linked In. The faculty remains in close contact with the alumni and follows their successes and joins in on their challenges. Employers also provide necessary feedback regarding the value and relevance of the knowledge or the students.

Describe the benefit of the program, faculty, staff and facilities to entities external to the University (services that faculty have provided to accreditation bodies, and regional, state, and national professional associations; manuscript reviewing; service on editorial boards; use of facilities for meetings, etc.).

DMSE teaching faculty are involved professionally, improving the climate of software engineering in Michigan.

What services for extra-University general public groups (e.g., presentations in schools or to community organizations) have faculty, staff or students provided? Describe how these services benefit students, program, and community.

DMSE faculty members are involved with the community. The faculty embraces the opportunity to be part of the programming, game design and media community.

The faculty is actively involved in:

- Kent County Technical Center Advisory Chairperson 2008 - 2010
- High School Career Path presenter 2007 – 2010
- Jr. Achievement presenter
- AIMwest
- Microsoft Partner
- Intel Parallel Processor Partner
- FIRST Robotics
- AppItNow
- IGDA, the International Game Developer's Association
- Wayne County Community College board

Section 2: Collection of Perceptions

The survey sections must include, among others, a discussion of techniques used in collecting the information, difficulties encountered during the surveying process, number and percent of respondents, and analysis of data in accordance with established methodologies. The survey instruments must be designed and distributed, in consultation with Institutional Research and Testing, to reflect general aspects of program review as well as the specific nature of the program itself. All comments should be included, but the names of individuals mentioned should be deleted.

Graduate follow-up survey:

The graduates their perceptions and experiences regarding employment based on program outcomes have been very affirming. Each student that has responded to the survey has indicated that they were well prepared for the workplace. Several students have indicated that they were actually on par with software engineers with 3 to 5 years industry experience. The students have strongly indicated that the program needs to include additional course material in the area of Database design and implementation. This feedback has largely come from the first cohort of students to complete the program. To fulfill their valid need to have industry level Database experience, students are introduced to Database design and best practices in their early 300 level software engineering courses. All courses moving forward have a

minimum of one project that must incorporate the use of a database. The data from the Graduate survey is located in Appendix A

Employer follow-up survey:

Employers' experiences with graduates and their perceptions of the program have been extremely positive. To date employers of student interns and graduates have indicated two consistent responses. The first response is that they are impressed with the level of programming knowledge and student understanding and appreciation of formal best practices. Secondly, employers are impressed with the courses that are offered, with many saying that, the program we offer is exactly we the industry needs and want to partner with Ferris for Software Engineering graduates. The data from the Employer survey is located in Appendix A

Graduating student exit survey:

The graduating students surveyed regarding quality of instruction, relevance of courses, and satisfaction with program outcomes have indicated that their expectations were met or exceeded, in most areas. Unfortunately, there are several areas that the students have indicated that need improvement. The first area is in Database design and implementation. The students would like to have more classes offered during the evening and the opportunity for online/mixed delivery. The survey must seek student suggestions on ways to improve the effectiveness of the program and to enhance the fulfillment of their expectations. The data for the Alumni survey is located in Appendix A

Faculty perceptions survey:

Faculty perceptions regarding:

Curriculum: The curriculum has a solid traditional engineering best practices foundation with emphasis on remaining relevant through the introduction and review of emerging technologies. Within the next few years courses must include greater emphasis and exposure to the students in the areas of mobile application development and cloud computing.

Resources: To ensure that the students are at the leading edge of the industry upon graduation requires a greater commitment in the acquisition of emerging software tools and technologies.

Admissions standards: Admission standards to date have not been a negative factor, due to the rigorous interview process prior to acceptance into the program. However, as the program continues to grow the admission standards should increase from a GPA of 2.7 to 3.0.

The data from the Faculty Perception survey is located in Appendix A

Faculty Perceptions of Program

How we can do things better as a department?

Improvement needs to be made in the area of scheduling meetings with students that have not taken a SENG course in two consecutive semesters. An improvement in this area will lead to greater student retention. Providing more connections with professional organizations and clubs for the students to join is also a necessary improvement.

Personal faculty goals for the next two years:

To develop a deeper approach to software methodologies in the area of Test Driven Development and Architectural design patterns.

To gain greater knowledge in the use of CryEngine 3, UDK3, and Unity 3D.

To establish a broader and deeper relationships with the organizations that employ or students.

To establish a minimum of 5 top-tier software companies to employ the our students

General Departmental Directions

What additional concentrations and classes are appropriate?

Currently reviewing courses in Circuit Analysis, Digital Circuits, and Digital Forensics.

How we can improve retention?

Improving the tracking and communicating with students that do not register for at least one class that leads to their DMSE degree.

How we can improve cross class collaboration?

Currently, the DAGD and DMSE staff are actively comparing courses that student in each program can complete that will enable them to have a broader and deeper knowledge base in their career of choice. To improve cross class collaboration requires a semester to semester meeting to review each programs current and projected offerings. DAGD and DMSE program collaborate on approximately 6 courses.

To wrap up the essay, please answer this question: "If we had to do it from scratch again, we should…."

We should have advertised to the business community, community colleges, and high schools at least 1 year prior to starting the program.

Advisory committee perceptions

The purpose of this survey is to obtain information from the members of the program advisory committee regarding the curriculum, outcomes, facilities, equipment, graduates, micro- and megatrends that might affect job placement (both positively and adversely), and other relevant information. Recommendations for improvement must be sought from this group (combine notes from meetings). The data for the Advisory committee survey is located in Appendix A

Section 3: Program Profile

Include Administrative Program Review document in this section. Provide the number and percentage for the variable addressed for each of the years since inception (for new programs) or the last program review.

A. PROFILE OF STUDENTS.

1) Student Demographic Profile.

Gender: 95% Male

Ethnicity: 100% Caucasian

Average Age: 23 years old

Residency (In-state and out-of-state)

Full-time and part-time.

Data unavailable

Attend classes during the day, in the evenings, and on weekends.

Data unavailable

Enrolled in classes on- and off-campus. 0%

Enrolled in 100% on-line and/or mixed delivery courses. 0%

Blended Courses: 0%

Discuss how the information presented in (a) through (f) impacts the curriculum, scheduling, and/or delivery methods in the program.

Women comprise about 15% of the student body. An overwhelming 90+ of the students are Caucasian and have an average age of 23. Most students work part-time or have full time positions in retail. All attend classes at the Grand Rapids facility, but some are taking classes online to help reduce travel time or to continue attending a local community college before fully transferring to the DMSE program in Grand Rapids.

As DMSE enters the next phase of development, additional work with community colleges will provide courseware at their schools or online, which will enable distant students to pursue a DMSE degree.

Several classes are now offered from 5:00 pm - 9:00, to help full-time workers an opportunity to obtain their DMSE degree.

2) Quality of Students.

In addition to ACT and GPA, identify and evaluate measures that are used to assess the quality of students entering the program.

With the distinction of becoming an "Engineer", the quality of the student applying to the program is much higher than the typical student. The active recruiting at high schools and community colleges, and clearly discussing the qualifications of an engineer, has in effect resulted in applications from students that have been preparing for engineering for several years.

Identify academic awards (e.g., scholarships or fellowships) students in the program have earned. Comment on the significance of these awards to the program and students.

Scholarships: Students who participate in internships outside the state are encouraged to submit for the DMSE Scholarship. They can receive up to $1000 to help defray tuition expenses if they accept an internship that will require travel outside of Michigan. The scholarship is funded by DMSE summer camp profits.

What scholarly/creative activities (e.g., symposium presentations, other presentations or awards) have students in the program participated in? Comment on the significance of these activities to the program and students.

The Game Developers Conference (GDC) is outstanding event that is a blend of all of the above. It is the main game developer conference in the world, attracting over 40,000 animators, programmers, artists and marketing teams showing the latest computer technologies. Software Engineers are highly regarded at GDC and people line up to get their autographs.

What are other accomplishments of students in the program? Comment on the significance of these accomplishments to the program and students.

Students in DMSE are growing and developing their programming and engineering abilities. The students continually impress the faculty with their projects. Students have created software applications for the iPhone, iPad, Android Phone, Web, Facebook, Xbox 360, Microsoft Kinect, Educational Software, Google, Twitter, Computer Labs, Wi-Fi devices, Desktop PCs, and soon quadric-copters. There is no limit to what they are able to create with the resources that have been provided by the DMSE program.

Employability of Students

How many graduates have become employed full-time in the field within one year of receiving their degree? Currently, we have four graduates that are employed full-time in capacity of a software engineer. This accounts for 100% graduate employment. It is important to note that every student to date that have completed a minimum of 4 semesters in the program have completed internships. Each student has also been offered a full-time position upon their graduation.

What is the average starting salary of graduates who become employed full-time in the field since inception (for new programs). The average starting salary of graduates employed full-time is 52k.

How many graduates have become employed as part-time or temporary workers in the field within one year of receiving their degree? Comment on this data. All are employed full-time.

Describe the career assistance available to the students. What is student perception of career assistance?

The career assistance councilor in Grand Rapids is available by appointment to discuss career options and strategies. The current methodology for career development includes mock interviews, resume writing and portfolio review as part of the capstone class.

As seen in the alumni surveys, the greatest help they could get is in career services.

How many graduates continue to be employed in the field? 100%

Describe and comment on the geographic distribution of employed graduates.

Only one graduate chose to leave Michigan.

However, he does plan to return to Michigan in 2 years.

How many students and/or graduates go on for additional educational training? (Give annual average.) Comment on this data. Two graduates have committed to completing a Masters in Software Engineering

Where do most students and/or graduates obtain their additional educational training? Comment on this data. NA

ENROLLMENT

Fall Enrollment for DMSE 2008 - 2011

2008/2009 = 10

2009 / 2010 = 11

2010 / 2011 = 16

2011 / 2012 = 15

What is the anticipated fall enrollment for the program?

Anticipated Fall Freshman enrollment for 2012 is 20.

Have enrollment and student credit hour production (SCH) increased or decreased since the last program review? This is the first program review

2008/2009 = 10

2010 / 2012 = 15

There has been an increase in credit hours in the freshman column. The trend is toward continued growth as the program has received increased attention.

Since the last program review, how many students apply to the program annually?

This is the programs first review, however, the average number of students applying to the program is 15 to 20.

Of those who apply, how many and what percentage are admitted?

Approximately 80%+ new student admit enrollment rate to DMSE based on 2007-2012 data.

Of those who are admitted, how many and what percentage enroll?

The estimated percentage that enroll is 90%+

What are the program's current enrollment goals, strategy, and efforts to maintain/increase/decrease the number of students in the program? Please explain.

The program enrollment goal is 15-20 quality freshman students and 5 quality transfer students. This is based on classroom capacity and anticipated graduation target of 12 students a year. Attrition is expected to be higher than average rate because of the nature of the students discovering the rigors of engineering

Students are attracted in a number of ways.

Relationships

Initially the program founder, Dr. Donald Green saw the need to partner with Grand Rapids Community College to help create opportunity for their students. This healthy symbiotic relationship is fostered by articulation of up to 60 credit hours that can be taken at GRCC to be applied to the Ferris DMSE degree. The majority of the students are GRCC students. This provides services to the students along the lines of a University but also benefits students from a cost standpoint as it represents over a sixty percent discount for half of their degree based on current tuition rates.

The DMSE program now has a close relationship with Kent County Technical Center. Their programming curriculum helps develop students who have a programming career in their future. Their new 'Software engineering' class is available to students who have completed introductory programming classes. When students complete their one year sequence they will have earned six college credits that can transfer directly into the DMSE program. This attracts the best from KCTC in this manner and should provide three to five students a year to the program.

Faculty visit 10 or so high schools and middle schools a year and attend 'career days'. A typical 'career day' exposes the program to sixty to seventy students who sign up for various seminars

Summer Camp

Summer Camp is another recruiting tool. This is a 'low risk' method of having students sit in the driver's seat and see if they want to pursue this as a career. The intent of the camps is to educate the students on how exciting software engineering is and to eventually have them apply to the DMSE program citing 'camp' as the motivating factor in their decision.

PROGRAM CAPACITY

What is the appropriate program enrollment capacity, given the available faculty, physical resources, funding, accreditation requirements, state and federal regulations, and other factors? Which of these items limits program enrollment capacity? Please explain any difference between capacity and current enrollment.

There is one classroom with 25 computers and seats dedicated to the DMSE program. Overall, the space is flexible and leverages the advantages that a community college can offer. Capacity of the program is scalable within some limits. In order to focus on the quality of the student that graduates from the program, a deliberate attempt is made to limit class size. This is not to imply that there is a limit of the number of students that are allowed to be active in the program.

Communal Space

First is the need for a communal student gathering area in the immediate area. Students enjoy meeting, playing tabletop games and enjoying lunch together at a central location near their classrooms. Currently, students use and sit on the well-travelled and worn floor to gather on. While charming, this does not present a professional image to parents and prospective students as they are toured around the building.

Lab Space

Lab space is only available through scheduling classrooms as labs.

RETENTION AND GRADUATION

Give the annual attrition rate (number and percent of students) in the program.

Retention and Graduation Rates of Full Time FTIAC Students

What are the program's current goals, strategy and efforts to retain students in the program?

The department goal is to retain 80%+ of the students who begin in DMSE. Accurate and informed advising is one of the best strategies to help with retention. Helping the student's balance their workload can avoid burning students out. Also, a reasonable and gradual course complexity distribution aids in retention as well.

Jobs are the Key to future success in DMSE

Describe and assess trends in number of degrees awarded in the program.

The trend is that most students that complete 2 years of the program continue to finish their degree, within 4 years.

How many students who enroll in the program graduate from it within the prescribed time? Comment on any trends. Approximately 80%

On average, how long does it take a student to graduate from the program? Please comment.

On the average, it takes about 4 years to complete the DMSE BS degree.

The typical student takes more time to graduate than a typical Ferris student. We believe this is attributable to a number of factors. First, the students tend to work part time which reduces their typical load to twelve full time hours. Secondly, many of the students are transfer students who do not have fully articulated classes so they need to take more for the DMSE program. Thirdly, the students would rather take fewer classes and do a better job. The business is driven by good portfolios and students work hard achieving them. Fourth, the students have a 400 hour internship which is only a three credit class. This increases the amount of time it takes to graduate.

ACCESS

Students have not experienced any difficulties with course access or timely offerings. In the rare instance that this may occur, a student may be advised to select an independent study.

Through partnerships with community colleges, we will have the ability to provide greater coverage and accessibility for the students.

CURRICULUM

Program requirements. Describe and assess the program-related courses required for graduation.

Course Descriptions

Programming	**SENG 101**	**Computer Programming 1** This course teaches fundamentals of computer programming. Students learn how to write, test and debug small programs. Basic coding concepts and best practices are discussed and explained. Functions, data types, logical constructs required to produce software solutions will be the basis for this exploration.
	SENG 102	**Computer Programming 2** This course is designed to complement Computer Programming I and begin to explore more sophisticated programming concepts. Students will begin to learn advanced coding techniques and structures such as Classes, Inheritance, and Polymorphism. (Prerequisite SENG 101).
	SENG 300	**Software Data Structures** This course explores advanced and abstract data structures. Students will be able to learn how to identify and correctly apply these structures to solve more complex programming problems. Topics include Sorting, Trees and Hash tables. (Prerequisite SENG 102)
	SENG 301	**Software Engineering Tools and Practices** **This course provides an introduction to industry accepted software engineering tools and practices used to develop quality software applications. Students will learn modeling and design using the Unified Modeling Language (UML), Application Revision Control, Code Analysis, Unit Testing, Code Coverage, Coding Best Practices, Continuous Integration, and Code Performance/Profiling.**
	SENG 400	**Engineering Enterprise Software Applications** This course discusses the concepts involved in designing large scale applications. Architectural concerns are examined along with issues stemming from multiple concurrent users. Students will also gain an understanding of the importance of performance when implementing applications that have high volume usage. (Prerequisite SENG 350)
	SENG 430	**Programming Graphical User Interfaces** This course focuses on the techniques and technologies employed in creating Software User Interfaces (UI). Students learn the importance of clean user interface design along with gaining an appreciation for programming techniques used to enhance the human user experience. UI methodologies and frameworks are explored. (Prerequisite SENG 350)
Foundation	**SENG 160**	**Software Engineering Methodologies and Processes** This course covers the processes and procedures practiced by software engineering organizations. Methodologies and processes are presented within the framework of the software development lifecycle. Covers popular methodologies being used in the real world and examines the merits of each. Students learn theory and process as well as examine the effects through case study and applied scenarios.
	SENG 170	**Software Requirements Management** This course covers the tools and methods involved in capturing and tracing end user requirements through the software development process. Examines dynamics and scenarios that organizations deal with in identifying requirements and processes used in defining software needs. (Prerequisite SENG 160)
	SENG 200	**Software Packaging, Configuration & Deployment** This course introduces students to technologies and techniques associated with packaging and deploying software products. Students learn the basic technologies involved with software installations and progress from simple desktop application deployments to full enterprise applications. This course also explains software configuration management approaches for maintaining large and small scale software projects. From the time the first line of code gets written to the time the software package is released to the public, Configuration Management plays an integral role in ensuring all software components are included in the package. Additionally strategies for managing code lines and employing automated process and tools will be examined. (Prerequisite SENG 160)

	SENG 302	**Software Quality Assurance** This course introduces the concepts of software quality assurance. Students learn processes involved in quality management. Topics include designing test cases, improving software quality, testing methods and tools. (Prerequisite SENG 160)
	SENG 350	**Software Design and Architecture** This course examines the process and design techniques employed by software architects to design enterprise scale software. Students will be introduced to design tools as well as diagramming techniques and other methods for communicating software designs. (Prerequisite SENG 160, 300)
Mathematics	**MATH 220**	**Analytical Geometry & Calculus 1** The first of a three-semester sequence in analytical geometry and calculus. Topics include: the limit, the derivative, differentiation of algebraic and transcendental functions, and definite and indefinite integration. (Prerequisite: MATH 126)
	MATH 230	**Analytical Geometry & Calculus 2** The second of a three-semester sequence in analytical geometry & calculus. Topics include: applications of integration, integration techniques, infinite series, conic sections, parametric equations, & polar coordinates. (Prerequisite: MATH 220)
	MATH 332	**Linear Algebra** An introduction to the theory of vector spaces with emphasis on matrix algebra. Topics included are linear transformation, independence, rank, and inverses. (Prerequisite: MATH 220)
	MFGE 341	**Quality Science Statistics** Practical aspects of sampling, data presentation, measures of central tendency and dispersion, basic probability theory, the normal probability distribution, the sampling distribution of sample means and sample proportions, confidence intervals and hypothesis tests for one-sample designs, simple linear regression and correlation. (Prerequisite: MATH 116)
	MFGE 423	**Engineering Economics** Designed to advance the student's knowledge in the subject of engineering economic analysis. Money and time relationships in respect to capital purchases and equipment justification are discussed in detail. (Prerequisite: MATH 126)
	MFGE 442	**Design of Experiments 1** Practical aspects of sampling, data presentation, measures of central tendency and dispersion, basic probability theory, the normal probability distribution, the sampling distribution of sample means and sample proportions, confidence intervals and hypothesis tests for one-sample designs, simple linear regression and correlation. (Prerequisite: MFGE 341)
Business	**PROJ 420**	**Project Management** Students use Project Management techniques to successfully plan, schedule, and manage projects.
	BLAW 321	**Contracts & Sales** Provides an introduction to the law and the legal system in the U.S. as well as a thorough examination of the law of contracts and sales. (Includes a review of articles 2 and 6 of the Uniform Commercial Codes.)
Professional Development	**SENG 420**	**Software Development Industry Certification** This course allows students the ability to prepare for a highly regarded industry certification program. Students will go through the recommended training materials and coursework and prepare to take the certification exam. On course completion the exam is taken and certification earned. Certification is not guaranteed and dependent on student's ability to pass exam. Extra exam fee applicable
	SENG 491	**Software Engineering Applied Internship** This course is intended to be completed between the Junior & Senior year. The internship shall be setup and approved by means of a internship contract, including approval by the University & employer in a related Software Engineering field. (Prerequisite senior status)
	SENG 499	**Capstone in Applied Software Engineering** **This course will focus on helping each student prepare for a career creating commercially viable software. A team of Students establish a software business. The students will follow the Software Engineering processes, procedures, and methodologies learned to create a software application from "Design to Delivery" (Prerequisite senior status)**

As part of the graduation requirements of the current program, list directed electives and directed General Education courses. Provide the rationale for these selections.

Application Domain	4 Course Sequence (consult advisor for details)	**Application Domain Sequence** This group of courses is comprised of a series of related courses pertinent to the student's particular area of focus. This sequence is designed to give the student an opportunity to apply skills learned specifically in the program to an area related or closely related to the student's intended area of career focus. Typical sequences of courses deal in topics listed below. An academic advisor will assist students in choosing the sequence of courses, provided certain conditions are met. These three courses should include at least two courses at the 200 level or above and be closely related to each other. Some examples of common domain sequences are shown below.
	Business	BUS 122 – Introduction to Business ACCT 201 – Principles of Accounting MGMT 301 – Principles of Management MGMT 370 – Quality and Operations Management
	Game Development	DAGD 150 – Intro. to Game Design & Development DAGD 260– Multimedia 1 DAGD 300 – Game Design Theory DAGD 320 – Multiplayer Game Programming
	Software Architect	ISYS 200 DB Design ISYS 330 System Analysis and Design ISYS 371 Adv DB Design ISYS 470 DB Admin

Business rationale: Many of the students have a desire to start their own software development business and this will require them to have at least a base understanding of how to manage their entrepreneurial venture.

Game Development rationale: Increasingly students have the desire to transition from a consumer of video games to a creator and producer of video games. The video game creation industry is a multi-billion dollar industry that has continued to grow over the last decade. The Game Development domain will allow students the opportunity to produce and prosper with passion in a game development career.

Software Architect: As the demand of software ubiquity continues to increase, the demand and necessity for software that has the ability to scale in functionality and delivery increases as well. To meet the demand students must understand how to architect their software to meet these inevitable needs.

"G" Global and "REG" Race, Ethnicity and Gender Courses Under Social Awareness:

ANTH 122 (AN 210 at GRCC) – Introduction to Cultural Anthropology [G/REG]

GEOG 100 (GE 135 at GRCC) – Geography of World Regions [G/REG]

GEOG 112 (GE 210 at GRCC) – Cultural Geography [G/REG]

*PLSC 323 – International Organization [G/REG]

*PLSC 331 (PS 201 at GRCC) – Comparative World Governments [G/REG]

*PLSC 341 (PS 202 at GRCC) – International Relations [G/REG]

SOCY 335 (SO 270 at GRCC) – Marriage and the Family [G/REG]

*FSU PLSC Courses are taught in Grand Rapids primarily during the Summer Semester.

REG Courses Under Cultural Enrichment:

EN 270 (at GRCC) – Multicultural Literature [REG only]

LITR 202 (EN 271 at GRCC) – Black Literature [REG only}

PO 103 (at GRCC) – Introduction to Photography [REG only]

Global Courses Under Cultural Enrichment:

Any foreign language course from GRCC

HIST 152 (HS 102 at GRCC) – Western Civilization Since 1500 [Global only]

HIST 373 (HS 290 at GRCC) – Twentieth Century Russia [Global only]

MUSI 232 (MU 107 at GRCC) – Introduction to Music Listening [Global only]

Note: MU 107 comes into FSU as a 200-level course and meets that requirement.

