CMSC 355: Fundamentals of Software Engineering

Spring, 2023

Syllabus

Catalog Listing:	**CMSC 355 – Fundamentals of Software Engineering**
Course Level:	Undergraduate
Prerequisites:	Students must have completed CMSC 256 or EGRE 246 with a grade of C or better
Instructor:	Dr. Robert Dahlberg
Office:	Engineering Building East – Room E4234
E-mail:	dahlbergra@vcu.edu
Classroom:	Engineering Building West - Room 101 Tuesdays & Thursdays 11:00 am - 12:15 pm
Office Hours:	**In person:** Immediately before/after class: 10:30am – 11:00 and 12:15pm – 12:45 Tuesdays and Thursdays **OR via Zoom by appointment**

1.0 – Overview (Catalog Course Description):

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 256 or EGRE 246, either with a minimum grade of C. Provides an overview of the software engineering process and software life-cycle models. Gives a detailed study of the analysis, specification and design phases. Students will work in teams to gain experience in software development methodology, developing specification and design documents and developing a prototype.

2.0 – Course Structure:

- Lecture hours/week – 3
- Lab hours/week – 0

3.0 – Course Goals

Upon successful completion of this course, the student will be able to:

1. Understand the software lifecycle and different software development methodologies
2. Express requirements and design of a software system
3. Work as a team to develop software products using agile software development methodologies
4. Understand code management when working in a team
5. Understand software quality and be able to effectively test software
6. Perform software maintenance and use appropriate tools
7. Use effective software architectures and design patterns

4.0 – ABET Criteria Addressed:

a. Practice the ability to communicate effectively

b. Assess the ability to apply computer science theory

c. Assess the ability to function effectively on teams to accomplish a common goal

5.0 – Major Topics Covered:

- Software development lifecycle
- Framework of the software lifecycle process
- Requirement engineering
- Software architecture and design
- Software modeling via UML
- Software verification and validation
- Agile software development
- Code Management (GitHub)
- Design patterns
- Serverless Coding
- Service Oriented Design Techniques
- Containerization
- Continuous Integration
- Continuous Improvement
- Test Plan Methodologies
- Software Maintenance

6.0 – Textbook(s):

No required textbook. Extensive notes and online reading materials will be provided.

7.0 – Class Schedule:

- **Lecture:** Tuesday/Thursday 11:00pm – 12:15pm in Engineering Building East – E2214

8.0– Evaluation:

General instructions:

There are few individual assignments in this course, but the majority of assignments are team projects. Significant amount of time outside class meetings will likely be required for the successful completion of the projects, including ample time spent communicating and planning with the team. Students will be graded on their individual contribution to the team project, using a combination of peer and individual ratings in the project status reports. Several mechanisms will be used to detect students that fail to contribute significantly to the team project; those students will not get full credit for their team's project(s).

Grading:

Category	***Percentage Weight***
Bi-weekly Quizzes	**30%**
Team Project & teamwork	**50%**
Class participation, attendance, individual assignments and team/individual meetings with instructor	**20%**

Grades will be posted on Canvas in a timely manner, and include appropriate feedback and weights. Students are encouraged to meet with professor, individually or in teams. Zoom meetings are preferred, so we can share documents and code.

Attendance policy:

Attendance is expected from all students in the course. To encourage attendance, the instructor will use periodic, short, in-class exercises, which will be graded as either acceptable or non-acceptable.

Grading scheme:

A:	>= 90%
B:	>= 80% and < 90%
C:	>= 70% and < 80%
D:	>= 60% and < 70%
F:	< 60%

Please consult external resources for VCU policies regarding academic honesty, students with disabilities, student conduct in the classroom, withdrawal from classes, and others.

9.0– Teaching Philosophy:

- The main focus of this professor is to guide and provide students learning opportunities. The professor's function is **not** to assign grades, but to guide student learning. If you learn, you will get a good grade. Learning is the goal in this course, **not** earning a grade.
- All students are unique. Every student responds to their own unique learning style(s). Students need to work with the professor to discover the learning styles(s) that provide the best potential for success (whether or not the student has an accommodation letter from the SAEO). This means that through interaction together, the professor and student need to identify a way to best measure the students learning progress
- Learning is hard work. To learn, a student must first embrace his/her/their lack of knowledge and put forth a lot of effort. Failure prepares us for learning, success gives us self-confidence in what we learn.
- Learning and teaching is a two-way street. Communication between professor and student are critical, if learning is to occur.
- The goal of an undergraduate degree is the get the students to learn how to learn. You should understand what learning style(s) are best for you. And begin to understand how to research concepts and new skills.
- Collaboration, not competition is the key to success. A lot of effort and resources are wasted on competition. Working in a team, and optimizing the skills and talents of teammates will result in superior work. The whole is always greater than the sum of the parts.