Additional REG Courses Under Social Awareness:

PLSC 121 (PS 110 at GRCC) – American Government 1 [REG only]

PSYC 150 (PY 201 at GRCC) – Introduction to Psychology [REG only]

PSYC 226 (PY 232 at GRCC) – Lifespan Human Development [REG only]

PSYC 341 (PY 233 at GRCC) – Child Psychology [REG only]

PSYC 342 (PY 234 at GRCC) – Psychology of Adolescence [REG only]

SOCY 121 (SO 251 at GRCC) – Introductory Sociology [REG only]

SOCY 122 (SO 254 at GRCC) – Social Problems [REG only]

SOCY 340 (SO 260 at GRCC) – Minority Groups in America [REG only]

Proposed Course Schedule Breakdown by Semester

		Year 1			Year 2			Year 3			Year 4		
	Description	Fall(1)	Spring(1)	Summer(1)	Fall(2)	Spring(2)	Summer(2)	Fall(3)	Spring(3)	Summer(3)	Fall(4)	Spring(4)	Summer(4)
SENG 101	Computer Programming 1	1											
SENG 102	Computer Programming 2		1										
SENG 300	Software Data Structures (SENG 102)			1									
SENG 301	Programming Languages (SENG 300)				1								
SENG 315	Software Component Design (SENG 300)/API					1							
SENG 400	Engineering Enterprise Software Applications (SENG 350)						1						
SENG 430	Programming Graphical User Interfaces (SENG 350)							1					
SENG 160	Software Engineering Methodologies and Processes	1											
SENG 170	Software Requirements Management (SENG 160)		1										
SENG 210	Software Configuration Management (SENG 160)			1									
SENG 302	Software Quality Assurance (SENG 160)				1								
SENG 350	Software Design and Architecture (SENG 160, SENG 300)					1							
SENG 355	Software Engineering Tools (SENG 160)						1						
SENG 420	Software Development Industry Certification (SENG 350, SENG 355)							1					
SENG 491	Applied Internship (SENG 301, SENG 302, SENG 315)								1				
SENG 499	Capstone in SENG (SENG 491)									1			
	Working at Internship										*	*	*
STQM 260 / MA 215	Intro to Statistics	1											
See Advisor	Design of Experiments 1 (MFGE 341)		1										
Application Domain	Application Domain			1									
Application Domain	Application Domain				1								
Application Domain	Application Domain					1							
Application Domain	Application Domain						1						
	All GRCC Courses Below this Line in red												
MATH 220 / MA 133	Calculus with Analytical Geometry 1 (MATH 126 / 130)							1					
PHYS 241 / PH 245	General Physics 1 (MATH 220) (AP transfer)								1				
ECON 221 / EC 252	Principles of Macroeconomics (MATH 110)									1			
MATH 230 / MA 134	Calculus with Analytical Geometry 2 (MATH 220)										1		
MATH 322 / MA 257	Linear Algebra (MATH 220)										1		

PHYS 242 / PH 246	General Physics 2 (PHYS 241 and MATH 230)											1	
ENGL 150 / EN 101	English 1	1											
ENGL 250 / EN 102	English 2 (ENGL 150)		1										
ECON 222 / EC 251	Microeconomics			1									
COMM 121 / COM 131	Fundamentals of Public Speaking				1								
PH 245	Geology, Biology, Chemistry (AP transfer) || /General Physics PH 245					1							
Cultural Enrichment	Cultural Enrichment						1						
Cultural Enrichment	Cultural Enrichment							1					
Social Awareness	Social Awareness								1				
ENGL 311/321/325 || EN 249	Advanced Technical Writing								1				
Social Awareness	Social Awareness									1			
PHIL 216 / PL 205	Introduction to Ethics										1		
Proj 320	Project Management (Junior Status)											1	
BLAW 321 / BA 207	Contracts & Sales											1	

Has the program been significantly revised since the last review, and if so, how?

While this is our first review, the program has not undergone significant change. The first iteration of DMSE was an outline with general educations and a few classes taken from a number of different disciplines, mainly programming.

Are there any curricular or program changes currently in the review process? If so, what are they? Not Currently

Are there plans to revise the current program within the next three to five years? If so, what plans are envisioned and why? Not Currently

QUALITY OF INSTRUCTION

Discuss student and alumni perceptions of the quality of instruction.

The student and alumni consider the quality of instruction to be far above average as they compare themselves with the professional counterparts. Alumni survey data is located in Appendix A

Discuss advisory committee and employer perceptions of the quality of instruction.

The advisory committee and employer perceptions indicate that the DMSE program provides the students with the necessary skills to compete in a global market place. Advisory and Employer survey data is located in Appendix A.

What departmental and individual efforts have been made to improve the learning environment, add and use appropriate technology, train and increase the number of undergraduate and graduate assistants, etc.?

The DMSE department continually improves the learning environment by bringing the latest professional grade software, hardware, and tools for the students.

Summer Camp provides an excellent vehicle for DMSE Student volunteers to reflect on and teach what they have learned to other students.

Describe the types of professional development have faculty participated in, in efforts to enhance the learning environment (e.g. Writing across the Curriculum; Center for Teaching and Learning, etc.).

Faculty members have attended many seminars offered by the FCTL and also plan to regularly use atomic learning. In addition, learning opportunities provided by Ferris, webcasts, conferences, seminars, and other trainings are attended every year, in order for each instructor to remain relevant and connected to the software engineering industry.

What efforts have been made to increase the interaction of students with faculty and peers?

DMSE faculty members have a remarkably close relationship with their students. The faculty encourages these relationships but keep a professional demeanor throughout. The proximity of the office to the classrooms is close and students can drop in any time and look for us. Time is also allocated to attend events that are personal to the students as well as gaming with them.

Include such items as developmental activities, seminars, workshops, guest lectures, special events, and student participation in the Honors Program Symposium.

DMSE faculty are proactive in searching for and participating in developmental activities with the students, and provide leadership regarding attending professional events. These events include, but are not limited to:

- Game Developer's Conference in San Francisco
- Visual Studio Live
- Webcast Conferences
- AIMWest
- Corporate Visits

Discuss the extent to which current research and practice regarding inclusive pedagogy and curriculum infuse teaching and learning in this program.

The pedagogy we use at DMSE is driven by goals. Much like in the medieval 'Guild' system, the students are the apprentices. They are provided instruction and inspiration, but are then driven to practical application by assignments that are designed to achieve the learning outcomes. Students work together in a 'workshop' setting helping each other while class is in session and during open 'lab' time. We are available during lab hours and additional hours we spend in the labs working out student problems. The faculty members are the master craftsmen and the students are the apprentices working on

solutions to the problems that we set before us. They work together discovering solutions to assignments and on their own when driven to personal excellence.

Interesting articles on pedagogy strategy can be found here.

http://www.aacu.org/inclusive_excellence/documents/Williams_et_al.pdf

Making Excellence Inclusive: Preparing Students and Campuses for an Era of Greater Expectations By Damon A. Williams, Joseph B. Berger, and Shederick A. McClendon

http://www.curriculum.edu.au/verve/_resources/pedagogy_strategy_file.pdf

Pedagogy Strategy : learning in an online world. ISBN 192086530 6.

1. Education - Computer network resources. 2. Internet in education - Australia. 3. Learning strategies. I. Ministerial Council on Education, Employment, Training and Youth Affairs (Australia and New Zealand).371.334

What effects have actions described in (5) and (6) had on the quality of teaching and learning in the program?

When reviewing the quality of work that has evolved over the last 4 years of DMSE, there has been a significant improvement of work as evidenced by the submitted projects of the students.

Process drives product and the processes used incorporate traditional engineering best practices, methodologies, innovation, and emerging technologies which have inspired students to become engineers of excellence.

COMPOSITION AND QUALITY OF FACULTY

Describe and assess the composition of the faculty teaching courses in the program.

Each faculty member of the DMSE program has 20+ years of industry experience. The benefit to the University is credibility, experience and professional focus. The DMSE faculty is focused on real-world practicality with a vision of future innovation, which helps the students appreciate the importance of foresight and continued learning.

List the names of all tenured and tenure-track faculty by rank.

No tenured or tenure-track faculty.

Identify their rank and qualifications.

No tenured or tenure-track faculty.

Full Time Temporary Faculty – one year

Frederick Baker, Program Coordinator. BS, Electrical Engineering, GVSU, MS Software Engineering, GVSU

Indicate the number of promotions or merit awards received by program faculty since the last program review.

There have not been any promotions of merit awards received by the program faculty.

Summarize the professional activities of program faculty since inception or the last program review (attendance at professional meetings, poster or platform presentations, responsibilities in professional organizations, etc.).

Professional activities (Most Recent):

AIMWest

Microsoft Development Partner

Intel Parallel Processing Partner

Workload

What is the normal, annualized teaching load in the program or department? Indicate the basis of what determines a "normal" load. On a semester-by-semester basis, how many faculty have accepted an overload assignment?

Faculty load is four 3 credit classes a semester. In addition, all 'on the ground' faculty is assigned an advising load of about 10 new students a year. Maintain 20 'office hours' per week that are reasonably accommodating to students. Faculty performs the duties of Internship Coordinator, Student Activities Coordinator, and Event Coordinator.

Faculty accepts overload assignment as needed by the department. It is not untypical to have 4 'preps' per semester and overload is not encouraged as to maintain quality in the classroom.

List the activities for which faculty receive release time.

- professional development conferences

Additionally, the program coordinator is released 50% to coordinate departmental issues. These include:

- Recruitment
- Interviewing potential students
- Outreach into the community with 10+ school visits a year
- Market the program across the state
- Coordinate and troubleshooting all activities
- Organize and maintain DMSE summer camp
- Maintain 20 office hours during summer semester
- An additional release of one class per semester is provided during Academic Program Review.

What is the normal recruiting process for new faculty?

We approach professionals that we have close associations and classified advertisements.

What qualifications (academic and experiential) are typically required for new faculty?

Software Engineering Graduate degree and 15 or more years of industry experience.

What are the program's diversity goals for both gender and race/ethnicity in the faculty?

DMSE strives to be neutral in both gender and ethnicity in the outreach.

The DMSE program recruits in all areas of the Grand Rapids, and greater Grand Rapids area.

Describe and assess the efforts being made to attain goals in (c).

No specific plan to achieve REG equality.

Describe and assess the orientation process for new faculty.

Orientation for new faculty consists of the three day advising seminar held in Big Rapids along with mentoring by the entire faculty in DMSE. New faculty is introduced to all the classes that are taught in the program and depth of courseware coverage.

Reward Structure: e.g., salary, professional development funds, travel funds, UCEL and FSUGR incentive money

We have competitive salary structure that attracts and retains talented individuals.

Faculty are encouraged to attend professional development opportunities and are supported with reasonable travel budgets. Faculty members are encouraged to attend the Game Developer's conference in San Francisco to remain current in the video gaming industry and on alternate years VS LIVE for professional development conferences. Timme travel Grants are pursued for funding.

Describe the reward structure in the program/department/college as it relates to program faculty. Indicate the type of reward and eligibility criteria.

There is no specific reward structure for DMSE faculty other than what is offered through the general college sources.

Does the existing salary structure have an impact on the program's ability to recruit and retain quality faculty?

The salary structure seems to be unreasonable to both attract faculty and retain faculty.

Is the reward structure currently in place adequate to support faculty productivity in teaching, research, and service? If not, what recommendations would you make to correct the situation.

The reward system is apparently able to sustain faculty productivity. However, it would be helpful to develop a reward system to encourage more curricular development and higher end professional development. Currently, there is no support for a DMSE faculty to obtain tuition reimbursement for a Doctoral program.

Is enhancing diversity and inclusion a component of the reward structure? Please explain.

Enhancing the diversity and inclusion is a very important component. Unfortunately, it has not been discussed regarding the reward structure.

Graduate Instruction (if applicable)

List all faculty teaching graduate courses.

To date the DMSE program is an undergraduate degree

What percentage of graduate courses is taught by non-tenure-track faculty? Please comment.

To date the DMSE program is an undergraduate degree

What are the program's (or department's) criteria for graduate faculty?

To date the DMSE program is an undergraduate degree

Have all graduate faculty (including non-tenure-track faculty) met the criteria? Please comment.

To date the DMSE program is an undergraduate degree

Non-Tenure-Track and Adjunct Faculty.

Please provide a list for the last academic year of full-time non-tenure-track and adjunct faculty who taught courses in the program. For full-time non-tenure track faculty, indicate the length of their appointments and the number of years of service at the University. Comment on the program's ability to retain non-tenure-track faculty.

Full Time Non-Tenure Track Faculty

Frederick Baker – one year term contract – with DMSE since 2007

What percentage of program courses is taught by the faculty in (a)? What courses are they teaching? Please comment.

DMSE fulltime faculty teach 100 percent of the course work.

Describe the required qualifications (academic and experiential) for faculty listed in (a). Indicate if all faculty have met the criteria, and if not, what is being done to resolve the situation?

- All full time faculty must have a Graduate degree.
- All full time faculty must have 15 or more years in the software industry as a designer and programmer.
- All faculty must have a passion for teaching and serving students.

Does the program consider the current use of non-tenure-track faculty to be appropriate? Why or why not?

The program is staffed with non-tenured and non-tenured track faculty. The current faculty is serving the program well.

***If the program is accredited, what position if any does the accrediting body have regarding the use of non-tenured and adjunct faculty*?**

The program is not accredited; however, DMSE has a 5 year plan of achieving ABET accreditation.

SERVICE TO NON-MAJORS

Describe and assess the impact that delivery of service courses offered by the program or the department has on the program.

DMSE offers SENG 100 and SENG 101 to DAGD students. This has allowed DAGD students the opportunity to experience application development, and in some instances have prompted them to select software engineering as a career.

Identify and describe the General Education service courses provided by the program faculty for other departments at FSU.

Not offered to any general education courses for FSU

Identify and describe any non-General Education service courses or courses required for other programs. Comment on your interaction with the departments or programs for which the courses are provided.

The DMSE faculty works closely with the DAGD Program Coordinator to ensure that the course material in the DAGD program adequately prepares the DMSE students for employment in that gaming industry.

Students wishing to focus on game design inside of the SENG degree must take the following courses.

- DAGD 150 – Introduction to Game Design and Development
- DAGD 260 – Multimedia Authoring I
- DAGD 300 – Game Design and Theory
- DAGD 320 – Multiplayer Game Programming

Discuss the impact of the provision of General Education and non-General Education courses has on the program.

General education courses are a traditional part of a University education. The exercise of liberal arts in addition to career focus is a time honored approach to learning. DMSE honors this approach by requiring at least 41 credit hours of liberal arts be addressed. We agree that a rounded understanding of the world will enhance all facets in software engineering.

Certain General Education classes provide the DMSE student with a greater career advantage than other classes. These classes are strongly suggested by the student's advisor.

Does the program plan to increase, decrease, or keep constant its level of service courses? Explain.

There has not been any discussion regarding increasing or decreasing the programs level of service.

DEGREE PROGRAM COST AND PRODUCTIVITY DATA.

Submit Institutional Research and Testing data. Comment on the data.

Student Credit hours and the SCH/FTEF have increased 56% from 2008-2011. Based upon the increase in scheduled office appointments from interested students, it appears this strong growth trend will continue, as more students become aware of the program.

ASSESSMENT AND EVALUATION

Describe and evaluate the program's assessment mechanisms.

The DMSE program applies various forms of assessment as relevant to the course type.

There are two DMSE course types.

The first course type is a hands-on programming model to Software Engineering, requiring the students to complete a minimum of three software project. Two of the projects are completed in a team structure that encourages collaboration and compromise. The third project is an individual project that promotes autonomy and personal time and project management.

The second course type is an approach that focuses on the management of Software Engineering. This model incorporates intensive writing and documenting skills, while the students complete two programming projects. In this model the students complete one project as a team and one project individually.

The assessments that are used in these courses are

Each course type includes extensive text book reading and current event articles. Text book homework assignments are given weekly. Exams, quizzes, are offered on average every 3 to 4 weeks, depending on course workload.

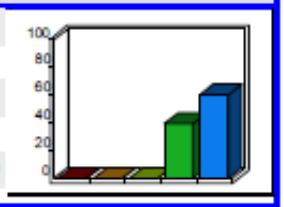
Each project is a concept to completion assignment and are graded based on a project rubric template, and are assessed with emphasis placed on completeness, quality, and innovation.

Additional Assessments: Every DMSE student must complete a professional internship in a Software Engineering department, a Microsoft Certification Exam, and a Capstone project. Each of these experiences is used to accurately profile a student's growth and level of competency in three major areas: Professionally, Personally, and Academically.

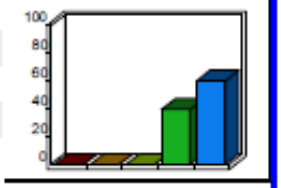
Sample Student Assessments

BAKER…2010-12-22…10F…SENG 100 AGA…SAI

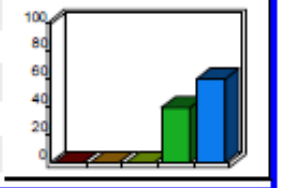
Q1 Expectations for graded assignments were clearly communicated — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

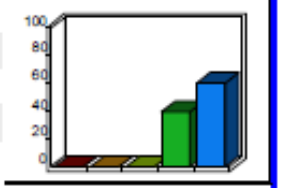
Q2 Course activities (lectures, projects, etc.) helped me learn the course material — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

Q3 Examinations, papers and other graded projects were returned in a reasonable amount of time — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

Q4 The course was well organized — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

Course Mapping

Program - Digital Media Software Engineering (B.A.S.) - Curriculum Map

Legend: (A) - Program Assessment, (I) - Introduced, (M) - Mastery, (R) - Reinforced

Outcomes	APS 301	BLAW 321	COMM	DAGD 15	DAGD 25	DAGD 30	DAGD 35	ECON 22	ENGL 150	ENGL 250	ENGL 311	ENGL 325	MATH 22	MATH 23	MATH 32	MFGE 341	MFGE 423	MFGE 442	PHIL 216	PHYS 241	PHYS 242	SENG 10	SENG 10	SENG 10	SENG 16	SENG 17	SENG 21	SENG 30	SENG 30	SENG 30	SENG 31	SENG 35	SENG 35	SENG 40	SENG 42	SENG 43	SENG 49	SENG 499
Development of professional grade skills in Software Application Development					I		R															I	I	R				R	R		R				R	M, R	M, R
Development of professional grade skills in Software Project Management	R																					I	I	R	I	R	R	M	I	R	M	R	R	M	M	I	R	M
Development of professional grade skills in Software Development Processes																									I	R	R			R			R	M, R		M, R	M, R

Outcomes	APS 301	BLAW 321	COMM	DAGD 15	DAGD 25	DAGD 30	DAGD 35	ECON 22	ENGL 150	ENGL 250	ENGL 311	ENGL 325	MATH 22	MATH 23	MATH 32	MFGE 341	MFGE 423	MFGE 442	PHIL 216	PHYS 241	PHYS 242	SENG 10	SENG 10	SENG 10	SENG 16	SENG 17	SENG 21	SENG 30	SENG 30	SENG 30	SENG 31	SENG 35	SENG 35	SENG 40	SENG 42	SENG 43	SENG 49	SENG 499
Development of professional grade skills in Software Development Methodologies																									I	I, R	I, R			I, R							M, R
Development of professional grade skills in Software Quality Assurance																									I					I, M						R	R
Development of professional grade skills in Software Analysis and Design																								I							I		M, R				M, R
Development of Industry Readiness																																I	I	I, M	I	M, R	R

See Appendix C – Course Syllabi – each includes defined course **outline, outcomes, and assessments**

See Appendix E – TracDat Data for Program and Course **Assessment mechanisms**

List and describe what variables are tracked and why when assessing the effectiveness of the program (e.g. mastery of essentials of subject area, graduation rates, employment rates, pass rates on professional exams).

The variables that are tracked in the program are quality, application of standards, project and exam scores, internship evaluations, and certification exams.

Quality has transitioned from a want to a need in the software industry. Apple and Google are two of the most valued companies in the world because of the quality products they have created. Students must understand and infuse quality in their design.

Standards are essential in allowing engineers the ability to define and maintain quality. Students must have demonstrated mastery of this concept to consider themselves professionals.

Projects are necessary from a team and individual focus because projects prepare the students for the inevitability of creating products. To have the ability to start with a concept and follow a software methodology to completion is mandatory multiple times. The students must make it a habit of completing a project with the same enthusiasm and thoroughness as when they started.

Exams and quizzes provide insight for the student into the areas that may require additional study for mastery. Exams are "labeled" in the program as "opportunities" not exams or assessments. When labeled as opportunities the students understand that the questions given are simply to expose weak areas necessary to strengthen.

Professional Certification Exams are essential for two main reasons. The first is the amount of study and preparation for the Certification Exam forces the student to strengthen weak areas and build confidence in areas that they have obtained mastery. Secondly, it allows them to compare themselves to a professional measuring tool. The students that have completed the tests have all conveyed the value of the tests in terms of refining what they "thought" they understood with what the "assumed" they understood.

Provide trend data for the variables listed in (1). Compare the data to accreditation benchmark standards if applicable, or provide some other type of assessment of the data.

Trend data indicates that as the program has grown so have the scores on projects and exams. There is not significant information to recognize a trend regarding certification exams.

Describe how the trend data in (2) is used to assess the rigor, breadth, and currency of the degree requirements and curriculum.

Each course is compared to a previous offering and adjustments are decided based upon student understanding, SAIs, exams and, feedback from the Advisory Board.

Describe how the trend data in (2) is used to assess the extent to which program goals are being met.

The leading indicators which indicate that the program goals are being met are best illustrated by the responses that we have received from the students and their employers.

Student employers regularly enquire about the program due to the level of expertise our students are displaying within their organization. Two graduates were promoted to leadership roles within 3 months of employment.

Several companies have asked to partner with our department and for us to contact them with future graduates. Students that have completed interviews are routinely invited for successive interviews and usually hired after extensive formal testing.

Students have performed beyond expectations at their internships and have been offered full-time positions 6 months to a year prior to their graduation.

The program goal is to produce graduates that have been introduced to the software industry's best practices, tools, and methodologies, and have gained competence and confidence in employing them. To provide the industry with graduates that upon entering the workforce, can easily be classified as a Software Engineer Level II.

ADMINISTRATION EFFECTIVENESS

Discuss the adequacy of administrative and clerical support for the program.

Administrative support for DMSE is excellent. We work together as a team helping students find answers to questions including classes, financial aid and general advising.

The front desk receptionists schedule time on faculty calendars for advising. They are able to answer student questions effectively and politely. They are timely and make sure everything can run smoothly as possible.

Administrative guidance is provided by Tracy Powers, Jocelyn Goheen, and Nancy Moore. As a team they provide solutions for main campus issues, scheduling classes, faculty contracts and budget demands.

Our registrar, Amanda Mitchell successfully manages simple to complex issues regarding student needs efficiently and without complaint. She is accurate and a significant asset to the organization.

Student services, is managed by Dawn Schavey. She is able to help students find housing and balance lifestyle issues. She volunteers with the community and helps out at all DMSE student functions and efficiently manages events issues such as food and rooms.

Enrollment is also managed efficiently and effectively. The administration team has worked diligently to improve the efficiency and effectiveness of the admission process. We are fortunate to have a strong and balanced marketing department headed by Jennifer Amlotte and supported by graphic designer Mary Dilley.

Are the program and/or department run in an efficient manner? Please explain.

The department is run efficiently and with a minimum. Office hours are convenient for students. Faculty is proactive about book lists and supplies. Administration is effective in responding to needs in scheduling classrooms. Changes to curriculum and courseware and completed in a timely manner.

Are class and teaching schedules effectively and efficiently prepared? Please comment.

Yes, there have been no scheduling conflicts. This is due to the assignment of 90%+ of DMSE classes being conducted in the same room.

Are students able to take the courses they need in a timely manner? Please comment.

Students are able to take courses in a timely manner. There is a balanced course sequence that allows students to work through the program efficiently and effectively. Courses are offered in the fall, spring, and summer.

Section 4: Facilities and equipment

INSTRUCTIONAL ENVIRONMENT

Are current classrooms, labs, and technology (both on-campus and at off-site locations) adequate? Explain.

The technology in the DMSE classroom is excellent and exceeds the current need. Software is also kept current through yearly subscriptions.

How does the condition of current facilities impact program delivery? Explain.

The current facility is physically tight, but is adequate to deliver reasonable instruction. Software has been installed that allows instructors can use to communicate with each student computer, lock out the internet or even blank all screens if attention is needed.
The necessary professional grade software tools have been provided for the students to use freely.

Describe the program's projected needs with respect to instructional facilities.

Classrooms: There will be a need for a "Studio" room within two years that allows the students to have a project room to innovate and develop their ideas in an "ownership" environment.

Describe current plans for facilities improvements and indicate their status.

No other plans for facility improvement.

COMPUTER ACCESS AND AVAILABILITY

Outside of computers in faculty and staff offices, identify the computing resources (hardware and software) that are allocated to the program.

The computers that are used have Intel processors that have 8 core processors, high-end NVidia graphic processors, and dual 19 inch monitors. The current equipment surpasses the hardware requirements that the software requires.

The software that is used regularly and maintained in the DMSE program is as follows: Visual Studio 2010 professional, Microsoft Visio, Unity 3D, UDK, CryEngine2, Microsoft Word, and Kodu.

Discuss the adequacy of these resources and identify needed additional resources.

Currently, these resources are adequate, but to maintain an industry and creative advantage the program must include emerging technologies, such as, Wi-Fi, Gesture based equipment, and microcontrollers.

Does an acquisition plan to address these needs currently exist?

Yes

Describe the plan.

An acquisition for the necessary hardware and software was supplied to Don Green and Tracy Powers.

Has it been included in the department or college's planning documents?

This is unknown at the moment.

Discuss the efficacy of online services available to the program.

Ferris Connect is used exclusively for all course work, and has been invaluable effectively delivering homework, projects, quizzes, exams, and reference material. The students are all technically savvy and have appreciated having all course work online. (This will continue as FSU transitions to the Blackboard 9.1 / Ferris Learn system.)

Discuss the adequacy of computer support, including the support for on-line instruction if applicable.

The on-site computer IT person maintains the computers as required. Students have not experienced any difficulty utilizing the lab computers of Ferris Connect.

Computers and Equipment

The DMSE program receives the used computers from the DAGD program every 4 years. Fortunately, the computers that are received are high-end computers that exceed the current demands of the software used in the DMSE program.

The classroom has a video projector, which is used during every class. One projector has been replaced and will need to be replaced in 2 years, and the projector in 4 years.

Discuss the impact of adequacy of other types of instructional technology resources and support of these resources on the program.

Students can access online instruction from Microsoft, Code Project, KhanAcademy.org, TechEd and many other top-tier Software Engineering websites. We also have a library of Software Engineering development videos the students may "check-out".

The impact of these instructional resources cannot be understated. Each resource can substitute many hours of frustration and confusion with hours of knowledge and understanding.

OTHER INSTRUCTIONAL TECHNOLOGY

Identify other types of instructional technology resources that are allocated or available to the program.

GRCC can provide AV carts with monitors and players on them. Additionally, if classroom or seminar space is needed, the building coordinator is always available to help.

All faculty are supplied "top-tier" Dell Laptop computers that are capable of running the software that is appropriate for the instructor.

Discuss how these resources are used.

These resources are part of the everyday work at Ferris GR.

Discuss the adequacy of these resources and identify needed additional resources.

These resources are adequate.

However, additional resources are needed for future software development: iPod Touch, iPad, Android Tablets, Windows Phone 7, Windows 8 Tablet, ARDrone 2.0, Xbox 360, PS3, and the Kinect for Windows. Students that have written applications on these platforms will significantly increase their skill set and their industry marketability.

Does an acquisition plan to address these needs currently exist? Describe the plan. Has it been included in the department or college's planning documents?

A request for the purchase of each item was submitted to the college for review and possible acquisition. The plan is to have the equipment as soon as possible, in order for the students to begin immediate development

Discuss the impact of adequacy of other types of instructional technology resources and support of these resources on the program.

Unfortunately the engineering students do not currently have a "Collaborative" room/area in which they are able to work on their projects as a team.

A **Software Development Collaboratory** (SDC) would allow students to take part in ongoing software development projects, for internal or external clients. It would allow them to take theory taught in the classroom and unite it with professional practice through intensive laboratory experience.

All software engineering students should complete 5 courses in the lab. In the SDC, student teams work independently, while the instructors act as coaches and managers.

Within the SDC, teams:

- Work within a defined software process.
- Assign roles within the team.
- Plan the work.
- Execute their plans.
- Report team results.
- Actively contribute to process improvement.

Lab Facilities

Ideally the SDC will be located in the ATC building. Each team would have their own work space, as well as shared meeting and presentation areas, for team meetings and project development activities.

The SDC would allow students the opportunity to join a team working on small/medium-scale, industry-sponsored projects, within an engaging environment, that closely models the corporate climate.

LIBRARY RESOURCES

Discuss the adequacy of the print and electronic and other resources available through FLITE for the program.

FLITE is a remarkable tool for Ferris Students. Its online resources are outstanding. However, DMSE students do not use FLITE for either online resource or physical resource. It is a matter of knowing what is available and DMSE students do not know how to access the FLITE system.

Discuss the service and instruction availability provided by the Library faculty and staff with respect to the needs of the program.

We have not requested any help from Library resources.

Discuss the impact of the budget allocation provided by FLITE to your program. Is the budget allocation adequate? Explain.

We are not aware of any budget resources available through FLITE. If FLITE would be willing to allocate financial resources, we would be happy to utilize them by the purchase of Lynda.com video tutorials.

http://www.lynda.com/

Section 5: Conclusions

Based on data analysis derived from Sections 2-4 and on the collective wisdom and judgment of the PRP. In arriving at these conclusions, the PRP should summarize the relationship of the program to each of following specific categories and any other categories it deems appropriate:

A. RELATIONSHIP TO FSU MISSION
B. PROGRAM VISIBILITY AND DISTINCTIVENESS
C. PROGRAM VALUE
D. ENROLLMENT
E. CHARACTERISTICS, QUALITY AND EMPLOYABILITY OF STUDENTS
F. QUALITY OF CURRICULUM AND INSTRUCTION
G. COMPOSITION AND QUALITY OF THE FACULTY

While there are many computer related programs throughout Michigan, the DMSE program at FSU is distinct in providing extensive formal and hands-on training in software engineering. DMSE utilizes the industries best practices, tools, and methodologies, while introducing emerging technologies.

This enables our graduates to become productive immediately. This philosophy pervades all of our courses and provides our students with a distinct competitive advantage, particularly during a prolonged recession. Many graduates from competing universities need to undergo several months of extensive retraining at company expense before they can be useful to their employers. DMSE graduates enter the workforce competent, confident, and can immediately contribute to the success of a company.

Instructors work with their students to develop projects for non-profit organizations in the community, thus engendering good relations with the community while providing realistic exposure to business challenges.

There had been a period of enrollment decline which reflected a wider national drop in the number of computer majors, possibly because of a distorted view of the role of outsourcing and misperceptions of the job market for new graduates. In any case, the enrollment for the DMSE program has steadily increased.

DMSE students and graduates are highly sought after by industry and receive well-paying entry jobs which are often followed by multiple lucrative and early promotions.

In conclusion, the DMSE program provides the formal training and encourages students to aspire to become the next Bill Gates, Steve Jobs, or Mark Zuckerburg. It is this inspiration that will fuel them to succeed as a corporate or entrepreneurial engineer. In either instance they are prepared to change the world and improve the quality of life for people globally.

The vision of the DMSE program fully embraces the original mission of Ferris:

> The primary task of the school has been that of developing character; inspiring young people with a desire to do something, and to be something.

~W.N. Ferris (1895)

Appendix A – Survey Results

DMSE APR...Alumni

Frequencies

Prepared by: Institutional Research & Testing, 06/12

Statistics

	N		Mean	Median	Std. Deviation
	Valid	Missing			
q1.a Overall quality of your experience	2	0	3.50	3.50	.707
q1.b Overall quality of the instructors	2	0	3.50	3.50	.707
q1.c Overall quality of the software	2	0	3.50	3.50	.707
q1.d Overall relevance of the software	2	0	3.50	3.50	.707
q1.e Overall quality of the hardware and facility	2	0	3.00	3.00	.000
q1.f Overall relevance of the hardware (computers) and facility	2	0	3.50	3.50	.707
q1.g Overall quality of the curriculum	2	0	3.50	3.50	.707
q1.h Overall relevance of the curriculum	2	0	3.50	3.50	.707
q2 Elaborate on any of the above	2	0			
q3.a SENG 100/101 Computer Programming 1	1	1	5.00	5.00	
q3.b SENG 102 Computer Programming 2 (SENG 101)	1	1	5.00	5.00	

q3.c SENG 300 Software Data Structures (SENG 102)	2	0	4.50	4.50	.707
q3.d SENG 301 Programming Languages (SENG 300)	2	0	4.50	4.50	.707
q3.e SENG 315 Software Component Design	2	0	4.50	4.50	.707
q3.f SENG 400 Engineering Enterprise Software Applications (SENG 350)	2	0	4.50	4.50	.707
q3.g SENG 430 Programming Graphical User Interfaces (SENG 350)	2	0	5.00	5.00	.000
q3.h SENG 160 Software Engineering Methodologies and Processes	2	0	4.50	4.50	.707
q3.i SENG 170 Software Requirements Management (SENG 160)	2	0	4.50	4.50	.707
q3.j SENG 210 Software Configuration Management (SENG 160)	2	0	4.50	4.50	.707
q3.k SENG 302 Software Quality Assurance (SENG 160)	2	0	4.50	4.50	.707
q3.l SENG 350 Software Design and Architecture (SENG 160, SENG 300)	2	0	4.50	4.50	.707
q3.m SENG 355 Software Engineering Tools (SENG 160)	2	0	4.50	4.50	.707
q3.n SENG 420 Software Development Industry Certification (SENG 350, SENG 355)	2	0	4.50	4.50	.707
q3.o SENG 491 Applied Internship (SENG 301, SENG 302, SENG 315)	2	0	4.00	4.00	1.414
q3.p SENG 499 Capstone in SENG (SENG 491)	2	0	5.00	5.00	.000
q4 How long to be employed w/ animation & Software Engineering field	2	0	2.00	2.00	.000
q5 Current salary	2	0	5.50	5.50	.707
q6 Able to get part time work in animation, Software Eng or media field w/in yr	2	0	1.00	1.00	.000

q7 Still employed in field	2	0	1.00	1.00	.000
q8 Where working	2	0			
q9 Gone for additional training	2	0	2.00	2.00	.000
q10 Where	2	0			
q11 Strongest features	2	0			
q12 Areas that require most improvement	2	0			
q13 Additional comments	2	0			

Frequency Table

q1.a Overall quality of your experience

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q1.b Overall quality of the instructors

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q1.c Overall quality of the software

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q1.d Overall relevance of the software

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q1.e Overall quality of the hardware and facility

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	2	100.0	100.0	100.0

q1.f Overall relevance of the hardware (computers) and facility

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q1.g Overall quality of the curriculum

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q1.h Overall relevance of the curriculum

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q2 Elaborate on any of the above

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		2	100.0	100.0	100.0

q3.a SENG 100/101 Computer Programming 1

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Satisfied	1	50.0	100.0	100.0
Missing	N/A	1	50.0		

q3.a SENG 100/101 Computer Programming 1

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Satisfied	1	50.0	100.0	100.0
Missing	N/A	1	50.0		
Total		2	100.0		

q3.b SENG 102 Computer Programming 2 (SENG 101)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Satisfied	1	50.0	100.0	100.0
Missing	N/A	1	50.0		
Total		2	100.0		

q3.c SENG 300 Software Data Structures (SENG 102)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.d SENG 301 Programming Languages (SENG 300)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.e SENG 315 Software Component Design

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.f SENG 400 Engineering Enterprise Software Applications (SENG 350)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.g SENG 430 Programming Graphical User Interfaces (SENG 350)

	Frequency	Percent	Valid Percent	Cumulative Percent

q3.g SENG 430 Programming Graphical User Interfaces (SENG 350)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Satisfied	2	100.0	100.0	100.0

q3.h SENG 160 Software Engineering Methodologies and Processes

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.i SENG 170 Software Requirements Management (SENG 160)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.j SENG 210 Software Configuration Management (SENG 160)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0

	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.k SENG 302 Software Quality Assurance (SENG 160)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.l SENG 350 Software Design and Architecture (SENG 160, SENG 300)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.m SENG 355 Software Engineering Tools (SENG 160)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.n SENG 420 Software Development Industry Certification (SENG 350, SENG 355)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.o SENG 491 Applied Internship (SENG 301, SENG 302, SENG 315)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Dissatisfied	1	50.0	50.0	50.0
	Very Satisfied	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q3.p SENG 499 Capstone in SENG (SENG 491)

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Satisfied	2	100.0	100.0	100.0

q4 How long to be employed w/ animation & Software Engineering field

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Within 6 months	2	100.0	100.0	100.0

q5 Current salary

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	$40,001 - $50,000	1	50.0	50.0	50.0
	$50,001 - $60,000	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q6 Able to get part time work in animation, Software Eng or media field w/in yr

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	2	100.0	100.0	100.0

q7 Still employed in field

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	2	100.0	100.0	100.0

q8 Where working

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Grand Rapids, MI	1	50.0	50.0	50.0
	South west mi	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q9 Gone for additional training

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	No	2	100.0	100.0	100.0

q10 Where

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		2	100.0	100.0	100.0

q11 Strongest features

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Real world assignments with tools actually being used in field.	1	50.0	50.0	50.0
	The DMSE program contained a wide variety of classes, covering the varying aspects of the industry in a true to real-life manor.	1	50.0	50.0	100.0

q11 Strongest features

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Real world assignments with tools actually being used in field.	1	50.0	50.0	50.0
	The DMSE program contained a wide variety of classes, covering the varying aspects of the industry in a true to real-life manor.	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q12 Areas that require most improvement

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	I felt that the program was too easy. I would have liked a more strict learning environment where more was expected of me.	1	50.0	50.0	50.0
	Its pretty broad category so you never know what a person will need so I not sure. I was satisfied.	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

q13 Additional comments

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	50.0	50.0	50.0
	Good quality program	1	50.0	50.0	100.0
	Total	2	100.0	100.0	

DMSE APR...Employer

Frequencies

Prepared by: Institutional Research & Testing, 06/12

Statistics

	N		Mean	Median
	Valid	Missing		
q1_1 I am a: Professional in the Animation industry	1	0	.00	.00
q1_2 I am a: Professional in the Software Engineering industry	1	0	1.00	1.00
q1_3 I am a: Professional in Video, Media or Advertising	1	0	.00	.00
q1_4 I am a: An employer or past employer of a DMSE alumni	1	0	.00	.00
q1_5 I am a: Educator	1	0	.00	.00
q1_6 I am a: Architecture	1	0	.00	.00
q1_7 I am a: Other	1	0	.00	.00
q1.a Other specified	1	0		
q2 Satisfaction with the DMSE graduate/intern	1	0	4.00	4.00
q3.a Foundation/Traditional Engineering	1	0	1.00	1.00
q3.b Project Management	1	0	3.00	3.00
q3.c Collaboration	1	0	4.00	4.00
q3.d UML	1	0	1.00	1.00
q3.e Database Design	1	0	4.00	4.00

q3.f Innovation	1	0	3.00	3.00
q3.g Professional Development	1	0	2.00	2.00
q3.h Multiple Programming Languages	1	0	2.00	2.00
q3.i Communication	1	0	3.00	3.00
q3.j General Education	1	0	3.00	3.00
q3.k Other	1	0	4.00	4.00
q3.l Other skills you believe important	1	0		
q4.a Foundation/Traditional Engineering	0	1		
q4.b Project Management	1	0	3.00	3.00
q4.c Collaboration	1	0	3.00	3.00
q4.d UML	0	1		
q4.e Database Design	1	0	2.00	2.00
q4.f Innovation	1	0	4.00	4.00
q4.g Professional Development	1	0	4.00	4.00
q4.h Multiple Programming Languages	1	0	4.00	4.00
q4.i Communication	1	0	4.00	4.00
q4.j General Education	1	0	4.00	4.00
q4.k Other	1	0	3.00	3.00
q4.l Other skills you believe important	1	0		
q5.a Visual Studio 2010 Ultimate	1	0	4.00	4.00
q5.b Unity 3D	1	0	2.00	2.00
q5.c Visio	1	0	2.00	2.00
q5.d Unreal	1	0	2.00	2.00
q5.e MS Project	1	0	2.00	2.00
q5.f CryEngine	1	0	2.00	2.00
q5.g Eclipse	1	0	2.00	2.00

q5.h Other	1	0	1.00	1.00
q5.i Other specified	1	0		
q6.a Visual Studio 2010 Ultimate	1	0	2.00	2.00
q6.b Unity 3D	1	0	1.00	1.00
q6.c Visio	1	0	1.00	1.00
q6.d Unreal	1	0	1.00	1.00
q6.e MS Project	1	0	1.00	1.00
q6.f CryEngine	0	1		
q6.g Eclipse	1	0	1.00	1.00
q6.h Other	1	0	1.00	1.00
q6.i Other specified	1	0		
q7 Additional comments about software	1	0		
q8 Specific skills stand out as excellent	1	0		
q9 Specific skills stand out as poor	1	0		
q10 Satisfaction w/ overall quality of program	1	0	5.00	5.00
q11 Expound on your answers to above	1	0		
q12 Strongest features of program	1	0		
q13 Areas that require most improvement	1	0		
q14 Additional comments	1	0		

Frequency Table

q1_1 I am a: Professional in the Animation industry

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	1	100.0	100.0	100.0

q1_2 I am a: Professional in the Software Engineering industry

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Selected	1	100.0	100.0	100.0

q1_3 I am a: Professional in Video, Media or Advertising

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	1	100.0	100.0	100.0

q1_4 I am a: An employer or past employer of a DMSE alumni

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	1	100.0	100.0	100.0

q1_5 I am a: Educator

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	1	100.0	100.0	100.0

q1_6 I am a: Architecture

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	1	100.0	100.0	100.0

q1_7 I am a: Other

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	1	100.0	100.0	100.0

q1.a Other specified

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	100.0	100.0	100.0

q2 Satisfaction with the DMSE graduate/intern

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Satisfied	1	100.0	100.0	100.0

q3.a Foundation/Traditional Engineering

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unimportant	1	100.0	100.0	100.0

q3.b Project Management

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	1	100.0	100.0	100.0

q3.c Collaboration

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Important	1	100.0	100.0	100.0

q3.d UML

	Frequency	Percent	Valid Percent	Cumulative Percent

q3.d UML

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unimportant	1	100.0	100.0	100.0

q3.e Database Design

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Important	1	100.0	100.0	100.0

q3.f Innovation

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	1	100.0	100.0	100.0

q3.g Professional Development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Unimportant	1	100.0	100.0	100.0

q3.h Multiple Programming Languages

		Frequency	Percent	Valid Percent	Cumulative Percent

q3.h Multiple Programming Languages

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Unimportant	1	100.0	100.0	100.0

q3.i Communication

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	1	100.0	100.0	100.0

q3.j General Education

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	1	100.0	100.0	100.0

q3.k Other

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Important	1	100.0	100.0	100.0

q3.l Other skills you believe important

	Frequency	Percent	Valid Percent	Cumulative Percent

q3.l Other skills you believe important

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Some exposure to real world development tools, languages and database use. Its very helpful if someone already knows the basics of visual studio, sql management studio etc.	1	100.0	100.0	100.0

q4.a Foundation/Traditional Engineering

		Frequency	Percent
Missing	NA/NO	1	100.0

q4.b Project Management

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Dissatisfied	1	100.0	100.0	100.0

q4.c Collaboration

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Dissatisfied	1	100.0	100.0	100.0

q4.d UML

		Frequency	Percent
Missing	NA/NO	1	100.0

q4.e Database Design

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Dissatisfied	1	100.0	100.0	100.0

q4.f Innovation

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	100.0	100.0	100.0

q4.g Professional Development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	100.0	100.0	100.0

q4.h Multiple Programming Languages

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	100.0	100.0	100.0

q4.i Communication

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	100.0	100.0	100.0

q4.j General Education

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Satisfied	1	100.0	100.0	100.0

q4.k Other

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Dissatisfied	1	100.0	100.0	100.0

q4.l Other skills you believe important

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	current software development tools and practices	1	100.0	100.0	100.0

q5.a Visual Studio 2010 Ultimate

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Relevant	1	100.0	100.0	100.0

q5.b Unity 3D

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	100.0	100.0	100.0

q5.c Visio

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	100.0	100.0	100.0

q5.d Unreal

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	100.0	100.0	100.0

q5.e MS Project

		Frequency	Percent	Valid Percent	Cumulative Percent

q5.e MS Project

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	100.0	100.0	100.0

q5.f CryEngine

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	100.0	100.0	100.0

q5.g Eclipse

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	100.0	100.0	100.0

q5.h Other

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unknown	1	100.0	100.0	100.0

q5.i Other specified

	Frequency	Percent	Valid Percent	Cumulative Percent

q5.i Other specified

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	100.0	100.0	100.0

q6.a Visual Studio 2010 Ultimate

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Low Quality	1	100.0	100.0	100.0

q6.b Unity 3D

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unknown	1	100.0	100.0	100.0

q6.c Visio

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unknown	1	100.0	100.0	100.0

q6.d Unreal

	Frequency	Percent	Valid Percent	Cumulative Percent

q6.d Unreal

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unknown	1	100.0	100.0	100.0

q6.e MS Project

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unknown	1	100.0	100.0	100.0

q6.f CryEngine

		Frequency	Percent
Missing	System	1	100.0

q6.g Eclipse

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unknown	1	100.0	100.0	100.0

q6.h Other

	Frequency	Percent	Valid Percent	Cumulative Percent

q6.h Other

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Unknown	1	100.0	100.0	100.0

q6.i Other specified

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	100.0	100.0	100.0

q7 Additional comments about software

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Would like to see sql mgt tools	1	100.0	100.0	100.0

q8 Specific skills stand out as excellent

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Able to learn quickly and handle similar work that experienced engineers do	1	100.0	100.0	100.0

q9 Specific skills stand out as poor

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	lack of database knowledge	1	100.0	100.0	100.0

q10 Satisfaction w/ overall quality of program

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Satisfied	1	100.0	100.0	100.0

q11 Expound on your answers to above

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	The students have been valuable to our company and I believe we have given them great value in training, and career opportunities	1	100.0	100.0	100.0

q12 Strongest features of program

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	100.0	100.0	100.0

q13 Areas that require most improvement

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Do more real world scenarios, and more common technologies used locally	1	100.0	100.0	100.0

q14 Additional comments

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	100.0	100.0	100.0

DMSE APR...Faculty

Frequencies

Prepared by: Institutional Research & Testing, 06/12

Statistics

	N	
	Valid	Missing
q1 Departmental Administrative Improvements	3	0
q2 3-5 year Departmental Vision	3	0
q3 Course modifications	3	0
q4 Viewpoints on accelerated and on-line classes	3	0
q5 Strategies and tactics to improve retention	3	0
q6 Strategies and tactics to improve cross class collaboration	3	0
q7 Are current classrooms, labs, and technology adequate	3	0
q8 Condition of current facilities impact program delivery	3	0
q9 Projected needs with respect to instructional facilities	3	0
q10 Identify computing resources allocated to program	3	0
q11 Adequacy of these resources/needed additional resources	3	0
q12 Does an acquisition plan to address these needs currently exist	3	0
q13 Describe the plan	3	0
q14 Included in the department/college's planning documents	3	0

q15 Discuss the efficacy of online services available	3	0
q16 Discuss the adequacy of computer support	3	0
q17 Discuss the computers/equipment available for instruction	3	0
q18 Impact of adequacy of other types of instructional technology/resources support	3	0
q19 Additional comments	3	0

Frequency Table

q1 Departmental Administrative Improvements

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	As the DMSE program moves from CPTS to the College of Education and Human Services, it will ideally have additional administrative resources to use for both on- and off-campus locations.	1	33.3	33.3	33.3
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	66.7
	You should let Rick teach. He's got a great connection to his students but has to split his work between being faculty and administration. Either get him a part time coordinator or download this to someone else. He's valuable, smart and effective as faculty.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q2 3-5 year Departmental Vision

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	A concentration on mobile applications will help students get jobs and start businesses. More work with the DAGD program and cross pollinization will help both programs get stronger.	1	33.3	33.3	33.3

	Continued migration into DMS trends and developments so as to not only stay current, but rather move forward with such changes. As such, there needs to be additional resources to help provide such forward transitions.	1	33.3	33.3	66.7
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q3 Course modifications

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	I don't know enough about the DMSE program to comment.	1	33.3	33.3	33.3
	More opportunity to work with DAGD students.	1	33.3	33.3	66.7
	Similarly, courses need to continually update as conditions in industry and software change. This also requires such needed resources as well.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q4 Viewpoints on accelerated and on-line classes

	Frequency	Percent	Valid Percent	Cumulative Percent

Valid	As the migration toward full online course and program offerings continue, I am not exactly of the belief that all courses in a given program should be in this format. I do appreciate the benefits from a travel and scheduling perspective for students, but I likewise think that human interaction in a real-time, same place format is needed to really understand all of the non-verbal communication and understanding of students and professors alike. Perhaps more mixed delivery courses are a good hybrid of this model, where there are some in-person classes mixed with predominantly online instruction.	1	33.3	33.3	33.3
	On-line is helpful, but only for certain motivated students.	1	33.3	33.3	66.7
	These can be used in a program if they are implemented and taught correctly. Not every course should be taught in an accelerated or on-line format.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q5 Strategies and tactics to improve retention

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	A better 4 year plan. Students cannot get to their diploma in 4 years as there are not enough teachers.	1	33.3	33.3	33.3
	Continued good communication with existing students helps retention, along with the ability to continue to attract desired recruits on the onset.	1	33.3	33.3	66.7
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	100.0

q5 Strategies and tactics to improve retention

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	A better 4 year plan. Students cannot get to their diploma in 4 years as there are not enough teachers.	1	33.3	33.3	33.3
	Continued good communication with existing students helps retention, along with the ability to continue to attract desired recruits on the onset.	1	33.3	33.3	66.7
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q6 Strategies and tactics to improve cross class collaboration

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Focusing on one or two main departmental projects can help focus collaboration.	1	33.3	33.3	33.3
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	66.7
	Migration into the College of Education will provide additional cross-class and possibly, cross-program collaboration on project and course levels.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q7 Are current classrooms, labs, and technology adequate

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	I don't know enough about the DMSE program to comment.	1	33.3	33.3	33.3
	The labs seem adequate from a computer POV, but the AV set ups are cobbled together at best. Look at Instructor stations at community colleges and you have good quality sound, video/computer switching and so forth. The Ferris lab AV in room 176 still uses desktop plug in speakers. Additionally, there are no game boxes (xbox. ps3 etc) that can be used for demo purposes.	1	33.3	33.3	66.7
	This aspect is dependent on what is available at each off-site location. GR and BR seem to be able to have such needed resources, although additional resources always are beneficial to students and the program in general.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q8 Condition of current facilities impact program delivery

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Again, the AV setup in the labs is not particularly good. All classrooms need game systems integrated into the AV. Additionally, the common space for students to gather is awkward and does not promote community.	1	33.3	33.3	33.3
	Based on available computing and software resources, the program could be limited in delivery, especially at new locations if such resources don't exist currently.	1	33.3	33.3	66.7

	I don't know enough about the DMSE program to comment.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q9 Projected needs with respect to instructional facilities

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	I don't know enough about the DMSE program to comment.	1	33.3	33.3	33.3
	Ideally, the program's growth will cause the need for additional instructional facilities, especially at other off-campus locations, such as at our WCCC location in Metro Detroit.	1	33.3	33.3	66.7
	It's adequate, but there is no real room to grow. Essential class space has been taken from the SENG/DAGD programs to start molecular diagnostics. Our programs are landlocked and we cannot create space from nothing.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q10 Identify computing resources allocated to program

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	33.3	33.3	33.3
	I am not aware of such pre-allocated resources, if they exist. If they do, I apologize for not knowing this previously.	1	33.3	33.3	66.7

	I don't know enough about the DMSE program to comment.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q11 Adequacy of these resources/needed additional resources

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	As the program is still small, the resources fit the need.	1	33.3	33.3	33.3
	I believe that there is a definite need for additional resources. Such resources will also aid in student recruitment and retention as well.	1	33.3	33.3	66.7
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q12 Does an acquisition plan to address these needs currently exist

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Any plans have been stalled from the dissolution of CPTS.	1	33.3	33.3	33.3
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	66.7
	Not exactly sure about any acquisition plans, but rather the move of DMSE into the College of Education and Human Services.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q13 Describe the plan

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	33.3	33.3	33.3
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	66.7
	Not aware of such planning details.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q14 Included in the department/college's planning documents

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	I don't know enough about the DMSE program to comment.	1	33.3	33.3	33.3
	Not aware of such planning details.	1	33.3	33.3	66.7
	Not that I am aware of.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q15 Discuss the efficacy of online services available

	Frequency	Percent	Valid Percent	Cumulative Percent

Valid	Based on the various "situations" that have occurred over the past few semesters casuing various outages or other problems with our FerrisConnect/WebCT system, I would say that our on-line support is not exactly the optimum, especially this past semester, which is our last main semester using this system that is not supported by Blackboard, the purchaser of WebCT. Quite honestly, I would rather see FSU use another course management system other than Blackboard - a company who has purchased many other packages over the years and forced users to take what they get in terms of software. I am not looking very forward to our migration and eventual transition to Blackboard 9.1.	1	33.3	33.3	33.3
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	66.7
	Migration to the new black board product will positively impact the program.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q16 Discuss the adequacy of computer support

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Based on the various "situations" that have occurred over the past few semesters casuing various outages or other problems with our FerrisConnect/WebCT system, I would say that our on-line support is not exactly the optimum, especially this past semester, which is our last main semester using this system that is not supported by Blackboard, the purchaser of WebCT.	1	33.3	33.3	33.3

	I don't know enough about the DMSE program to comment.	1	33.3	33.3	66.7
	Very good. Vel Pavlov and his staff are effective, proactive and polite.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q17 Discuss the computers/equipment available for instruction

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Good desktop systems equal to the students. Good quality up to date software.	1	33.3	33.3	33.3
	I don't know enough about the DMSE program to comment.	1	33.3	33.3	66.7
	Not specifically knowledgeable about the computers and equipment outside of GR available for instruction. This will impact future course/program delivery opportunities.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q18 Impact of adequacy of other types of instructional technology/resources support

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	I don't know enough about the DMSE program to comment.	1	33.3	33.3	33.3

	Quite honestly, I would rather see FSU use another course management system other than Blackboard - a company who has purchased many other packages over the years and forced users to take what they get in terms of software. I am not looking very forward to our migration and eventual transition to Blackboard 9.1.	1	33.3	33.3	66.7
	The AV is marginal.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q19 Additional comments

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		2	66.7	66.7	66.7
	Thankyou for requesting my input, I am always available for such activities and think that they only continue to add to the overall strength of the program.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

DMSE APR...Advisory Board

Frequencies

Prepared by: Institutional Research & Testing, 06/12

Statistics

	N		Mean	Median	Std. Deviation
	Valid	Missing			
q1_1 I am a: Professional in the Animation industry	3	0	.00	.00	.000
q1_2 I am a: Professional in the Software Engineering industry	3	0	.67	1.00	.577
q1_3 I am a: Professional in Video, Media or Advertising	3	0	.33	.00	.577
q1_4 I am a: An employer or past employer of a DMSE alumni	3	0	.00	.00	.000
q1_5 I am a: Educator	3	0	.33	.00	.577
q1_6 I am a: Architecture	3	0	.00	.00	.000
q1_7 I am a: Other	3	0	.00	.00	.000
q1.a Other specified	3	0			
q2.a Foundation/Traditional Engineering	3	0	3.33	3.00	.577
q2.b Project Management	3	0	3.00	3.00	.000
q2.c Collaboration	3	0	4.00	4.00	.000
q2.d UML	3	0	3.33	4.00	1.155

q2.e Database Design	3	0	3.67	4.00	.577
q2.f Innovation	3	0	3.33	3.00	.577
q2.g Professional Development	3	0	3.33	3.00	.577
q2.h Multiple Programming Languages	3	0	3.33	4.00	1.155
q2.i Communication	3	0	4.00	4.00	.000
q2.j General Education	3	0	3.00	3.00	.000
q2.k Other	1	2	3.00	3.00	
q3 Other important skill sets	3	0			
q4 Any specific skills stand out as excellent from our work/students	3	0			
q5.a Visual Studio 2010	3	0	4.00	4.00	.000
q5.b Unity 3D	2	1	3.00	3.00	1.414
q5.c Visio	3	0	3.00	3.00	1.000
q5.d Unreal Development Kit	2	1	2.50	2.50	.707
q5.e MS Project	3	0	3.00	3.00	1.000
q5.f CryEngine	2	1	2.50	2.50	.707
q5.g Android Development	3	0	3.00	3.00	1.000
q5.h iOS Development	2	1	3.00	3.00	1.414
q5.i Other	0	3			
q5.j Other specified	3	0			
q6.a Visual Studio 2010	3	0	4.00	4.00	.000
q6.b Unity 3D	3	0	3.67	4.00	.577
q6.c Visio	3	0	3.67	4.00	.577
q6.d Unreal Development Kit	3	0	3.67	4.00	.577
q6.e MS Project	3	0	3.67	4.00	.577
q6.f CryEngine	3	0	3.67	4.00	.577
q6.g Android Development	3	0	4.00	4.00	.000

q6.h iOS Development	3	0	4.00	4.00	.000
q6.i Other	0	3			
q6.j Other specified	3	0			
q7 Additional comments/suggestions about software	3	0			
q8 Level of satisfaction with overall quality of DMSE program	3	0	4.00	4.00	1.000
q9 Please use this space to expound on your answer	3	0			
q10 Strongest features of program	3	0			
q11 Areas that require most improvement	3	0			
q12 Additional comments	3	0			

Frequency Table

q1_1 I am a: Professional in the Animation industry

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	3	100.0	100.0	100.0

q1_2 I am a: Professional in the Software Engineering industry

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	1	33.3	33.3	33.3
	Selected	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q1_3 I am a: Professional in Video, Media or Advertising

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	2	66.7	66.7	66.7
	Selected	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q1_4 I am a: An employer or past employer of a DMSE alumni

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	3	100.0	100.0	100.0

q1_5 I am a: Educator

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	2	66.7	66.7	66.7
	Selected	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q1_6 I am a: Architecture

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	3	100.0	100.0	100.0

q1_7 I am a: Other

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Selected	3	100.0	100.0	100.0

q1.a Other specified

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		3	100.0	100.0	100.0

q2.a Foundation/Traditional Engineering

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	2	66.7	66.7	66.7
	Very Important	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q2.b Project Management

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	3	100.0	100.0	100.0

q2.c Collaboration

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Important	3	100.0	100.0	100.0

q2.d UML

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Unimportant	1	33.3	33.3	33.3
	Very Important	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q2.e Database Design

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	1	33.3	33.3	33.3
	Very Important	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q2.f Innovation

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	2	66.7	66.7	66.7
	Very Important	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q2.g Professional Development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	2	66.7	66.7	66.7
	Very Important	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q2.h Multiple Programming Languages

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Unimportant	1	33.3	33.3	33.3
	Very Important	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q2.i Communication

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Important	3	100.0	100.0	100.0

q2.j General Education

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	3	100.0	100.0	100.0

q2.k Other

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Somewhat Important	1	33.3	100.0	100.0
Missing	System	2	66.7		
Total		3	100.0		

q3 Other important skill sets

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	33.3	33.3	33.3
	Foundational design skills, application design, UX, and UI	1	33.3	33.3	66.7
	Software life cycle and methodologies such as Agile Also, software QA	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q4 Any specific skills stand out as excellent from our work/students

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	33.3	33.3	33.3
	ability to comprehend a complex system and become productive quickly	1	33.3	33.3	66.7
	I have no experience with students	1	33.3	33.3	100.0

q4 Any specific skills stand out as excellent from our work/students

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	33.3	33.3	33.3
	ability to comprehend a complex system and become productive quickly	1	33.3	33.3	66.7
	I have no experience with students	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q5.a Visual Studio 2010

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Very Relevant	3	100.0	100.0	100.0

q5.b Unity 3D

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	33.3	50.0	50.0
	Very Relevant	1	33.3	50.0	100.0
	Total	2	66.7	100.0	
Missing	Unknown	1	33.3		
Total		3	100.0		

q5.c Visio

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	33.3	33.3	33.3
	Somewhat Relevant	1	33.3	33.3	66.7
	Very Relevant	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q5.d Unreal Development Kit

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	33.3	50.0	50.0
	Somewhat Relevant	1	33.3	50.0	100.0
	Total	2	66.7	100.0	
Missing	Unknown	1	33.3		
Total		3	100.0		

q5.e MS Project

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	33.3	33.3	33.3
	Somewhat Relevant	1	33.3	33.3	66.7
	Very Relevant	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q5.f CryEngine

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	33.3	50.0	50.0
	Somewhat Relevant	1	33.3	50.0	100.0
	Total	2	66.7	100.0	
Missing	Unknown	1	33.3		
Total		3	100.0		

q5.g Android Development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	33.3	33.3	33.3
	Somewhat Relevant	1	33.3	33.3	66.7
	Very Relevant	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q5.h iOS Development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Not Relevant	1	33.3	50.0	50.0
	Very Relevant	1	33.3	50.0	100.0

	Total	2	66.7	100.0	
Missing	Unknown	1	33.3		
Total		3	100.0		

q5.i Other

		Frequency	Percent
Missing	Unknown	1	33.3
	System	2	66.7
	Total	3	100.0

q5.j Other specified

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		3	100.0	100.0	100.0

q6.a Visual Studio 2010

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	High Quality	3	100.0	100.0	100.0

q6.b Unity 3D

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Moderate Quality	1	33.3	33.3	33.3
	High Quality	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q6.c Visio

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Moderate Quality	1	33.3	33.3	33.3
	High Quality	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q6.d Unreal Development Kit

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Moderate Quality	1	33.3	33.3	33.3
	High Quality	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q6.e MS Project

		Frequency	Percent	Valid Percent	Cumulative Percent

Valid	Moderate Quality	1	33.3	33.3	33.3
	High Quality	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q6.f CryEngine

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Moderate Quality	1	33.3	33.3	33.3
	High Quality	2	66.7	66.7	100.0
	Total	3	100.0	100.0	

q6.g Android Development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	High Quality	3	100.0	100.0	100.0

q6.h iOS Development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	High Quality	3	100.0	100.0	100.0

q6.i Other

		Frequency	Percent
Missing	Unknown	2	66.7
	System	1	33.3
	Total	3	100.0

q6.j Other specified

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		3	100.0	100.0	100.0

q7 Additional comments/suggestions about software

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		2	66.7	66.7	66.7
	Students should be also be familiar with Unix scripting.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q8 Level of satisfaction with overall quality of DMSE program

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Neutral	1	33.3	33.3	33.3
	Somewhat Satisfied	1	33.3	33.3	66.7

	Very Satisfied	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q9 Please use this space to expound on your answer

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	33.3	33.3	33.3
	all the interns from the program have become employees, it doesn't get any better than that	1	33.3	33.3	66.7
	There should be more software being produced from the students.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q10 Strongest features of program

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		1	33.3	33.3	33.3
	critical thinking	1	33.3	33.3	66.7
	The strongest feature is its integration with the game design program which helps create software/language diversity for the students.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

q11 Areas that require most improvement

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		2	66.7	66.7	66.7
	More practice in the use of many languages and pipelines. Students should output a more diverse set of projects.	1	33.3	33.3	100.0
	Total	3	100.0	100.0	

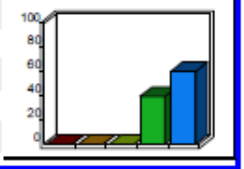
q12 Additional comments

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid		3	100.0	100.0	100.0

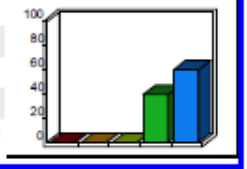
Appendix B – Sample Student Assessments

BAKER…2010-12-22…10F…SENG 100 AGA…SAI

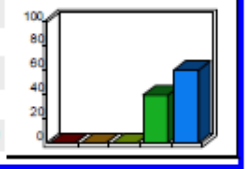
Q1 Expectations for graded assignments were clearly communicated — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

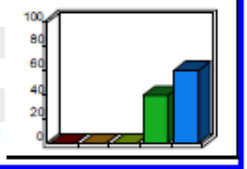
Q2 Course activities (lectures, projects, etc.) helped me learn the course material — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

Q3 Examinations, papers and other graded projects were returned in a reasonable amount of time — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

Q4 The course was well organized — Mean: 4.60

Response	Value	Frequency	Percent	Cum. Percent	Valid Percent	Cum. Val. Percent	Graph
Strongly Disagree	1.00	0	0.00	0.00	0.00	0.00	
Disagree	2.00	0	0.00	0.00	0.00	0.00	
Neutral	3.00	0	0.00	0.00	0.00	0.00	
Agree	4.00	4	40.00	40.00	40.00	40.00	
Strongly Agree	5.00	6	60.00	100.00	60.00	100.00	
Total Valid		10	100.00		100.00		

Appendix C – Course Syllabis

SENG 100

Introduction to Computer Programming

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M-Th 9:30am – 12:30pm
- Office location: Room 171
- Phone: 616-643-5721
- e-mail: FerrisConnect

Class Meetings

Day(s)	Time of Day	Location
MW	01:00pm – 02:30pm	Rm 176

For a few of the scheduled days, shown in the **course calendar**, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Software engineering is a very broad field. It encompass virtual everything a person might want to know in order to develop software -- software that is correct, on time, and on budget. Software Engineering courses emphasize the technical foundations of software development, such as programming, algorithms, data structures, languages.

This course teaches fundamentals of computer programming. Students learn how to write, test and debug small programs. Basic coding concepts and best practices are discussed and explained. Functions, data types, logical constructs required to produce software solutions will be the basis for this exploration. Students are introduced to Software Development Life-Cycles.

Objectives

Students leaving the course should:

1. Students will learn how to write a simple computer programs.

2. Students will learn the software lifecycle through assignments
3. Students will use various development environments to complete assignments.

Materials

Textbook

Required:

Learning C# 3.0
Jesse Liberty (Author)
* Pub. Date: November 2008
* Publisher: O'Reilly Media, Incorporated
* ISBN-13: 9780596521066
* ISBN: 0596521065

Suggested:

Illustrated C 2008 (Windows.Net) (Paperback)
Paperback: 728 pages
Publisher: Apress; 1st ed. 2008. Corr. 2nd printing edition (February 18, 2008)
ISBN-10: 1590599543 : ISBN-13: 978-1590599549

Class notes, assignments, and other information you need to read for the course will be provided in FerrisConnect in the 'Ref Material" folder, as the term progresses.

Assignments

Readings: You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some topics, and less depth on others.

Class Participation: The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then attempt to catch up with the rest of the class.

Examinations: There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course.

Quizzes: Short quizzes will be given in class

Assignments: Give you experience applying the principles and techniques covered in the course, project details will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before the scheduled due date and time. Late Assignments receive a 50% reduction.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will be reduced.

Academic Honor Code

You are required to read the FSU **Academic Honor Code** and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

SENG 101
Computer Programming 1
Syllabus

Instructor Contact Information

- Instructor: Frederick Baker
- URL: http://www.ferris.edu/gr/programs/bachDeg/assets/pdfs/DMSE.pdf
- Office hours: M/W/F 2:30-4:00 PM, and by e-mail and appointment
- Office location: Rm 176
- Phone: 616-643-5721
- e-mail: frederickbaker@ferris.edu

Class Meetings

Day(s)	Time of Day	Location
MW	1:00pm – 2:30pm	Rm 176

For a few of the scheduled days, shown in the course calendar, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

This course teaches fundamentals of object-oriented software development using Microsoft C#. Students learn to write, test and debug small computer programs. Basic coding concepts and best practices are discussed and explained. Topics covered include Functions, data types, logical constructs, looping techniques, user-defined classes, and arrays required to produce software.

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses:

- experience with the software development process

- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

The course is SENG 101, with a practical component. Students leaving the course are able to:

- employ branch if/if-else/switch commands to control program flow in algorithmic solutions
- create/interpret complex code expressions that use relational and boolean operators
- use do-while/for/each-next statements to control repetition in algorithmic solutions
- design and implement, and incorporate classes in a simple application
- design and implement object methods and incorporate them in a simple application
- use existing Microsoft C# class libraries
- design, implement simple Microsoft C# programs
- analyze and troubleshoot successfully C# code

Materials

Textbook: **Learning C#**

Author: Jessy Liberty

Publisher: O'Reilly Media

Table of Contents:

Chapter 14. Generics and Collections

Chapter 15. Strings

Chapter 16. Throwing and Catching Exceptions

Chapter 17. Delegates and Events

Chapter 18. Creating Windows Applications

Chapter 19. Programming ASP.NET Applications

Other References

C# Cookbook by Stephen L. Teilhet, Jay Hilyard
* Publisher: O'Reilly Media, Incorporated
* Edition Number: 2

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things, and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material.

Experience has shown a high correlation between absenteeism and low grades. Therefore, you are required to attend all class meetings. Attendance may be checked at random times throughout the term. You will be allowed three (3) absences without question or justification. After that, your "Deliverables" score will be reduced by an amount to be determined. If you have a documented religious holiday or emergency, additional absences may be excused.

Examinations. There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course.

Students are expected to take the examinations at the scheduled time and date.

If you know in advance of a reason you will not be able to take a scheduled examination you are responsible for making arrangements in advance with me. If you make prior arrangements, or have a documented last-minute emergency conflict -- such as a medical emergency or your employer requires you to make an out-of-town trip -- I will attempt to accommodate you. **However, under no circumstances will a student be allowed to take an examination before the rest of the class**. Also, it must be understood that we cannot give the same

examination, and because of time constraints, I may administer a makeup examination of an entirely different format. In the past, most students who have missed midterm examinations have chosen not to take a makeup examination.

Quizzes. Short quizzes will be given in class each week.

Programming assignments. There are several software engineering projects which you will complete individually. The objective of each project/assignment is to give you experience applying the principles and techniques covered in the course.

Project details will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before midnight on the due date.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

59 – 00 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	20%
In Class Quizzes	20%
Project Deliverables	20%
Midterm Exam	20%
Final Exam	20%

All graded work will be returned in class. Students who miss class are responsible for obtaining their papers no later than one week after the start of the next semester. The one exception are final examinations. The University requires that the original copies of all final examinations be retained in the Department for two years.

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will ordinarily be reduced. The amount of the reduction will be entirely at my discretion, and may depend on how late the assignment is. In any case, no work will be

accepted for grading after a solution is discussed or handed out, either in the class or on the web.

Academic Honor Code

You are required to read the FSU Academic Honor Code and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

In the context of this course -- because programming assignments are used to assess what you have learned, and because even a few lines of code may represent a great deal of creative thought -- we must adhere to a higher standard for originality and independence of programming assignments than you may have found for the work in some other courses.

- For the team assignments in this course, you are required to design and write your own solutions within the team. You are not allowed to show your code or documents to another person outside the team, other than me (the instructor of this course), or to read the solution code or documents of another student outside the team,
- The only source code or documents written by another person outside your team that you may reuse in a program that you turn in for grade are the following:
    1. Materials I provide for students in this course, *this term*, in a public forum, provided you clearly identify the material that is reused and where it came from.
    2. Material you found in a textbook used in a course you have taken, provided you clearly identify it as quoted and include a citation of the textbook in your comments.
    3. Material you found published on the Internet, provided you mark it clearly as quoted, include the URL in a comment, and the web site where you found it has a clear statement giving permission for anyone to reuse the material.
- If you reuse material of type (2) above, limit yourself to comparatively short blocks of quoted material. Excessively long quotations from a textbook may violate the "fair use" provision of the copyright law, and in any case are considered poor academic practice.
- **Failure to follow these rules will be considered a violation of the Academic Honor Code.** The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

SENG 102

Computer Programming 2

Syllabus

Instructor Contact Information

- Instructor: Frederick Baker
- URL: http://dmse.ferris.edu
- e-mail: frederickbaker@ferris.edu
- Phone: 616-643-5721
- Office location: ATC Room 171
- Office hours: M-Th : 11:00 AM-1:00 PM, Friday 11:00 AM – 6:00 PM

Class Meetings

Day(s)	Time of Day	Location
MW	1:00pm – 2:30pm	ATC Room 176

For a few of the scheduled days, shown in the course calendar, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Description: This course is designed to complement Computer Programming I and begin to explore more sophisticated programming concepts. Students will begin to apply advanced coding techniques using Classes, Inheritance, Polymorphism, while implementing Generics,

Regular Expressions, Delegates, ADO.NET, SQL, Web Services, Reflection, Remoting, Threading, and Streams. In addition, best practices and industry case studies are presented to anchor concepts and presented material.

Prerequisites

Successful completion **SENG 101** or **equivalent** and:

- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

The course is SENG 102, with a practical component. Students leaving the course are able to:

- explain the rationale for object-oriented design and programming
- translate UML class and sequence diagrams into Microsoft C# code
- define polymorphism
- design and implement Microsoft C# classes
- implement software systems using problem requirements analysis and use case analysis
- document the implementation of small software systems
- Implement within an application
    - Classes and Objects (2 classes)
    - Inheritance / polymorphism /Operator Overload (2 classes)
    - Exception handling (1 classes)
    - Strings and Regular Expressions (1 class)
    - ArrayList / Collections (2 classes)
    - Interfaces/Abstract classes (2 classes)
    - Delegates and Events (1 class)
    - Accessing Data with ADO.net (2 classes)
    - Web Services(2 class)
    - Reflection (2 class)
    - Marshalling and Remoting (2 classes)
    - Threads and Synchronization (2 classes)
    - Serialization
    - To effectively and efficiently use Streams (2 Classes)

Materials

Textbook: **Programming C# 4th edition**

Author: Jessy Liberty

Publisher: O'Reilly Media

Chapter 1: C# and the .NET Framework

Chapter 2: Getting Started: "Hello World"

Chapter 3: C# Language Fundamentals

Chapter 4: Classes and Objects

Chapter 5: Inheritance and Polymorphism

Chapter 6: Operator Overloading

Chapter 7: Structs

Chapter 8: Interfaces

Chapter 9: Arrays, Indexers, and Collections

Chapter 10: Strings and Regular Expressions

Chapter 11: Handling Exceptions

Chapter 12: Delegates and Events

Chapter 13: Building Windows Applications

Chapter 14: Accessing Data with ADO.NET

Chapter 15: Programming Web Application with Web Forms

Chapter 16: Programming Web Services

Chapter 17: Assemblies and Versioning

Chapter 18: Attributes and Reflection

Chapter 19: Marshaling and Remoting

Chapter 20: Threads and Synchronization

Chapter 21: Streams

Chapter 22: Programming .NET and COM

Italized – Overview ***BOLD*** *– In Depth* <u>*Underlined – if time permits*</u>

References

ISBN13: 978-1-59059-954-9

ISBN10: 1-59059-954-3

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things, and less depth on others.

Class Participation. The study of much of Digital Media Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material.

Experience has shown a high correlation between absenteeism and low grades. Therefore, you are required to attend all class meetings. Attendance may be checked at random times throughout the term. You will be allowed three (3) absences without question or justification. After that, your "Deliverables" score will be reduced by an amount to be determined. If you have a documented religious holiday or emergency, additional absences may be excused.

Examinations (Mid/Final**).**There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course.

Students are expected to take the examinations at the scheduled time and date.

If you know in advance of a reason you will not be able to take a scheduled examination you are responsible for making arrangements in advance with me. If you make prior arrangements, or have a documented last-minute emergency conflict -- such as a medical emergency or your employer requires you to make an out-of-town trip -- I will attempt to accommodate you. **However, under no circumstances will a student be allowed to take an examination before the rest of the class**. Also, it must be understood that we cannot give the same examination, and because of time constraints, I may administer a makeup examination of an entirely different format. In the past, most students who have missed midterm examinations have chosen not to take a makeup examination.

Quizzes (2).

Programming assignments. There are several projects which you will complete individually. The objective of each project/assignment is to give you experience applying the principles and techniques covered in the course.

Project details will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before midnight on the due date.

Grading

All work is assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

59 – 00 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	20%
In Class Quizzes	20%
Project Deliverables	20%
Midterm Exam	20%
Final Exam	20%

Graded work is returned in class or on the website. Students who miss class are responsible for obtaining their papers no later than one week after the start of the next semester. The one exception are final examinations. The University requires that the original copies of all final examinations be retained in the Department for two years.

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will ordinarily be reduced. The amount of the reduction will be entirely at my discretion, and may depend on how late the assignment is. In any case, no work will be accepted for grading after a solution is discussed or handed out, either in the class or on the web.

Academic Honor Code

You are required to read the FSU Academic Honor Code and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

In the context of this course -- because programming assignments are used to assess what you have learned, and because even a few lines of code may represent a great deal of creative thought -- we must adhere to a higher standard for originality and independence of programming assignments than you may have found for the work in some other courses.

- For the team assignments in this course, you are required to design and write your own solutions within the team. You are not allowed to show your code or documents to another person outside the team, other than me (the instructor of this course), or to read the solution code or documents of another student outside the team,
- The only source code or documents written by another person outside your team that you may reuse in a program that you turn in for grade are the following:
    1. Materials I provide for students in this course, *this term*, in a public forum, provided you clearly identify the material that is reused and where it came from.
    2. Material you found in a textbook used in a course you have taken, provided you clearly identify it as quoted and include a citation of the textbook in your comments.
    3. Material you found published on the Internet, provided you mark it clearly as quoted, include the URL in a comment, and the web site where you found it has a clear statement giving permission for anyone to reuse the material.
- If you reuse material of type (2) above, limit yourself to comparatively short blocks of quoted material. Excessively long quotations from a textbook may violate the "fair use" provision of the copyright law, and in any case are considered poor academic practice.
- **Failure to follow these rules will be considered a violation of the Academic Honor Code.** The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

SENG 160
Software Engineering Methodologies

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M-Th 9:30am – 12:30pm
- Office location: Room 171
- Phone: 616-643-5721
- e-mail: Use FerrisConnect

Class Meetings

Day(s)	Time of Day	Location
TTH	1:00pm – 2:15pm	Rm 176

For a few of the scheduled days, shown in the course calendar, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

This course covers the processes and procedures practiced by software engineering organizations. Methodologies and processes are presented within the framework of the software development lifecycle, project planning, metrics, peer reviews, quality control. Covers popular methodologies being used in the real world and examines the merits of each. Students learn theory and process as well as examine the effects through case study and applied scenarios.

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses:

- Ready to learn, willing to participate, positive attitude.

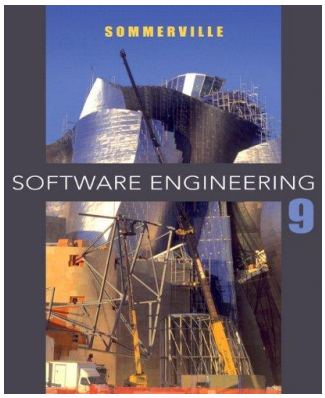
Objectives

Students leaving the course are able to:

- Effectively select a software development methodology for a project
- Can use software product metrics
- Can apply several software development processes
- Create the necessary documents required in each phase of the development.
- Effectively utilize personal design and code reviews
- Analyze process data and propose improvements
- Successfully implement all software methodologies

Materials

Required Textbook:



Software Engineering (9th Edition) (Hardcover)

Hardcover: 864 pages

Publisher: Addison Wesley; 9 edition (March 1, 2010)

Language: English

ISBN-10: 0137035152

ISBN-13: 978-0137035151

Assignments

Homework and Readings. You will be assigned readings and questions from the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some topics.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Experience has shown a high correlation between absenteeism and low grades. Therefore, you are required to attend all class meetings. Attendance may be checked at random times throughout the term. You will be allowed three (3) absences without question or justification. After that, your "Deliverables" score will be reduced by an amount to be determined. If you have a documented religious holiday or emergency, additional absences may be excused.

Examinations. There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room.

Students will be expected to take the examinations at the scheduled time and date.

If you know in advance of a reason you will not be able to take a scheduled examination you are responsible for making arrangements in advance with me. If you make prior arrangements, or have a documented last-minute emergency conflict -- such as a medical emergency or your employer requires you to make an out-of-town trip -- I will attempt to accommodate you. **However, under no circumstances will a student be allowed to take an examination before the rest of the class**. Also, it must be understood that we cannot give the same examination, and because of time constraints, I may administer a makeup examination of an entirely different format. In the past, most students who have missed midterm examinations have chosen not to take a makeup examination.

Quizzes. 2 Short quizzes will be given in class.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

59 – 00 F

All graded work will be returned in class. Students who miss class are responsible for picking up their papers no later than one week after the start of the next semester. Work that is not picked up by then may be discarded. The one exception are final examinations. The University requires that the original copies of all final examinations be retained in the Department for two years.

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will ordinarily be reduced by 50%.

Academic Honor Code

You are required to read the FSU Academic Honor Code and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

SENG 170
Software Engineering Requirements
Syllabus

Instructor Contact Information

- Instructor: Frederick Baker
- URL: http://dmse.ferris.edu
- e-mail: frederickbaker@ferris.edu
- Phone: 616-643-5721
- Office location: ATC Room 171
- Office hours: M-Th : 11:00 AM-1:00 PM, Friday 11:00 AM – 6:00 PM

Class Meetings

Day(s)	Time of Day	Location
M/W	03:00pm – 4:30pm	Rm 176

For a few of the scheduled days, shown in the course calendar, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

This course covers the tools and methods involved in capturing and tracing end user requirements through the software development process. Examines dynamics and scenarios that organizations deal with in identifying requirements and processes used in defining software needs.

Prerequisites (SENG 160)

The following are the minimum specific capabilities you will need from the prerequisite courses:

- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

The course is SENG 170, with a practical component. Students will learn to:

- Create and document the Requirements derived from Customer feedback
- Understand Requirement Management Principles and Practices
- Validate the Requirements
- Specify software requirements using industry standard documentation techniques specify requirements that are verifiable, traceable, measurable and testable
- Verify that specified requirements are accurate, unambiguous, complete and consistent
- Communicate software requirements in written documents and oral presentations

Materials

Software Requirements, Second Edition

by Karl E. Wiegers

Publisher: Microsoft Press:

Pub Date: February 26, 2003 : Print ISBN-10: 0-7356-1879-8 : Print ISBN-13: 978-0-7356-1879-4

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Experience has shown a high correlation between absenteeism and low grades. Therefore, you are required to attend all class meetings. Attendance may be checked at random times throughout the term. You will be allowed three (3) absences without question or justification. After that, your "Deliverables" score will be reduced by an amount to be determined. If you have a documented religious holiday or emergency, additional absences may be excused.

Examinations. There will be a **Midterm** examination and a **Final** examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room.

Students will be expected to take the examinations at the scheduled time and date.

If you know in advance of a reason you will not be able to take a scheduled examination you are responsible for making arrangements in advance with me. If you make prior arrangements, or have a documented last-minute emergency conflict -- such as a medical emergency or your employer requires you to make an out-of-town trip -- I will attempt to accommodate you. **However, under no circumstances will a student be allowed to take an examination before the rest of the class**. Also, it must be understood that we cannot give the same examination, and because of time constraints, I may administer a makeup examination of an entirely different format. In the past, most students who have missed midterm examinations have chosen not to take a makeup examination.

Quizzes. 2 quizzes will be given in class.

Projects. There are 3 projects. The objective of each project is to give you experience applying the principles and techniques covered in the course. More details on the project will be presented as the term progresses. Unless specified otherwise, deliverables must be submitted before midnight on the due date.

Homework: There are 8 homework assignments

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

59 – 00 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	20%
Quizzes	20%
Projects	20%
Midterm Exam	20%
Final Exam	20%

All graded work will be returned in class. Students who miss class are responsible for picking up their papers no later than one week after the start of the next semester. Work that is not picked up by then may be discarded. The one exception are final examinations. The University requires that the original copies of all final examinations be retained in the Department for two years.

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will ordinarily be reduced. The amount of the reduction will be entirely at my discretion, and may depend on how late the assignment is. In any case, no work will be accepted for grading after a solution is discussed or handed out, either in the class or on the web.

Academic Honor Code

You are required to read the FSU Academic Honor Code and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

In the context of this course -- because programming assignments are used to assess what you have learned, and because even a few lines of code may represent a great deal of creative thought -- we must adhere to a higher standard for originality and independence of programming assignments than you may have found for the work in some other courses.

- For the team assignments in this course, you are required to design and write your own solutions within the team. You are not allowed to show your code or documents to another person outside the team, other than me (the instructor of this course), or to read the solution code or documents of another student outside the team,
- The only source code or documents written by another person outside your team that you may reuse in a program that you turn in for grade are the following:
    1. Materials I provide for students in this course, *this term*, in a public forum, provided you clearly identify the material that is reused and where it came from. (Material from other courses, or prior offerings of this course may *not* be used.)
    2. Material you found in a textbook used in a course you have taken, provided you clearly identify it as quoted and include a citation of the textbook in your comments.
    3. Material you found published on the Internet, provided you mark it clearly as quoted, include the URL in a comment, and the web site where you found it has a clear statement giving permission for anyone to reuse the material.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

SENG 210

Software Configuration Management

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M/T/Th 12:30pm - 4:30 PM
- Office location: Rm 176
- Phone: 616-643-5721
- e-mail: frederickbaker@ferris.edu

Class Meetings

Day(s)	Time of Day	Location
T/Th	5:00pm – 6:30pm	Rm 176

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses:

- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

Students leaving the course can:

- Develop the next version of a product while fixing problems with the current one.
- Develop code in parallel with other developers and join up with the current state of codeline.
- Identify what versions of code went into a particular component.
- Analyze where a change happened in the history of a component's development.
- Use current tools more effectively, and decide when to use a manual process.
- Identify crucial aspects of the software process so that team projects can run smoothly.

Materials

Textbook: Software Configuration Management Patterns: Effective Teamwork, Practical Integration

Author: Stephen P. Berczuk; Brad Appleton

Publisher: Addison-Wesley Professional

ISBN: 978-0-201-74117-9

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things, and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will be reduced.

SENG 300

Data Structures and Algorithms

Syllabus

Instructor Contact Information

- Instructor: Frederick Baker
- URL: http://dmse.ferris.edu
- e-mail: frederickbaker@ferris.edu
- Phone: 616-643-5721
- Office location: ATC Room 171
- Office hours: M-Th : 11:00 AM-1:00 PM, Friday 11:00 AM – 6:00 PM

Class Meetings

Day(s)	Time of Day	Location
T/Th	01:00pm – 2:30pm	Rm 176

For a few of the scheduled days, shown in the course calendar, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

This course explores advanced and abstract data structures and algorithms. Students will learn how to identify and correctly apply these Algorithms to data structures to solve simple and complex programming problems. Topics include Sorting/Searching, Trees, Dictionaries, Linked Lists, Stacks, Queues, and Hash tables.

Prerequisites (SENG 102)

The following are the minimum specific capabilities you will need from the prerequisite courses:

- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

The course is SENG 300, with a practical component. Students leaving the course are able to:

- Understand the importance of effectively employing Data Structures and Algoritms
- Can apply BitArrays, Arrays and Arraylists to problem domains
- Can implement Dictionaries, Hashtables to enhance application performance
- Effectively utilize Pattern Matching and Text Processing
- Analyze process data and propose improvements using Algorithms
- Successfully implement Data Structures and Algorithms covered in class

Materials

Required Text:

Data Structures and Algorithms for Game Developers

By: **Allen Sherrod**
Publisher: Course Technology PTR
Print ISBN-10: 1-58450-495-1
Print ISBN-13: 978-1-58450-495-5
Web ISBN-10: 1-58450-663-6
Web ISBN-13: 978-1-58450-663-8

Optional:

Textbook: **Data Structures and Algorithms Using C#**

by Michael McMillan (Author)

Hardcover: 366 pages

Publisher: Cambridge University Press; 1 edition (March 26, 2007)

ISBN-10: 0521876915

ISBN-13: 978-0521876919

Textbook: **Data Structures and Algorithms, (Paperback)**

by Alfred V. Aho (Author), Jeffrey D. Ullman (Author), John E. Hopcroft (Author)
Paperback: 427 pages
Publisher: Addison Wesley (January 11, 1983)
ISBN-10: 0201000237
ISBN-13: 978-0201000238

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Experience has shown a high correlation between absenteeism and low grades. Therefore, you are required to attend all class meetings. Attendance may be checked at random times throughout the term. You will be allowed three (3) absences without question or justification. After that, your "Deliverables" score will be reduced by an amount to be determined. If you have a documented religious holiday or emergency, additional absences may be excused.

Examinations. There will be a **midterm** examination and a **final** examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room.

Students will be expected to take the examinations at the scheduled time and date.

If you know in advance of a reason you will not be able to take a scheduled examination you are responsible for making arrangements in advance with me. If you make prior arrangements, or have a documented last-minute emergency conflict -- such as a medical emergency or your employer requires you to make an out-of-town trip -- I will attempt to accommodate you. **However, under no circumstances will a student be allowed to take an examination before the rest of the class**. Also, it must be understood that we cannot give the same examination, and because of time constraints, I may administer a makeup examination of an entirely different format. In the past, most students who have missed midterm examinations have chosen not to take a makeup examination.

Quizzes. 2 Short quizzes will be given in class.

Programming assignments. There will be 2 projects. The objective of each project is to give you experience applying the principles and techniques covered in the course.

More details on the project will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before midnight on the due date.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

59 – 00 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	20%
In Class Quizzes	20%
Project Deliverables	20%
Midterm Exam	20%
Final Exam	20%

All graded work will be returned in class. Students who miss class are responsible for picking up their papers no later than one week after the start of the next semester. Work that is not picked up by then may be discarded. The one exception are final examinations. The University requires that the original copies of all final examinations be retained in the Department for two years.

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will ordinarily be reduced. The amount of the reduction will be entirely at my discretion, and may depend on how late the assignment is. In any case, no work will be accepted for grading after a solution is discussed or handed out, either in the class or on the web.

Academic Honor Code

You are required to read the FSU Academic Honor Code and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

In the context of this course -- because programming assignments are used to assess what you have learned, and because even a few lines of code may represent a great deal of creative thought -- we must adhere to a higher standard for originality and independence of programming assignments than you may have found for the work in some other courses.

- For the team assignments in this course, you are required to design and write your own solutions within the team. You are not allowed to show your code or documents to another person outside the team, other than me (the instructor of this course), or to read the solution code or documents of another student outside the team,
- The only source code or documents written by another person outside your team that you may reuse in a program that you turn in for grade are the following:
    1. Materials I provide for students in this course, *this term*, in a public forum, provided you clearly identify the material that is reused and where it came from. (Material from other courses, or prior offerings of this course may *not* be used.)
    2. Material you found in a textbook used in a course you have taken, provided you clearly identify it as quoted and include a citation of the textbook in your comments.
    3. Material you found published on the Internet, provided you mark it clearly as quoted, include the URL in a comment, and the web site where you found it has a clear statement giving permission for anyone to reuse the material.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

SENG 301

Programming Languages

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M/T/W 9:00am – 12:00pm and 3:00-4:00 PM
- Office location: Rm 176
- Phone: 616-643-5721
- e-mail: frederickbaker@ferris.edu

Class Meetings

Day(s)	Time of Day	Location
MW	4:00pm – 5:15pm	Rm 176

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses:

- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

Students completing the course will:

- have developed a wide-ranging perspective on over 40 different languages
- understand compilation and linking, and how data types are implemented in memory
- understand functional and logical programming
- have expanded understanding of how programming languages work
- know how code is compiled (or interpreted) and executed on computer hardware

Materials

Textbook: **Programming Language Pragmatics, Second Edition**

Author: Michael Scott

Publisher: Morgan Kaufmann; 2nd edition

ISBN: 978-0126339512

Assignments

Readings. You will be assigned readings in the textbook, and on the web. In class we will go into more depth on some things, and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	15%
Quizzes	15%
Project	20%
Midterm Exam	25%

Final Exam	25%

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will be reduced.

SENG 302
Software Quality Assurance
Syllabus

Instructor Contact Information

- Instructor: Frederick Baker – "Rick"
- URL: http://www.ferris.edu/gr/programs/bachDeg/assets/pdfs/DMSE.pdf
- Office hours: M/W/F 2:00-3:30 PM by e-mail and appointment
- Office location: Rm 171
- Phone: 616-643-5721
- e-mail: frederickbaker@ferris.edu

Class Meetings

Day(s)	Time of Day	Location
T/Th	04:00pm – 5:15pm	Rm 176

For a few of the scheduled days, shown in the course calendar, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

This course introduces the concepts of software quality assurance. Students learn processes involved in quality management. The type of testing addressed in this course is black-box testing at the system and acceptance levels. Topics include test planning and documentation, risk analysis, defect reporting and tracking, exploratory and plan-driven strategies, and test management and reporting. Topics include designing test cases, improving software quality, testing methods and tools.

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses:

- Unit testing understanding
- Experience using high-level programming language
- Understanding of software development life-cycle models

Objectives

The course is SENG 302, with a practical component. Students leaving the course are able to:

- Apply the basic principles of software quality assurance
- Create unit and system tests that integrate into the software development process.
- Understand the role of testing
- Can create and execute manual and automated testing.
- Create test plans that exercise the software application boundaries.
- Design robust and repeatable tests from use cases that exercise and provides thorough code coverage

Materials

Software Quality Assurance
Hardcover: 616 pages
Publisher: Addison Wesley (September 21, 2003)
Language: English
ISBN-10: 0201709457
ISBN-13: 978-0201709452

Table of Contents:

Part I: **Introduction (Ch. 1 – 4)**

Part II: **Pre-Project Software Quality Components (Ch. 5 -6)**

Part III: **SQA Components in the Project Life Cycle(Ch. 7 - 13)**

Part IV: **Software Quality Infrastructure Components (Ch. 14 – 19)**

Part V: **Software Quality Management Components (Ch. 20 – 22)**

Part VI: **Standards, Certification and Assessment (Ch. 23 - 25)**

Part VII: **Organizing for Quality Assurance (Ch. 26 – 28)**

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Experience has shown a high correlation between absenteeism and low grades. Therefore, you are required to attend all class meetings. Attendance may be checked at random times throughout the term. You will be allowed three (3) absences without question or justification. After that, your "Deliverables" score will be reduced by an amount to be determined. If you have a documented religious holiday or emergency, additional absences may be excused.

Examinations. There will be a **midterm** examination and a **final** examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. Students will be expected to take the examinations at the scheduled time and date.

If you know in advance of a reason you will not be able to take a scheduled examination you are responsible for making arrangements in advance with me. If you make prior arrangements, or have a documented last-minute emergency conflict -- such as a medical emergency or your employer requires you to make an out-of-town trip -- I will attempt to accommodate you. **However, under no circumstances will a student be allowed to take an examination before the rest of the class**. Also, it must be understood that we cannot give the same examination, and because of time constraints, I may administer a makeup examination of an entirely different format. In the past, most students who have missed midterm examinations have chosen not to take a makeup examination.

Quizzes. 2 quizzes will be given in class.

Projects. I will assign 3 projects. The objective of each project is to give you experience applying the principles and techniques covered in the course. More details on the project will be presented as the term progresses. Unless specified otherwise, deliverables must be submitted before midnight on the due date.

Homework: I will assign 8 homework assignments

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

59 – 00 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	15%
Quizzes	15%
Projects	20%
Midterm Exam	25%
Final Exam	25%

All graded work will be returned in class. Students who miss class are responsible for picking up their papers no later than one week after the start of the next semester. Work that is not picked up by then may be discarded. The one exception are final examinations. The University requires that the original copies of all final examinations be retained in the Department for two years.

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will ordinarily be reduced. The amount of the reduction will be entirely at

my discretion, and may depend on how late the assignment is. In any case, no work will be accepted for grading after a solution is discussed or handed out, either in the class or on the web.

Academic Honor Code

You are required to read the FSU Academic Honor Code and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

In the context of this course -- because programming assignments are used to assess what you have learned, and because even a few lines of code may represent a great deal of creative thought -- we must adhere to a higher standard for originality and independence of programming assignments than you may have found for the work in some other courses.

- For the team assignments in this course, you are required to design and write your own solutions within the team. You are not allowed to show your code or documents to another person outside the team, other than me (the instructor of this course), or to read the solution code or documents of another student outside the team,
- The only source code or documents written by another person outside your team that you may reuse in a program that you turn in for grade are the following:
    1. Materials I provide for students in this course, *this term*, in a public forum, provided you clearly identify the material that is reused and where it came from. (Material from other courses, or prior offerings of this course may *not* be used.)
    2. Material you found in a textbook used in a course you have taken, provided you clearly identify it as quoted and include a citation of the textbook in your comments.
    3. Material you found published on the Internet, provided you mark it clearly as quoted, include the URL in a comment, and the web site where you found it has a clear statement giving permission for anyone to reuse the material.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

SENG 310

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: T/Th 11:30am – 01:30pm, and by e-mail and appointment
- Office location: Rm 171
- Phone: 616-643-5721
- e-mail: Use FerrisConnect

Class Meetings

Day(s)	Time of Day	Location
T/Th	03:30pm – 05:00pm	Rm 176

For a few of the scheduled days, shown in the **course calendar**, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Software engineering is a very broad field. It encompass virtual everything a person might want to know in order to develop software -- software that is correct, on time, and on budget. Software Engineering courses emphasize the technical foundations of software development, such as programming, algorithms, data structures, languages.

This course introduces students to the concepts and practices involved in creating Application Programming Interfaces (API). Students learn how to make use of existing API's and examine how their developers designed them. As the course progresses they will learn to design and create usable, reliable API's and Software Development Kits (SDKs).

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses: SENG 160, 300 and

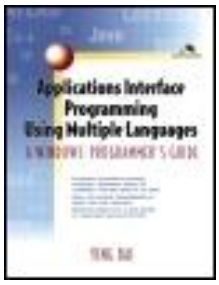
- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

1. Students will learn how to find information and use any exposed application interfaces (SDKs).
2. Students will be able to design and build their own application programming interfaces.
3. Understand how APIs and SDKs are used in the software development industry.

Materials

Textbook



Applications Interface Programming Using Multiple Languages: A Windows® Programmer's Guide
by Ying Bai

Publisher: **Prentice Hall**

Pub Date: **March 21, 2003**

Print ISBN-10: **0-13-100313-5**

Print ISBN-13: **978-0-13-100313-2**

References

Class notes, assignments, and other information you need to read for the course will be provided in FerrisConnect in the “Ref Material” folder under the “Course Content” selection.

Assignments

Readings: You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly.

Class Participation: The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Examinations: There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course. Students are expected to take the examinations at the scheduled time, shown on the **course calendar**.

Quizzes: Short quizzes will be given in class

Assignments / Projects: Give you experience applying the principles and techniques covered in the course, project details will be presented as the term progresses.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	15%
Quizzes	15%
Projects	20%
Midterm Exam	25%
Final Exam	25%

Acceptance of late turn-ins of assignments will be entirely at my discretion. **If a late assignment is accepted, the score will be reduced.**

Academic Honor Code

You are required to read the FSU **Academic Honor Code** and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be

explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

SENG 315

Software Component Design

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: T/Th 10:00am – 01:00pm, and by e-mail and appointment
- Office location: Rm 171
- Phone: 616-643-5721
- e-mail: Use FerrisConnect

Class Meetings

Day(s)	Time of Day	Location
T/Th	01:00pm – 03:00pm	Rm 176

For a few of the scheduled days, shown in the **course calendar**, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Software engineering is a very broad field. It encompass virtual everything a person might want to know in order to develop software -- software that is correct, on time, and on budget. Software Engineering courses emphasize the technical foundations of software development, such as programming, algorithms, data structures, languages.

This course introduces students to the concepts and practices involved in creating Application Programming Interfaces (API). Students learn how to make use of existing API's and examine how their developers designed them. As the course progresses they will learn to design and create usable, reliable API's and Software Development Kits (SDKs).

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses: SENG 160, 300

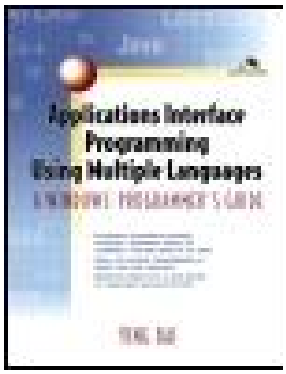
- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

1. Students will learn how to find information and use any exposed application interfaces (SDKs).
2. Students will be able to design and build their own application programming interfaces.
3. Understand how APIs and SDKs are used in the software development industry.

Materials

Textbook



Applications Interface Programming Using Multiple Languages: A Windows® Programmer's Guide
by Ying Bai

Publisher: **Prentice Hall**

Pub Date: **March 21, 2003**

Print ISBN-10: **0-13-100313-5**

Print ISBN-13: **978-0-13-100313-2**

References

Class notes, assignments, and other information you need to read for the course will be provided in FerrisConnect in the “Ref Material” folder under the “Course Content” selection.

Assignments

Readings: You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly.

Class Participation: The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Examinations: There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course. Students are expected to take the examinations at the scheduled time, shown on the **course calendar**.

Assignments / Projects: Give you experience applying the principles and techniques covered in the course, project details will be presented as the term progresses.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

Acceptance of late turn-ins of assignments will be entirely at my discretion. **If a late assignment is accepted, the score will be reduced.**

Academic Honor Code

You are required to read the FSU **Academic Honor Code** and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation

will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

SENG 350 Software Design and Architecture

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M/T/W/Th 11:30am – 1:00pm
- Office location: Room 171
- Phone: 616-643-5721
- e-mail: Use FerrisConnect

Class Meetings

Day	Time of Day	Location
Wednesday	01:00pm – 04:00pm	Rm 176

For a few of the scheduled days, shown in the **course calendar**, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Software engineering is a very broad field. It encompass virtual everything a person might want to know in order to develop software -- software that is correct, on time, and on budget. Software Engineering courses emphasize the technical foundations of software development, such as programming, algorithms, data structures, languages.

This course examines the process and design techniques employed by software architects to design enterprise scale software. Students will be introduced to design tools as well as diagramming techniques and other methods for communicating software designs. Students will develop complete software solutions using design and architecture techniques and best practices.

Objectives

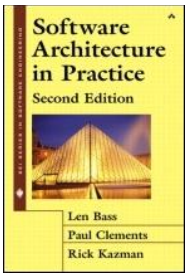
Upon course completion the student:

1. Understands the role of a Software Architect.
2. Can design software using various modeling
3. Can use the tools available for software architecture.

4. Can apply Software Design techniques in projects.
5. Appropriately select software design patterns for a given software architecture.

Materials

Textbook



Software Architecture in Practice, Second Edition

- **By:** Len Bass; Paul Clements; Rick Kazman
- **Publisher:** Addison-Wesley Professional
- **Pub. Date:** April 09, 2003
- **Print ISBN-10:** 0-321-15495-9
- **Print ISBN-13:** 978-0-321-15495-8

References

Class notes, assignments, and other information you need to read for the course will be provided in FerrisConnect in the 'Ref Material" folder, as the term progresses.

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things, and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Examinations. There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course.

Quizzes. 2 Short quizzes will be given in class

Projects. 2 Projects will provide experience applying the principles and techniques covered in the course, project details will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before midnight on the due date shown on the **course calendar**.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

The following are the items that will be graded, and the weights they will be given:

Item	Weight
Home Work	15%
Quizzes	15%
Projects	20%
Midterm Exam	25%
Final Exam	25%

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will be reduced.

Academic Honor Code

You are required to read the FSU **Academic Honor Code** and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be

explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

SENG 400 Engineering Enterprise Software Applications

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M/T/W 11:30am – 1:00pm
- Office location: Room 171
- Phone: 616-643-5721
- e-mail: Use FerrisConnect

Class Meetings

Day	Time of Day	Location
Monday	01:00pm – 03:00pm	Rm 176

For a few of the scheduled days, shown in the **course calendar**, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Software engineering is a very broad field. It encompass virtual everything a person might want to know in order to develop software -- software that is correct, on time, and on budget. Software Engineering courses emphasize the technical foundations of software development, such as programming, algorithms, data structures, languages.

This course discusses the concepts involved in designing large scale applications. Architectural concerns are examined along with issues stemming from multiple concurrent users. Students will also gain an understanding of the importance of performance when implementing applications that have high volume usage.

Objectives

Upon course completion the student can:

- Work effectively and demonstrate initiative as a project team member
- Demonstrate basic knowledge of the .NET and Enterprise Architecture
- Communicate appropriate project aspects to a variety of customers in a public forum
- Manage project resources, risks, and contingency plans
- Perform research and investigate technologies to reduce project risks and support design and planning
- Identify and address relevant engineering standards and constraints in a design project context
- Prepare appropriate documentation for a complex project
- Design, implement, and test hardware and software components and systems, if appropriate

Materials

Textbook

Microsoft .NET: Architecting Applications for the Enterprise (PRO-Developer) (Paperback)

Paperback: 464 pages
Publisher: Microsoft Press; 1 edition (October 15, 2008)
Language: English
ISBN-10: 073562609X
ISBN-13: 978-0735626096

Suggested

Professional Enterprise .NET (Wrox Programmer to Programmer)

Paperback: 504 pages

Publisher: Wrox (October 12, 2009)

Language: English

ISBN-10: 0470447613

ISBN-13: 978-0470447611

References

Class notes, assignments, and other information you need to read for the course will be provided in FerrisConnect in the 'Ref Material" folder, as the term progresses.

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things, and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Examinations. There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course.

Quizzes. 2 quizzes will be given in class

Projects. 2 Projects will provide experience applying the principles and techniques covered in the course, project details will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before midnight on the due date shown on the **course calendar**.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale: Earned Pts / Total Pts

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will be reduced by **50%.**

Academic Honor Code

You are required to read the FSU **Academic Honor Code** and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems. To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

SENG 420

Software Development Industry Certification

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M-Th: 9:30a – 12:00p
- Office location: Rm 171
- Phone: 616-643-5721
- e-mail: Use FerrisConnect

Class Meetings

Day(s)	Time of Day	Location
MW	3:30pm – 05:00pm	Rm 176

For a few of the scheduled days, shown in the **course calendar**, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Software engineering is a very broad field. It encompass virtual everything a person might want to know in order to develop software -- software that is correct, on time, and on budget. Software Engineering courses emphasize the technical foundations of software development, such as programming, algorithms, data structures, languages.

This course allows students the ability to prepare for a highly regarded industry certification program. Students will go through the recommended training materials and coursework and prepare to take the certification exam. On course completion, the exam is taken and certification earned. Certification is not guaranteed and dependent on student's ability to pass exam. Extra exam fee applicable.

Prerequisites

The following are the minimum specific capabilities you will need from the prerequisite courses:

- experience with the software development process
- skill in independent programming and problem solving
- skill using an object oriented language

Objectives

Students completing this course

1. Will complete the coursework for an industry recognized certification
2. Will understand the requirements of the certification.
3. Will take the certification exam at the completion of the course.

Materials

Textbook

Required

MCTS Self-Paced Training Kit (Exam 70-536): Microsoft® .NET Framework—Application Development Foundation, Second Edition
By: Tony Northrup
Publisher: Microsoft Press
Pub. Date: November 12, 2008
Print ISBN-10: 0-7356-2619-7
Print ISBN-13: 978-0-7356-2619-5

MCTS Self-Paced Training Kit (Exam 70-505): Microsoft® .NET Framework 3.5 -- Windows Forms Application Development
Published: February 25, 2009
Author: Matthew A. Stoecker, Steven J. Stein
ISBN 13:9780735626379
ISBN 10: 0-7356-2637-5

References

Class notes, assignments, and other information you need to read for the course will be provided in FerrisConnect in the "Ref Material" folder under the "Course Content" selection.

Assignments

Readings: You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things, and less depth on others.

Class Participation: The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Examinations: Practice Tests are assigned at the completion of each chapter

Assignments: Give you experience applying the principles and techniques covered in the course, project details will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before midnight on the due date shown on the **course calendar**.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale:

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will be reduced.

Academic Honor Code

You are required to read the FSU **Academic Honor Code** and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be

explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

You are required to verify that your e-mail address is included in the class e-mail list. I will send everyone a message at the start of the term. If you don't get that first message, make sure you are added to the list.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems.

To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

SENG 430 Programming Graphical User Interfaces

Instructor Contact Information

- Instructor: **Frederick Baker**
- Office hours: M/T/W 11:30am – 1:00pm
- Office location: Room 171
- Phone: 616-643-5721
- e-mail: Use FerrisConnect

Class Meetings

Day	Time of Day	Location
Monday	03:00pm – 05:00pm	Rm 176

For a few of the scheduled days, shown in the **course calendar**, there may be no regular class because I am required to travel on University business. This class time will be made up by specially scheduled class meetings (to be arranged) in which students will demonstrate their projects for me and the other students who are able to attend. These meetings will be arranged at multiple times, if necessary, so that every student can demonstrate his or her project.

Catalog Description & Overview

Software engineering is a very broad field. It encompass virtual everything a person might want to know in order to develop software -- software that is correct, on time, and on budget. Software Engineering courses emphasize the technical foundations of software development, such as programming, algorithms, data structures, languages.

This course focuses on the techniques and technologies employed in creating Software User Interfaces (UI). Students learn the importance of an efficient and effective user interface design along with gaining an appreciation for programming techniques used to enhance the human user experience. UI methodologies and frameworks are explored.

Objectives

Upon course completion the student can:

- Demonstrate a broad familiarity with the major areas of current GUI development and research
- Successfully apply a variety of GUI design processes and techniques.
- Examine GUI designs and implement improvements where needed.
- Apply usability evaluation methods and know when they are appropriate
- Design GUI to accommodate the diversity of users
- Utilize different interaction styles and their pros and cons
- Analyze GUI experiences and develop solutions by observing users and their challenges

Materials

Textbook

Designing Interfaces: Patterns for Effective Interaction Design (Paperback)
Paperback: 352 pages
Publisher: O'Reilly Media; 1 edition (November 21, 2005)
Language: English
ISBN-10: 0596008031
ISBN-13: 978-0596008031

Suggested
GUI Bloopers 2.0, Second Edition: Common User Interface Design Don'ts and Dos
Paperback: 424 pages
Publisher: Morgan Kaufmann; 2 edition (September 10, 2007)
Language: English
ISBN-10: 0123706432
ISBN-13: 978-0123706430

References

Class notes, assignments, and other information you need to read for the course will be provided in FerrisConnect in the 'Ref Material" folder, as the term progresses.

Assignments

Readings. You will be assigned readings in the textbook, and on the web. However, what we cover in class will not correspond exactly. In class we will go into more depth on some things, and less depth on others.

Class Participation. The study of much of Software Engineering is cumulative (*i.e.*, understanding earlier material well is necessary to grasp later material. Do not allow yourself to fall behind, by postponing studying, and then figure you will jump ahead to catch up with the rest of the class.

Examinations. There will be a midterm examination and a final examination. These examinations will be "closed book". That is, no books or reference materials will be allowed in the examination room. These are the main check on how much you have learned from the course.

Quizzes. 2 quizzes will be given in class

Projects. 2 Projects will provide experience applying the principles and techniques covered in the course, project details will be presented as the term progresses.

Unless specified otherwise, deliverables must be submitted before midnight on the due date shown on the **course calendar**.

Grading

Every piece of graded work will be assigned a score in the range 0 to 100. At the end of the term these numeric grades will be averaged to compute your final grade for the term, using the following scale: Earned Pts / Total Pts

100 – 93 A ; 92-90 A-

89 – 87 B+; 86 – 83 B; 82 – 80 B-

79 – 77 C+; 76 – 73 C; 72 – 70 C-

69 – 67 D+; 66 – 63 D; 62 – 60 D-

00 – 59 F

Acceptance of late turn-ins of assignments will be entirely at my discretion. If a late assignment is accepted, the score will be reduced by **50%.**

Academic Honor Code

You are required to read the FSU **Academic Honor Code** and abide by it. Take note that this policy not only puts you on your honor not to cheat. It requires that you report any cheating that you observe. First violations will result in lowering of the final course grade by one whole letter. Repeat violations will result in a grade of F with no provision for retaking the course.

By turning in work for a grade in this course you are representing it as being entirely your own individual work. Unless otherwise specified in writing, all homework (including all programming assignments) is expected to be individual work. If any assignment permits teamwork, it will be explicitly stated so in the assignment, and then the work is required to be only the work of the people on the team.

Failure to follow these rules will be considered a violation of the Academic Honor Code. The first detected violation will result in a *negative* score, of -100% A second detected violation will result in a grade of "F" for the course, with no possibility of taking the course for grade forgiveness.

Communication

If you are experiencing difficulty or are concerned about your progress, please contact me right away. Problems are usually easier to solve when they are addressed early.

Check regularly for electronic mail sent to you containing information about this course. You are also encouraged to use e-mail to ask questions and report problems. To protect against impersonation, please use only your FSU e-mail account.

Tobacco Free Environment

The use of tobacco products is not allowed in/on any GRCC building, grounds or parking structure. For more information on this policy or for resources about quitting go to www.grcc.edu/tobaccofree

Appendix D – Curriculum Map

Program - Digital Media Software Engineering (B.A.S.) - Curriculum Map

Legend: (A) - Program Assessment, (I) - Introduced, (M) - Mastery, (R) - Reinforced

Outcomes	APPS 301	BLAW 321	COMM	DAGD 15	DAGD 25	DAGD 30	DAGD 35	ECON 22	ENGL 150	ENGL 250	ENGL 311	ENGL 325	MATH 22	MATH 23	MATH 32	MFGE 341	MFGE 423	MFGE 442	PHIL 216	PHYS 241	PHYS 242	SENG 10	SENG 10	SENG 10	SENG 16	SENG 17	SENG 21	SENG 30	SENG 30	SENG 30	SENG 31	SENG 35	SENG 35	SENG 40	SENG 42	SENG 43	SENG 49	SENG 499
Development of professional grade skills in Software Application Development					I		R															I	I	R				R	R		R				R	M, R	M, R
Development of professional grade skills in Software Project Management	R																					I	I	R	I	R	R	M	I	R	M	R	R	M	M	I	R	M
Development of professional grade skills in Software Development Processes																									I	R	R			R			R	M, R		M, R	M, R

Outcomes	APPS 301	BLAW 321	COMM	DAGD 15	DAGD 25	DAGD 30	DAGD 35	ECON 22	ENGL 150	ENGL 250	ENGL 311	ENGL 325	MATH 22	MATH 23	MATH 32	MFGE 341	MFGE 423	MFGE 442	PHIL 216	PHYS 241	PHYS 242	SENG 10	SENG 10	SENG 10	SENG 16	SENG 17	SENG 21	SENG 30	SENG 30	SENG 30	SENG 31	SENG 35	SENG 35	SENG 40	SENG 42	SENG 43	SENG 49	SENG 499
Development of professional grade skills in Software Development Methodologies																									I	I, R	I, R			I, R							M, R
Development of professional grade skills in Software Quality Assurance																									I					I, M						R	R
Development of professional grade skills in Software Analysis and Design																								I							I		M, R				M, R
Development of Industry Readiness																																I	I	I, M	I	M, R	R

Appendix E - Program Assessment

Digital Media Software Engineering (B.S.)

Mission Statement: To provide a digital media software engineering program with formal and rigorous engineering approaches, that employ significant and relevant application of learned materials, with the intent of producing software engineers that are capable, confident, and competent designing, developing, and deploying effective innovative software systems.

Advisory Board/Committee Meetings:
Once per year

Next FSU Academic Program Review:
2012-2013

College: CPTS

Outcome: Development of professional grade skills in Software Application Development

Student demonstration of current industry accepted programming languages that are necessary for the development of quality software.

Outcome Type: Learning

Means of Assessment

Student demonstration of completed Software Projects that employ branch control, loop control and program flow commands from a high level programming language
3.0 average on a 4.0 point scale

Assessment Method Category:
Project/Model/Invention
Oral presentation in the capstone course detailing the implementation of branch, loop, and control flow commands in the capstone final project
3.0 average on a 4.0 point scale

Assessment Method Category:
Presentation(Oral)
Student demonstration of UML class diagrams in 5 projects that incorporate and implement Object-Oriented techniques
3.0 average on a 4.0 point scale

Assessment Method Category:
Project/Model/Invention

Outcome: Development of professional grade skills in Software Project Management

Student demonstration of the ability to apply the awareness, knowledge, and core competencies necessary to develop requirements and processes for software development.

Means of Assessment

Successful completion of the management of 5 software projects utilizing resources, developing software plans and identifying project risks
3.2 average on a 4.0 point scale

Assessment Method Category:
Project/Model/Invention
Oral presentation in capstone course attended by the Digital

Media Software Engineering staff, rated on an established rubric
by the Digital Media Software Engineering Coordinator
3.2 average on a 4.0-point scale

Assessment Method Category:
Presentation(Oral)

Outcome: Development of professional grade skills in Software Development Processes

Students will apply Software Processes to the software development lifecycle, demonstrating an understanding of project planning, metrics, peer
reviews, quality control.

Means of Assessment

Completion of 5 projects that Implement 3 different software
process and engineering best practices
3.2 average on a 4.0 point scale

Assessment Method Category:
Written Product (essay, research paper, journal, newsletter, etc.)
Student knowledge and skills to think effectively and critically are demonstated in 5 software development projects
3.0 average on a 4.0 point scale

Assessment Method Category:
Project/Model/Invention

Outcome: Development of professional grade skills in Software Development Methodologies

Students will identify, analyze and implement Software Methodologies within the framework of the software development lifecycle, project planning.

Means of Assessment

Completion of 5 projects that Implement 3 different software
methodologies
3.2 average on a 4.0 point scale

Assessment Method Category:
Written Product (essay, research paper, journal, newsletter, etc.)
Student demonstration of understanding and competency in conveying the appropriate implementations of software methodologies in capstone project assignments
Project completion with a 3.2 average on a 4.0 point scale

Assessment Method Category:
Presentation(Oral)

Outcome: Development of professional grade skills in Software Quality Assurance

Students will identify, analyze and implement Software Quality Assurance techniques, processes, and best practices within the framework of the
software development lifecycle.

Means of Assessment

Student demonstration of knowledge and skills utilizing SQA
techniques in 5 projects including capstone project assignments
Project completion with a 3.0 average on a 4.0 point scale

Assessment Method Category:
Project/Model/Invention
Student demonstration of the design and implementation of SQA

procedures and processes on assigned projects, including the capstone project
Project completion with a 3.0 average on a 4.0 point scale

Assessment Method Category:
Project/Model/Invention

Outcome: Development of professional grade skills in Software Analysis and Design

Students demonstration of the analysis and design required for software application implementations

Means of Assessment

Student demonstration of the knowledge and skills required to elicit and analyze requirements as part of the software development lifecycle in 5 projects
Project completion with an average of 3.0 on a 4.0 point scale

Assessment Method Category:
Project/Model/Invention
Student demonstration of the knowledge and skills required to write software design specifications from elicited and analyzed requirements as part of the software development in 5 projects
3.2 average on assigned projects

Assessment Method Category:
Project/Model/Invention

Outcome: Development of Industry Readiness

Student demonstration of the ability to think effectively, critically, and efficiently coupled with industry readiness

Means of Assessment

Administer and review intern interviews with students and sponsoring organization using an accepted performance rubric. Rubric results must rate a minimum of satisfactory performance.

Assessment Method Category:
Internship Evaluation

Means of Assessment

Student demonstration of professional techniques of oral and graphic presentations
3.2 out of 4.0 on projects that have presentation elements, including the capstone project
Yes

Assessment Method Category:
Project/Model/Invention

Course Assessment

Course Assessment Plan

Ferris State University

Z - SENG Courses

Z - SENG Courses

Course Outcome: SENG 101 - Computer Programming 1: Application Design (Created By Z - SENG Courses)

Students will design software applications from user requirements, using proven Software Engineering design techniques

Course Outcome Status: Active

Means of Assessment

Assessment Method	Criterion for Success	Assessment Schedule	Active
Students take assessments that illustrate their ability to solve problem domains using Software Engineering design techniques **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	Success = minimum of 80% completion of each assessment method		Yes
Students will present their Project Requirements Document in Class **Assessment Method Category:** Presentation(Oral)	80% of the Project requirements are met		Yes

Course Outcome: SENG 101 - Computer Programming 1: Application Development (Created By Z - SENG Courses)

Students will develop and implement software applications from user requirements, using proven Software Engineering development tools and techniques

Means of Assessment

Assessment Method	Criterion for Success	Assessment Schedule	Active
Students are assessed using small projects that allow them to solve problem domains using Software Engineering development tools and techniques **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of each assessment method		Yes

Course Outcome: SENG 101 - Computer Programming 1: Application Analysis (Created By Z - SENG Courses)

Students will demonstrate the ability to analyze user requirements, using proven Software Engineering modeling and analysis techniques

Course Outcome Status: Active

Means of Assessment

Assessment Method	Criterion for Success	Assessment Schedule	Active
Students are assessed using quizzes, test their ability to solve problem domains using proven Software Engineering modeling and analysis techniques **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	Success = minimum of 80% completion of each assessment method		Yes

Course Outcome: SENG 102 - Computer Programming 2: Application Design (Created By Z - SENG Courses)

Students will design software applications from user requirements, using proven Software Engineering Object-Oriented design techniques

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using quizzes, tests that demonstrate their ability to solve problem domains using Software Engineering design techniques **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	Success = minimum of 80% completion of each assessment method		Yes

Course Outcome: SENG 102 - Computer Programming 2: Application Development (Created By Z - SENG Courses)

Students will develop and implement software applications from user requirements, using proven Software Engineering development tools and Object Oriented techniques

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small projects that demonstrate their ability to solve problem domains using Software Engineering development tools and techniques **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of each assessment method		Yes

Course Outcome: SENG 102 - Computer Programming 2: Application Analysis (Created By Z - SENG Courses)

Students will demonstrate the ability to analyze user requirements, using proven Software Engineering Object-Oriented modeling and analysis techniques

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using quizzes, tests that demonstrate their ability to solve problem domains using Software Engineering design techniques **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	Success = minimum of 80% completion of each assessment method		Yes

Course Outcome: SENG 160 - SENG Methodologies - Processes: Software Methodologies (Created By Z - SENG Courses)

Students can effectively apply the appropriate Software Methodology to a problem domain.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are given assessments to solve problem domains using a Software Methodology **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	Success = minimum of 80% completion of each assessment method	Semester	Yes

Course Outcome: SENG 160 - SENG Methodologies - Processes: Software Processes (Created By Z - SENG Courses)

Students can effectively apply learned Software Processes to a problem domain.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using projects that allow them to solve problem domains using a Software Process **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of each assessment method	Semester	Yes

Course Outcome: SENG 160 - SENG Methodologies - Processes: Software Process Evolution (Created By Z - SENG Courses)

Students can effectively develop Software Processes to solve a problem domain.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using Software Process projects to solve problem domains **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of each assessment method	Semester	Yes

Course Outcome: SENG 170 - Software Requirements Mgmt: Software Requirement Elicitation (Created By Z - SENG Courses)

Students will demonstrate Software Requirement Elicitation using industry standard techniques

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small team projects that demonstrate their ability to obtain user requirements using industry standard techniques **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to Elicitation **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	80% or better		Yes

Course Outcome: SENG 170 - Software Requirements Mgmt: Software Requirement Management (Created By Z - SENG

Students will demonstrate an understanding of Software Requirement Management Principles and Practices

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small projects to demonstrate their understanding of Software Requirement Management Principles and Practices **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to Requirements Management **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	80% or better		Yes

Course Outcome: SENG 170 - Software Requirements Mgmt: Software Requirement Specification (Created By Z - SENG

Students will demonstrate understanding and appreciation for the necessity of the Software Requirement Specifications

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small team projects that demonstrate their ability to analyze user requirements using industry standard techniques to create a Software Requirement Specification **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to the **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	min 80%		Yes

Course Outcome: SENG 190 - Special Topics in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Written Product (essay, research paper, journal, newsletter, etc.)	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 197 - Special Studies in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 200 - Software Packaging-Deployment: CLASS REMOVED FROM PROGRAM (Created By Z - SENG

CLASS REMOVED FROM PROGRAM

Course Outcome: SENG 210 - Software Configuration Mgmt: Software Configuration Analysis (Created By Z - SENG Courses)

Students can analyze an application domain using industry standard techniques to create a Software Configuration Management Plan

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small team projects that demonstrate their ability to analyze an application and create a SCM plan **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 210 - Software Configuration Mgmt: Software Configuration Tools (Created By Z - SENG Courses)

Students can develop and apply a SCM plan to an application domain using industry standard techniques.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small team projects that demonstrate their ability to develop and apply a SCM plan using industry accepted tools **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 210 - Software Configuration Mgmt: Software Configuration Relevance (Created By Z - SENG

Students can correctly define the role of configuration management and the value to the software project.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students will demonstrate understanding and appreciation for the necessity of the SCM in the form of essays and assignments **Assessment Method Category:** Written Product (essay, research paper, journal, newsletter, etc.)	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to Software Configuration Management **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	min 80%		Yes

Course Outcome: SENG 290 - Special Topics in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 297 - Special Studies in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 300 - Software Data Structures: Data Structures and Algorithm Analysis (Created By Z - SENG

Students will demonstrate the ability to analyze complexity of algorithms, both sequential and recursive.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using projects and assignments that demonstrate their ability to analyze the complexity of Data Structures and Algorithms **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 300 - Software Data Structures: Data Structures and Algorithm Design (Created By Z - SENG

Students will demonstrate the ability to design complexity of algorithms, both sequential and recursive.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using projects and assignments that demonstrate their ability to design Data Structures and Algorithms **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 300 - Software Data Structures: Data Structures and Algorithm Development (Created By Z - SENG Courses)

Students will demonstrate the ability to develop complexity of algorithms, both sequential and recursive.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using projects and assignments that demonstrate their ability to develop the complex of Data Structures and Algorithms that solve problem domains **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to Data Structures and Algorithm Development **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	min 80%		Yes

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**

Course Outcome: SENG 301 - Programming Languages: CLASS REMOVED FROM PROGRAM (Created By Z - SENG Courses)

CLASS REMOVED FROM PROGRAM

Course Outcome: SENG 302 - Software Quality Assurance: Software Quality Assurance Principles (Created By Z - SENG

Students can apply the basic principles of software quality assurance

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using quizzes, tests, and assignments that allow them to apply SQA principles to solve problem **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to the basic principles of software quality assurance **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post			Yes

Course Outcome: SENG 302 - Software Quality Assurance: Software Quality Assurance Metrics (Created By Z - SENG

Students can create unit and system tests that integrate into the software development process.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small projects that demonstrate their ability to develop and gather metrics in problem domains using Software Engineering development tools and techniques **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 302 - Software Quality Assurance: Software Quality Assurance Tests (Created By Z - SENG Courses)

Students can create test plans that exercise the software application boundaries.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small projects that demonstrate their ability to create test plans that exercise application boundaries **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 310 - Application Program Interfaces: CLASS REMOVED (Created By Z - SENG Courses)

CLASS REMOVED

Course Outcome: SENG 350 - Software Design - Architecture: Software Architecture (Created By Z - SENG Courses)

Students will demonstrate knowledge of software architecture and their application in software development

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using quizzes, tests that demonstrate their ability to discuss and synthesis solutions in the problem domain as they relate to software architecture and design **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to Software Architecture Design and Development **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	min 80%		Yes

Course Outcome: SENG 350 - Software Design - Architecture: Software Design Fundamentals (Created By Z - SENG Courses)

Students can demonstrate Design Principles to solve architecture and design issues

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using quizzes, tests, and assignments that demonstrate their ability to solve architecture and design issues **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 350 - Software Design - Architecture: Software Design Analysis (Created By Z - SENG Courses)

Students can review and evaluate the design of software architectures

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small team projects that demonstrate their ability to analyze software architectures and to create software frameworks **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 390 - Special Topics in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 397 - Special Studies in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 400 - Eng Enterprise Software Applic: Enterprise Architecture (Created By Z - SENG Courses)

Students will demonstrate basic knowledge of the .NET and Enterprise Architecture

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small projects that demonstrate their ability to solve problem domains using .NET to create application frameworks that are based on Enterprise Architecture fundamentals **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to .NET Enterprise Architecture Design and Implementation **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	min 80%		Yes

Course Outcome: SENG 400 - Eng Enterprise Software Applic: Enterprise Project Development (Created By Z - SENG

Student will prepare appropriate documentation for a complex project

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small projects that demonstrate their ability to prepare appropriate documentation for a complex project that solves problem domains **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 400 - Eng Enterprise Software Applic: Enterprise Software Management (Created By Z - SENG

Students will manage project resources, risks, and contingency plans

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small team projects that demonstrate their ability to apply Enterprise Software Management techniques to solve complex software problems **Assessment Method Category:**	Success = minimum of 80% completion of assessment method(s)		Yes

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Project/Model/Invention			
Students are assessed using tests and quizzes related to Enterprise Software Management	min 80%		Yes

Course Outcome: SENG 420 - Software Develop Industry Cert: Software Industry Certification (Created By Z - SENG Courses)

Student is Certified

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Certification Exam(s) **Assessment Method Category:** Test - External - Post or Pre/Post	Certification Exam defined		Yes

Course Outcome: SENG 430 - Program Graphical Interfaces: GUI Design (Created By Z - SENG Courses)

Students will successfully apply a variety of GUI design processes and techniques.

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using quizzes, tests, and assignments that demonstrate their ability to successfully apply a variety of GUI design processes and techniques to meet the needs of the **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes
Students are assessed using tests and quizzes related to the Design, Analysis, and Implementation of GUIs **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post	min 80%		Yes

Course Outcome: SENG 430 - Program Graphical Interfaces: GUI Analysis (Created By Z - SENG Courses)

Students can analyze GUI experiences and develop solutions by observing users and their challenges

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using quizzes, tests, and assignments that demonstrate their ability to successfully analyze a variety of GUI design processes and techniques **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 430 - Program Graphical Interfaces: GUI Development (Created By Z - SENG Courses)

Students will demonstrate GUI designs that accommodate the diversity of user requirements

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using small projects that demonstrate their ability to solve GUI design issues that meet user requirements **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 490 - Special Topics in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 491 - SENG Applied Internship: Internship (Created By Z - SENG Courses)

Students will apply academic knowledge to real world problems and will learn industry best practices

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Employers will complete an Internship Evaluation **Assessment Method Category:** Internship Evaluation	Successfully fullfilling employers and instructor documented expectations		Yes

Course Outcome: SENG 497 - Special Studies in SENG: Relevant Software Engineering Topic (Created By Z - SENG Courses)

Ability to discuss and synthesize related material

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcome: SENG 499 - Capstone in Applied SENG: Application Portfolio (Created By Z - SENG Courses)

Students will demonstrate through an application portfolio, industry qualified applications.

Course Outcome Status: Active

Course Outcome: SENG 499 - Capstone in Applied SENG: Production Process (Created By Z - SENG Courses)

Students will demonstrate during the completion of the Capstone project the steps necessary to complete a successful software engineering project

Course Outcome Status: Active

Means of Assessment			
Assessment Method	**Criterion for Success**	**Assessment Schedule**	**Active**
Students are assessed using a large and complex application that demonstrates their ability to complete the entire software application lifecycle on schedule. **Assessment Method Category:** Project/Model/Invention	Success = minimum of 80% completion of assessment method(s)		Yes

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Z - SENG Courses - SENG 101 - Computer Programming 1 - Application Design - Students will design software applications from user requirements, using proven Software Engineering design techniques (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students take assessments that illustrate their ability to solve problem domains using Software Engineering design techniques **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** Success = minimum of 80% completion of each assessment method
	Assessment Method: Students will present their Project Requirements Document in Class **Assessment Method Category:** Presentation(Oral) **Criterion for Success:** 80% of the Project requirements are met
Z - SENG Courses - SENG 101 - Computer Programming 1 - Application Development - Students will develop and implement software applications from user requirements, using proven Software Engineering development tools and techniques (Created By Z - SENG Courses)	**Assessment Method:** Students are assessed using small projects that allow them to solve problem domains using Software Engineering development tools and techniques **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of each assessment method
Z - SENG Courses - SENG 101 - Computer Programming 1 - Application Analysis - Students will demonstrate the ability to analyze user requirements, using proven Software Engineering modeling and analysis techniques (Created By Z - SENG Courses)	**Assessment Method:** Students are assessed using quizzes, test their ability to solve problem domains using proven Software Engineering modeling and analysis techniques **Assessment Method Category:**

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Course Outcome Status: Active	Test - Internally Developed - Pre/Post or Post **Criterion for Success:** Success = minimum of 80% completion of each assessment method
Z - SENG Courses - SENG 102 - Computer Programming 2 - Application Design - Students will design software applications from user requirements, using proven Software Engineering Object-Oriented design techniques (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using quizzes, tests that demonstrate their ability to solve problem domains using Software Engineering design techniques **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** Success = minimum of 80% completion of each assessment method
Z - SENG Courses - SENG 102 - Computer Programming 2 - Application Development - Students will develop and implement software applications from user requirements, using proven Software Engineering development tools and Object Oriented techniques (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small projects that demonstrate their ability to solve problem domains using Software Engineering development tools and techniques **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of each assessment method
Z - SENG Courses - SENG 102 - Computer Programming 2 - Application Analysis - Students will demonstrate the ability to analyze user requirements, using proven Software Engineering Object-Oriented modeling and analysis techniques (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using quizzes, tests that demonstrate their ability to solve problem domains using Software Engineering design techniques **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** Success = minimum of 80% completion of

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
	each assessment method
Z - SENG Courses - SENG 160 - SENG Methodologies - Processes - Software Methodologies - Students can effectively apply the appropriate Software Methodology to a problem domain. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are given assessments to solve problem domains using a Software Methodology **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** Success = minimum of 80% completion of each assessment method
Z - SENG Courses - SENG 160 - SENG Methodologies - Processes - Software Processes - Students can effectively apply learned Software Processes to a problem domain. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using projects that allow them to solve problem domains using a Software Process **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of each assessment method
Z - SENG Courses - SENG 160 - SENG Methodologies - Processes - Software Process Evolution - Students can effectively develop Software Processes to solve a problem domain. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using Software Process projects to solve problem domains **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of each assessment method
Z - SENG Courses - SENG 170 - Software Requirements Mgmt - Software Requirement Elicitation - Students will demonstrate Software Requirement Elicitation using industry standard techniques (Created By Z - SENG Courses)	**Assessment Method:** Students are assessed using small team projects that demonstrate their ability to obtain user requirements using industry standard techniques **Assessment Method Category:** Project/Model/Invention

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Course Outcome Status: Active	**Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to Elicitation **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** 80% or better
Z - SENG Courses - SENG 170 - Software Requirements Mgmt - Software Requirement Management - Students will demonstrate an understanding of Software Requirement Management Principles and Practices (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small projects to demonstrate their understanding of Software Requirement Management Principles and Practices **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to Requirements Management **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** 80% or better
Z - SENG Courses - SENG 170 - Software Requirements Mgmt - Software Requirement Specification - Students will demonstrate understanding and appreciation for the necessity of the Software Requirement Specifications (Created By Z - SENG Courses)	**Assessment Method:** Students are assessed using small team projects that demonstrate their ability to analyze user requirements using industry standard techniques to create a Software Requirement Specification **Assessment Method Category:** Project/Model/Invention

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Course Outcome Status: Active	**Criterion for Success:** Success = minimum of 80% completion of assessment method(s) **Assessment Method:** Students are assessed using tests and quizzes related to the SRS **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** min 80%
Z - SENG Courses - SENG 190 - Special Topics in SENG - Relevant Software Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Written Product (essay, research paper, journal, newsletter, etc.) **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 197 - Special Studies in SENG - Relevant Software Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 200 - Software Packaging-Deployment - CLASS REMOVED FROM PROGRAM - CLASS REMOVED FROM PROGRAM (Created By Z - SENG	

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Configuration Mgmt - Software Configuration Analysis - Students can analyze an application domain using industry standard techniques to create a Software Configuration Management Plan (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small team projects that demonstrate their ability to analyze an application and create a SCM plan **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 210 - Software Configuration Mgmt - Software Configuration Tools - Students can develop and apply a SCM plan to an application domain using industry standard techniques. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small team projects that demonstrate their ability to develop and apply a SCM plan using industry accepted tools **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 210 - Software Configuration Mgmt - Software Configuration Relevance - Students can correctly define the role of configuration management and the value to the software project. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students will demonstrate understanding and appreciation for the necessity of the SCM in the form of essays and assignments **Assessment Method Category:** Written Product (essay, research paper, journal, newsletter, etc.) **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to Software Configuration Management **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:**

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
	min 80%
Z - SENG Courses - SENG 290 - Special Topics in SENG - Relevant Software Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 297 - Special Studies in SENG - Relevant Software Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 300 - Software Data Structures - Data Structures and Algorithm Analysis - Students will demonstrate the ability to analyze complexity of algorithms, both sequential and recursive. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using projects and assignments that demonstrate their ability to analyze the complexity of Data Structures and Algorithms **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 300 - Software Data Structures - Data Structures and Algorithm Design - Students will demonstrate the ability to design complexity of algorithms, both sequential and recursive. (Created By Z - SENG	**Assessment Method:** Students are assessed using projects and assignments that demonstrate their ability to design Data Structures and Algorithms **Assessment Method Category:** Project/Model/Invention

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Courses) **Course Outcome Status:** Active	**Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 300 - Software Data Structures - Data Structures and Algorithm Development - Students will demonstrate the ability to develop complexity of algorithms, both sequential and recursive. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using projects and assignments that demonstrate their ability to develop the complex of Data Structures and Algorithms that solve problem domains **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to Data Structures and Algorithm Development **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** min 80%
Z - SENG Courses - SENG 301 - Programming Languages - CLASS REMOVED FROM PROGRAM - CLASS REMOVED FROM PROGRAM (Created By	
Z - SENG Courses - SENG 302 - Software Quality Assurance - Software Quality Assurance Principles - Students can apply the basic principles of software quality assurance (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using quizzes, tests, and assignments that allow them to apply SQA principles to solve problem domains **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
	quizzes related to the basic principles of software quality assurance **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post
Z - SENG Courses - SENG 302 - Software Quality Assurance - Software Quality Assurance Metrics - Students can create unit and system tests that integrate into the software development process. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small projects that demonstrate their ability to develop and gather metrics in problem domains using Software Engineering development tools and techniques **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 302 - Software Quality Assurance - Software Quality Assurance Tests - Students can create test plans that exercise the software application boundaries. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small projects that demonstrate their ability to create test plans that exercise application boundaries **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 310 - Application Program Interfaces - CLASS REMOVED - CLASS REMOVED (Created By Z - SENG Courses)	
Z - SENG Courses - SENG 350 - Software Design - Architecture - Software Architecture - Students will demonstrate knowledge of software architecture and their application in software development (Created By Z - SENG Courses)	**Assessment Method:** Students are assessed using quizzes, tests that demonstrate their ability to discuss and synthesis solutions in the problem domain as they relate to software

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Course Outcome Status: Active	architecture and design **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to Software Architecture Design and Development **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** min 80%
Z - SENG Courses - SENG 350 - Software Design - Architecture - Software Design Fundamentals - Students can demonstrate Design Principles to solve architecture and design issues (Created By Z - SENG **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using quizzes, tests, and assignments that demonstrate their ability to solve architecture and design issues **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 350 - Software Design - Architecture - Software Design Analysis - Students can review and evaluate the design of software architectures (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small team projects that demonstrate their ability to analyze software architectures and to create software frameworks **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 397 - Special Studies in SENG - Relevant Software Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 400 - Eng Enterprise Software Applic - Enterprise Architecture - Students will demonstrate basic knowledge of the .NET and Enterprise Architecture (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small projects that demonstrate their ability to solve problem domains using .NET to create application frameworks that are based on Enterprise Architecture fundamentals **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to .NET Enterprise Architecture Design and Implementation **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** min 80%

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Z - SENG Courses - SENG 400 - Eng Enterprise Software Applic - Enterprise Project Development - Student will prepare appropriate documentation for a complex project (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small projects that demonstrate their ability to prepare appropriate documentation for a complex project that solves problem domains **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 400 - Eng Enterprise Software Applic - Enterprise Software Management - Students will manage project resources, risks, and contingency plans (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small team projects that demonstrate their ability to apply Enterprise Software Management techniques to solve complex software problems **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to Enterprise Software Management **Criterion for Success:** min 80%
Z - SENG Courses - SENG 420 - Software Develop Industry Cert - Software Industry Certification - Student is Certified (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Certification Exam(s) **Assessment Method Category:** Test - External - Post or Pre/Post **Criterion for Success:** Certification Exam defined

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
will successfully apply a variety of GUI design processes and techniques. (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using quizzes, tests, and assignments that demonstrate their ability to successfully apply a variety of GUI design processes and techniques to meet the needs of the user **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
	Assessment Method: Students are assessed using tests and quizzes related to the Design, Analysis, and Implementation of GUIs **Assessment Method Category:** Test - Internally Developed - Pre/Post or Post **Criterion for Success:** min 80%
Z - SENG Courses - SENG 430 - Program Graphical Interfaces - GUI Analysis - Students can analyze GUI experiences and develop solutions by observing users and their challenges (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using quizzes, tests, and assignments that demonstrate their ability to successfully analyze a variety of GUI design processes and techniques **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 430 - Program Graphical Interfaces - GUI Development - Students will demonstrate GUI designs that accommodate the diversity of user requirements (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using small projects that demonstrate their ability to solve GUI design issues that meet user requirements **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
	assessment method(s)
Z - SENG Courses - SENG 490 - Special Topics in SENG - Relevant Software Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 491 - SENG Applied Internship - Internship - Students will apply academic knowledge to real world problems and will learn industry best practices (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Employers will complete an Internship Evaluation **Assessment Method Category:** Internship Evaluation **Criterion for Success:** Successfully fullfilling employers and instructor documented expectations
Z - SENG Courses - SENG 497 - Special Studies in SENG - Relevant Software Engineering Topic - Ability to discuss and synthesize related material (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using research papers that demonstrate their understanding of the selected topics **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)
Z - SENG Courses - SENG 499 - Capstone in Applied SENG - Application Portfolio - Students will demonstrate through an application portfolio, industry qualified applications. (Created By Z - SENG **Course Outcome Status:** Active	

Course Outcomes	Means of Assessment & Criteria for Success / Tasks
Z - SENG Courses - SENG 499 - Capstone in Applied SENG - Production Process - Students will demonstrate during the completion of the Capstone project the steps necessary to complete a successful software engineering project (Created By Z - SENG Courses) **Course Outcome Status:** Active	**Assessment Method:** Students are assessed using a large and complex application that demonstrates their ability to complete the entire software application lifecycle on schedule. **Assessment Method Category:** Project/Model/Invention **Criterion for Success:** Success = minimum of 80% completion of assessment method(s)

Appendix F - Resumes

Frederick Baker

5270 Londonderry Drive - Kentwood, Michigan 49508

Education

- BS in Electrical Engineering: Grand Valley State University, Grand Rapids, Michigan
- Minor in Computer Science: Grand Valley State University, Grand Rapids, Michigan
- MS in Software Engineering Management: Grand Valley State University, Grand Rapids, MI
- 2 years of study in Artificial Intelligence and Machine Learning at Wayne State University, Detroit, MI
- Planning to pursuing Doctorate in Community College Leadership

Professional Profile

- Performed all the responsibilities of a University Program Director in a Software Engineering Department
- Twenty two years of experience designing and developing software from concept to completion
- Expertise using .Net Framework, C#, VB.Net, VB6, ASP, ASP.Net, XML, HTML, SQL
- Experience using: Blackboard, Lotus Notes, Novell, PeopleSoft, Wireless Network Technology
- Software Development and configuration tools expertise (.NET,MS Visual Studio, Source Safe, nUnit)
- Proficiency with project management tools, including Microsoft Project

Professional Development

- Effective Project Management Certificate
- Business Communication Skills Certificate
- Microsoft. Net C# Certificate (5 courses)
- Microsoft .Net Visual Basic Certificate (5 courses)
- Personal Software Process(PSP) Graduate
- Phillip Crosby Software Quality College Certificate
- Robust Software Engineering Certificate
- Ford Power Presentations Graduate
- Dale Carnegie Course: Effective Communications & Human Relations

Professional Accomplishments

- Consistently maintained an average of 4.6/5.0 of better on student evaluations
- Patent submission for high tech security detection device
- Received several awards for outstanding contributions in the software systems division
- Received an award from peers for outstanding contributions in software systems department
- Received several Dale Carnegie awards

Activities

- Software Engineering Advisory Board Chair
- Mentor and Tutor for Kent Intermediate School District
- Wayne County Community College Board Member
- Art Institute of Michigan Advisory Board Member
- AimWest Member
- FIRST Robotics Competition Judge

Professional Experience

Ferris State University. Grand Rapids, Michigan

Ferris State University prepares students for successful careers, responsible citizenship, and lifelong learning.

Program Director and Instructor (2007 - present)

- Developed (all instructor material) and Implemented twelve Software Engineer Courses
- Developed and Executed Marketing and Advertising Campaigns
- Developed department tactical and strategic plans
- Responsible for student recruiting and retention
- Software Engineering Principle Instructor
- Software Engineering Advisory Board Chair
- Academic Advisory Diversity Committee Member
- Performed the responsibilities of an Academic Advisor
- Established Internship relationships with successful local and national Engineering Companies
- Initiated, Implemented, and taught the Digital Media Software Summer Camp Academy
- Keynote Speaker at Technology Summits, Universities, Community Colleges, and High Schools
- Initiated and Collaborated on many Technological Grants, focused on community outreach programs

Harris Inc. Grand Rapids, Michigan

A leader in the development and delivery of communications and information technology,

serving government and commercial markets in more than 150 countries.

Software Design Engineer (2000 - 2007)

- Designed, Developed, Deployed the System Interface for enterprise Traffic and Billing Software
- Developed, Implemented, Deployed, Maintained the enterprise Accounts Receivable Software
- Designed, Developed, Implemented, and Deployed the Enhanced Network Programming application. The design of this application replaced the legacy application and improved input efficiency by over 400%.
- Designed, Developed, Implemented, Deployed, and provided support for 10+ core system applications, utilizing technologies on multiple-platforms incorporating several computer languages and disciplines.

Gordon Food Service (GFS) Grand Rapids, Michigan

GFS is the largest, privately held food service distributor in North America.

Software Development Engineer (1999 - 2000)

- Maintained and Enhanced the Inventory Management/Order software used daily by thousands of Clients.
- Designed, Developed, Deployed the Product Inventory Acquisition, Control, and Management software embedded in a Handheld Barcode Scanner responsible for capturing all field technician Inventory data.
- Designed, Developed, Deployed the Product Inventory Acquisition, Control, and Management Graphical User Interface software for the desktop PC, which synchronized and managed communication between a Handheld Barcode scanner and an IBM AS/400 Mainframe.

Ford Motor Company Dearborn, Michigan

Ford Motor Company, a global automotive industry leader that manufactures and distributes automobiles.

Application Development Engineer (1996 - 1999)

- Designed, Developed, Deployed and Maintained the communication software that performed exhaustive testing of automotive modules utilizing object-oriented methodology to communicate and exercise multiple modules using multiple proprietary communication protocols.

- Designed, Developed, and Implemented software that simultaneously controlled and monitored environmental chambers, multiple power supplies, multiplexers, and embedded controllers. The software provided a flexible, intuitive, and informative real-time activity display, which enabled the tester to respond appropriately to any malfunctions during the testing process.

Assessment and Control International(ACI) Bloomfield Hills, Michigan

An Industry leader in the manufacturing and distribution of High-Tech perimeter security systems world wide.

Chief Systems Engineer (1992 - 1996)

- Responsible for the Engineering Department Budget, Engineering Service Department, Project Management, Outsourced Consultants, and Strategic Planning

- Managed and Provided leadership and growth opportunities for engineers that were direct-reports

- Managed, Designed, Developed, Deployed, and Maintained mission critical level security software and hardware that provided emergency response perimeter surveillance for nuclear facilities, Correctional Institutions, and installations controlled by the United States Department of Defense.

Appendix G – Survey Assessments

FSU Digital Media Software Engineering

Alumni Survey

You are now an alumnus of Ferris State University! Congratulations and we hope your hard work and dedication pay personal and professional dividends in the years to come. We'd like to ask you for some of your opinions that speak about the overall DMSE program and your experience at Ferris. Not only does this type of feedback offer us an opportunity to improve the program, but Ferris State University requires this type of information and it is also required by accreditation bodies to demonstrate our efforts for continuous improvement and striving to deliver quality and worthwhile education. Please take some time to provide us with information as to how you see the program now, as well as suggestions for possible improvement in the future.

Program Assessment
1. Please indicate your level of satisfaction with the overall quality of your experience in the DMSE program.
O Very Satisfied
O Somewhat Satisfied
O Somewhat Dissatisfied
O Very Dissatisfied
Comments:

Faculty Assessment
Please indicate your level of satisfaction with the overall quality of the instructors from the DMSE program.
O Very Satisfied
O Somewhat Satisfied
O Somewhat Dissatisfied
O Very Dissatisfied
Comments:

Software Assessment
Please indicate your level of satisfaction with the overall quality and relevance of the software you were exposed to in the DMSE program.
O Very Satisfied
O Somewhat Satisfied
O Somewhat Dissatisfied
O Very Dissatisfied
Comments:

Facility/Hardware Assessment
Please indicate your level of satisfaction with the overall quality and relevance of the hardware (computers) and facility made available to you in the DMSE program.
O Very Satisfied
O Somewhat Satisfied
O Somewhat Dissatisfied
O Very Dissatisfied
Comments:

Curriculum Assessment
Please indicate your level of satisfaction with the overall quality and relevance of the curriculum you were exposed to in the DMSE program.
O Very Satisfied
O Somewhat Satisfied
O Somewhat Dissatisfied
O Very Dissatisfied

Comments:

Please indicate how effective you think the DMSE program was in delivering content and skill building in each of the following areas of study:
Foundation/Traditional Media O
(1)Very Unimportant
(2)Somewhat Unimportant
(3)Neutral
(4)Somewhat Important
(5)Very Important

Please indicate your level of satisfaction with each of the following classes. If you didn't take one, please select "N/A."

Course	Description	(1-5[Very Satisfied])
SENG 100/101	Computer Programming 1	
SENG 102	Computer Programming 2 (SENG 101)	
SENG 300	Software Data Structures (SENG 102)	
SENG 301	Programming Languages (SENG 300)	
SENG 315	Software Componenet Design	
SENG 400	Engineering Enterprise Software Applications (SENG 350)	
SENG 430	Programming Graphical User Interfaces (SENG 350)	
SENG 160	Software Engineering Methodologies and Processes	
SENG 170	Software Requirements Management (SENG 160)	
SENG 210	Software Configuration Management (SENG 160)	
SENG 302	Software Quality Assurance (SENG 160)	
SENG 350	Software Design and Architecture (SENG 160, SENG 300)	
SENG 355	Software Engineering Tools (SENG 160)	
SENG 420	Software Development Industry Certification (SENG 350, SENG 355)	
SENG 491	Applied Internship (SENG 301, SENG 302, SENG 315)	
SENG 499	Capstone in SENG (SENG 491)	

In your opinion, what are the strongest features of the DMSE program?

In your opinion, what are the areas that require the most improvement in the DMSE program? Why?

Employment Feedback

1. How much time has it taken for you to be employed with in the Software Engineering field?
   a. Within six months
   b. Within a year
   c. Within two years
   d. Within three years
   e. I am not working in the field

2. What salary are you currently making?
   a. Under 20K a year
   b. Over 40 K a year

c. Over 60 K a year
d. Over 80 K a year
e. I am not working in the field

3. Were you able to get *part time* work in the Software Engineering or media field within a year of graduation?
    a. Yes
    b. No
4. Are you still employed in this field?

    a.Yes

    b.No

5. Geographically, where are you working?
   (create a type in field here)

6. Have you gone on for additional training?
    a. Yes
    b. No

7. If you went on for more training, where did you get it?
   (create a type in field here)

Is there anything else you'd like to comment on that we didn't ask about (Additional Comments)?

Thank you for your input.
We will use this information to refine the DMSE program and strengthen it in many aspects.

Digital Media Software Engineering

Employer Evaluation Survey

Please rate your Employee using the following guide.

1	**Poor**	(**Never** demonstrates this ability/does not meet expectations)
2	**Unsatisfactory**	(**Seldom** demonstrates this ability/rarely meets expectations)
3	**Fair**	(**Sometimes** demonstrates this ability/meets expectations)
4	**Satisfactory**	(**Usually** demonstrates this ability/sometimes exceeds expectations)
5	**Exceptional**	(**Always** demonstrates this ability/consistently exceeds expectations)

If any criteria are not applicable to this internship experience, please leave the response blank.

A. Ability to Learn

1. Asks pertinent and purposeful questions 1 2 3 4 5

2. Seeks out and utilizes appropriate resources 1 2 3 4 5

3. Accepts responsibility for mistakes and learns from experiences 1 2 3 4 5

B. Reading/Writing/Computation Skills

1. Reads/comprehends/follows written materials 1 2 3 4 5

2. Communicates ideas and concepts clearly in writing 1 2 3 4 5

3. Works with mathematical procedures appropriate to the job 1 2 3 4 5

C. Listening & Oral Communication Skills

1. Listens to others in an active and attentive manner 1 2 3 4 5

2. Effectively participates in meetings or group settings 1 2 3 4 5

3. Demonstrates effective verbal communication skills 1 2 3 4 5

D. Creative Thinking & Problem Solving Skills

1. Breaks down complex tasks/problems into manageable pieces 1 2 3 4 5

2. Brainstorms/develops options and ideas 1 2 3 4 5

3. Demonstrates an analytical capacity 1 2 3 4 5

E. Professional & Career Development Skills

1. Exhibits self-motivated approach to work 1 2 3 4 5

2. Demonstrates ability to set appropriate priorities/goals 1 2 3 4 5

3. Exhibits professional behavior and attitude 1 2 3 4 5

4. Able to apply Software Processes and Tools effectively 1 2 3 4 5

F. Interpersonal & Teamwork Skills

1. Manages and resolves conflict in an effective manner 1 2 3 4 5

2. Supports and contributes to a team atmosphere 1 2 3 4 5

3. Demonstrates assertive but appropriate behavior 1 2 3 4 5

G. Organizational Effectiveness Skills

1. Seeks to understand and support the organization's mission/goals 1 2 3 4 5

2. Fits in with the norms and expectations of the organization 1 2 3 4 5

3. Works within appropriate authority and decision-making channels 1 2 3 4 5

H. Basic Work Habits

1. Reports to work as scheduled and on-time 1 2 3 4 5

2. Exhibits a positive and constructive attitude 1 2 3 4 5

3. Dress and appearance are appropriate for this organization 1 2 3 4 5

I. Character Attributes

1. Brings a sense of values and integrity to the job	1 2 3 4 5
2. Behaves in an ethical manner	1 2 3 4 5
3. Respects the diversity (religious/cultural/ethnic) of co-workers	1 2 3 4 5

J. Open Category: Industry-Specific Skills

Are there any skills or competencies that you feel are important to the profession or career-field (represented by your organization) that have not been previously listed in this evaluation? If so, please list these skills below and assess the intern accordingly.

(comment field)

K. Other Comments:

(comment field)

L. Overall Performance (if I were to rate the Ferris DAGD alumni at the present time) 1 2 3 4 5

Employer Survey

Digital Media Software Engineering (DMSE) at Ferris State University, Grand Rapids MI. -

Ferris State University's Digital Media Software Engineering Program (DMSE) was instituted in 2007. We are now undergoing a process all academic programs at Ferris do at six year intervals named 'academic program review'. In it, we look at what we have achieved, where we are and what should be improved based on feedback from DMSE Graduate employers. As an employer of one of our alumni, we would appreciate your feedback.

We welcome your insights on how the program should evolve to meet the professional world.

Not only does this type of feedback offer us an opportunity to improve the program, Ferris State University requires this type of information. Additionally it is required by accreditation bodies to demonstrate our efforts for continuous improvement and striving to deliver quality and worthwhile education. Please take some time to provide us with information from a professional/real world perspective.

For a review of our work and more information, you can visit our website at http://DMSE.ferris.edu.

1. **Please indicate how important you think each of the following skills are:**

Skill / Key Knowledge Item	*Unimportant*	*Somewhat Unimportant*	*Somewhat Important*	*Very Important*
Foundation/Traditional Engineering				
Project Management				
Collaboration				
UML				
Database Design				
Innovation				

Professional Development				
Multiple Programming Languages				
Project Management				
Communication				
General Education				
Other				

Please specify any other skillsets you believe are important.

7. **If you have had one of our students work for you, Please indicate how effective you think the DMSE program was in delivering content and skill building in the following areas of study:**

Skill / Key Knowledge Item	*Unimportant*	*Somewhat Unimportant*	*Somewhat Important*	*Very Important*
Foundation/Traditional Engineering				
Project Management				
Collaboration				
UML				
Database Design				
Innovation				
Professional Development				
Multiple Programming Languages				

Project Management				
Communication				
General Education				
Other				

Please Specify:

1. Software Assessment

Please rate the overall quality and relevance of the software we use in the Ferris DMSE program.

Software	*Not important At all*	*Somewhat Important*	*Very Important*	*Required Knowledge*	*Unknown*
Visual Studio 2010 Ultimate					
Unity 3D					
Visio					
Unreal					
MS Project					
CryEngine					
Eclipse					

Please Specify:

6. Comments or Software Suggestions:

8. Are there any specific skills that stand out as having been excellent from our work or students?

9. Are there any specific skills or (aspects) that stand out as having been poor from our students?

10. **Program Assessment:** On a scale from poor to excellent, rate the overall quality of the Ferris DMSE program.

    (poor) (good) (very good) (excellent) (undecided)

    Comments:

11. In your opinion, what are the strongest features of the Ferris DMSE program?

12. In your opinion, what are the areas that require the most improvement in the Ferris DMSE program?

13. Is there anything else you'd like to comment on that we didn't ask about?

14. I am a (Check all that applies):
    a. Professional in the Animation industry
    b. Professional in the Software Engineering industry
    c. Professional in Video, Media or Advertising
    d. An employer or past employer of a DMSE alumni
    e. Educator
    f. Architecture
    g. Other

If you have hired a DMSE alumni or been an internship site, please help us out by completing the following.

15. On a 1 – 10 scale with 10 being outstanding, I would rate the DMSE Graduate experience as a:

Thank you for your input. We will use this information to refine the DMSE program and strengthen it in many aspects. If you would like to comment by phone or email, please contact Frederick Baker, Program Director at 616-643-5721 or FrederickBaker@ferris.edu

Exit Survey

You are now (or soon to be) an alumnus of Ferris State University! Congratulations and we hope your hard work and dedication pay personal and professional dividends in the years to come. We'd like to ask you for some of your opinions that speak about the overall DMSE program and your experience at Ferris. Not only does this type of feedback offer us an opportunity to improve the program, Ferris State University require this type of information, and it is also required by accreditation bodies to demonstrate efforts for continuous improvement and striving to deliver quality and worthwhile education. Please take some time to provide us with information as to how you see the program now, as well as suggestions for possible improvement in the future.

1. **Name the three most valuable items you learned in the DMSE program**
   - Software Engineering concepts
   - Documentation, as far as design docs, tech docs, and the like is hugely important, although it may be an odd choice.
   - Programming
   - Collaboration Skills
   - Software Engineering Tools

2. **Name at least two items that were covered extensively in the program that you wish were not covered, because you either:**
   a. **Had previous extensive knowledge**
   b. **Had no need for coverage of this topic**
   -

3. **Were there any particular topics that you wish were covered that were not, or those that needed to be covered more extensively? (name these and comment)**
4. **On a scale from 1 (poor) to 10 (excellent), rate the overall quality of your education in the Ferris DMSE program**

(poor)		(undesirable)			(useful)				(excellent)
1	2	3	4	5	6	7	8	9	10

(8 5 7 6 7 9)

5. **On a scale from 1 (poor) to 10 (excellent), rate the overall quality of instructors from the DMSE program**

(poor)		(undesirable)			(useful)				(excellent)
1	2	3	4	5	6	7	8	9	10

(6 6 7 4 7 9)

6. **From question 4 (instruction), are there any specific instructors that stand out as either excellent or poor? (You can comment on any number of instructors and use the back of this page as necessary)**

7. **Below is a table which contains key items identified as useful and critical for graduates of the DMSE program. The list is presented in alphabetical order. On a scale from 1 (poor) to 10 (outstanding), please rank each of these skills as their importance AND how effective the DMSE program was in delivering content and skill building in this area of emphasis.**

8. **In your opinion, what should change in the coverage of topics in the current program? Why?**
9. **In your opinion, what are the strongest features of the Ferris DMSE program?**
10. **In your opinion, what are the areas that require the most improvement in the Ferris DMSE program? Why?**
11. **Is there anything else you'd like to comment on that we didn't ask about?**

Faculty Perceptions of Program

The DMSE program has had consistent growth, development and transformation in the last four years. The initial curricula were developed with focus on Traditional Software Engineering Methodologies, Processes, and Tools while implementing innovative emerging technologies.

The focus of the program review, we are looking at how we can do things better and develop for the future. Your input will be critical for this development. To maintain relevance and viability, please provide detailed feedback regarding the DMSE program through written response to the questions listed below before December 18th.

DMSE Program feedback:

Please discuss in detail your views on

1. Departmental Administrative Improvements
2. 3 – 5yr Departmental Vision
3. Course modifications
4. Viewpoints on accelerated and Online classes
5. Strategies and Tactics to improve retention
6. Strategies and Tactics to improve cross class collaboration

Advisory Board Survey

Digital Media Software Engineering (DMSE) at Ferris State University, Grand Rapids MI. -

Ferris State University's Digital Media Software Engineering Program (DMSE) was instituted in 2007. We are now undergoing a process that all academic programs at Ferris must complete at six year intervals named 'Academic Program Review'.

We welcome your insights on how the program should evolve to meet the professional world.

Not only does this type of feedback offer us an opportunity to improve the program, Ferris State University requires this type of information. Additionally it is required by accreditation bodies to demonstrate the efforts for continuous improvement and striving to deliver quality and worthwhile education. Please take some time to provide us with information from a professional/real world perspective.

For a review of work and more information, you can visit the website at http://DMSE.ferris.edu.

2. **Please indicate how important you think each of the following skills are:**

Skill / Key Knowledge Item	*Unimportant*	*Somewhat Unimportant*	*Somewhat Important*	*Very Important*
Foundation/Traditional Engineering				
Project Management				
Collaboration				
UML				
Database Design				
Innovation				
Professional Development				
Multiple Programming Languages				

Project Management				
Communication				
General Education				
Other				

3. Please specify any other skill sets you believe are important.

4. **Software Assessment**

Please rate the overall quality and relevance of the software we use in the Ferris DMSE program.

Software	*Not important At all*	*Somewhat Important*	*Very Important*	*Required Knowledge*	*Unknown*
Visual Studio 2010					
Unity 3D					
Visio					
Unreal Development Kit					
MS Project					
CryEngine					
Android Development					
iOS Development					

Please Specify:

4. Comments or Software Suggestions:

5. Are there any specific skills that stand out as having been excellent from our work or students?

6. **Program Assessment:** On a scale from poor to excellent, rate the overall quality of the Ferris DMSE program.

(poor) (good) (very good) (excellent) (undecided)

Comments:

7. In your opinion, what are the strongest features of the Ferris DMSE program?

8. In your opinion, what are the areas that require the most improvement in the Ferris DMSE program?

9. Is there anything else you'd like to comment on that we didn't ask about?

10. I am a (Check all that applies):

h. Professional in the Animation industry
i. Professional in the Software Engineering industry
j. Professional in Video, Media or Advertising
k. An employer or past employer of a DMSE alumni
l. Educator
m. Architecture
n. Other

Thank you for your input. We will use this information to refine the DMSE program and strengthen it in many aspects. If you would like to comment by phone or email, please contact Frederick Baker, Program Director at 616-643-5721 or FrederickBaker@ferris.edu