CSE 416 Software Engineering

Overview

February 28, 2023

Announcements

- Read the course website:

  https://www3.cs.stonybrook.edu/~alee/cse416/

- Email me survey.txt (see today's lecture notes area)

- Today: Introduction, overview of the course
- Reading: see the course website

- Break around 2:45pm

Acknowledgements

Some of these slides are from Prof. Alex Kuhn.

Course overview

- This is a capstone project-based course
- Teamwork is required – important to contribute
- You will primarily learn by doing – supplemented with lectures, readings, and discussions

Textbooks

- No course textbook
- Some assigned readings will be introduced during the class
- Optional supplement: Ian Sommerville's Software Engineering (10th Edition)
- Optional supplement: Martin Fowler's UML Distilled (3rd Edition).

Course requirements

- Assignments and in-class activities (10%)
- One midterm exam (20%)
- Group project (70%)

- Class participation is expected

Class participation and attendance

- To participate fully, need to be in-class on time
- Can have three unexcused absences without any penalty, assuming no delivery due that day
  - Additional absences may result in a penalty to your grade
  - Missing part of the class may also count as an unexcused absence
- If you have unexcused absences for more than 20% of the classes you will receive a failing grade for the course

Individual SWE presentation

- Everyone will give a 5 minute presentation on a company's software engineering process
  - Your choice on the company (with instructor approval)
  - Will be scheduled throughout the semester

- Pick a company and email the instructor to schedule by March 21

- See assignment handout on shared Google Drive folder for details (individual_SWE_presentation.pdf)

Benefits of this course

- This course can be an opportunity to improve your job prospects
  - Create a portfolio project of your choice
  - Gain experience with modern software engineering practices and technologies
  - Practice working as a team in a professional manner
  - Gain experience presenting your work
- The more energy you put into your project, the more you will benefit

Self-learning is key

- Expected to learn tools and technologies on your own
- Learning new technologies can be challenging
  - Self-learning is critical to your career – technologies gain and lose dominance quickly

Programming by coincidence

- With web development, it can be easy to start “programming by coincidence”
    - You keep making changes until it seems to work

- However, this will not help you in the long run
    - Try to understand the details for why and how your code works
    - You may not fully understand everything now, but keep trying to expand your knowledge
    - Putting extra effort upfront on this will make it far easier later on as things get more complicated

Group project

- Many milestones and assignments, some of which will include presentations
- The tentative breakdown is (which adds up to 70% of your total grade):
  - Create Teams (1%)
  - Requirements Document + Presentation (10%)
  - Data Design and Web Views (8%)
  - Design Document + Presentation (8%)
  - Project Milestone 1 (ungraded check-in)
  - Project Milestone 2 (ungraded check-in)
  - Project Milestone 3 (ungraded check-in)
  - Project Milestone 4: Beta Release + Presentation (8%)
  - Code Review (ungraded feedback)
  - Project Milestone 5 (ungraded check-in)
  - Project Milestone 6 (ungraded check-in)
  - Final Release + Presentation (35%)

Notes on group work

- Everyone is expected to contribute to their team
  - Will have to state own contributions and evaluate yourself and teammates
  - Coding contributions also tracked in GitHub
- If you fail to make expected contributions, can receive a different grade than teammates
- If your teammates are not doing their fair share, or are preventing you from contributing, please discuss with me
  - Teammates not contributing may be removed from the team (with consent of instructor) and need to complete the project on their own

Peer evaluations

- You will send me peer evaluations at points during the semester
  - If there is any issue with your teammates, work to resolve it early on
  - Talk to me if you continue to have issues

- At the end of the semester, you will provide a grade for your teammates and rationale for the grade
  - The peer evaluations you receive will affect your final project grade

Project overview

- Substantial Software Project – typically web application
  - Includes a front-end and back-end + database component
  - Can choose technologies and project (with instructor approval)
    - Must use a back-end framework (e.g., Java Spring, Express.js)
    - Suggested to use a CSS Library or framework for front-end (e.g., Bootstrap)
  - Must be a new project (no re-using your prior projects)

- Work in teams of ~3 students – I will assign teams

Example prior “A” project

- **Problem:**

  The Merchandising Society Club (MSC) makes Fashion Institute of Technology (FIT) and Stony Brook University merchandise. Their goal is to not only produce merchandise for students and faculties but also to showcase and sell their products that are crafted after extensive research on fashion trends and sales planning. Thus, to display their products for sale, MSC runs its own physical Retail Revolution Store at FIT, and they are currently in need of expanding their collection online. Particularly during the COVID 19 crisis, customers cannot regularly access the offline retail store. Thus, having an online means of accessing MSC is a massive benefit for them.

- **Solution:**

  Our solution is to create a flexible website for MSC that will allow them to manage and maintain an online platform that has several different sections which include an MSC about page, mailing subscriptions, an MSC product collection for showcasing merchandise, and a virtual showroom to replicate their physical Retail Revolution Store online.

  See it deployed at: https://merchandising-society.web.app/

Project theme

- Projects this semester should focus on the theme of learning technologies
  - Build a web app targeting specific users trying to learn something
  - Exact specifics up to your group (with instructor approval)

- See “Create Team” Assignment (Create_Team_Assignment.pdf) on shared Google Drive folder for more details and ideas

Project pitch

Sample project idea 1

Parents want their children to learn to play piano, but private lessons are expensive, and especially in the time of Covid, music academies can even be dangerous to attend! I will create a web application that will teach children how to play piano online, using an interactive keyboard and lessons that start with the basics of reading music and the names of notes, all the way to playing songs and eventually being able to play on a real keyboard or piano by providing interactive sheet music online that helps the children once they move past the basics. (e.g., https://pianu.com/)

Sample project idea 2

College students have trouble keeping track of all their notes and documents when taking classes and working on projects. I will create a web application that is a personal note-taking system that enables creating, editing, and categorizing a variety of multimedia documents. These documents can also be shared with peers to collaborate together.
(e.g., a specialized evernote.com)

Sample Project Idea 3

For someone with a busy schedule, it’s hard to start and keep a routine of healthy habits, which are often started and discontinued after a few days. I will create a website that allows you to choose habits you want to develop and it reminds you (via the website or email / text messages) to do the habit at a set time. You can visualize what habits you’re on track with and also sign up for a human coach who will personally work with you and send you messages of encouragement (e.g., coach.me).

Sample Project Idea 4

Young professionals find it hard to keep track of personal finances and budgeting and are unsure how much they should save for the future. I will build a website where you can track your budget and expenses and receive recommendations and projections of your finances for future purchases and retirement (e.g., mint.com).

Project constraints

- Your website should have at least two different users with different views, e.g., standard user + admin
- An “A” project will have substantial functionality and be well-executed. At minimum:
  - Complicated features (e.g., real-time chat, non-trivial algorithms, complex flows or interactions)
  - Should allow users to upload and interact with some form of media or files (e.g., images/audio/video)
  - Must have secure login – I recommend using Google Oauth
  - Website should be responsive and work on both mobile and desktop browsers
- Will need to deploy the website for the beta release (see schedule)

Some advice from prior students

- “Take your time to learn and apply good practices when using Git and GitHub. ***Do you let anyone commit to master? Review your code and checkout working branches and test before merging to master. You will save yourselves from rewriting it and introducing the same bugs countless times”
- “March 13th is the last day you can drop courses without a W. Drop and escape while you can... lol jk But if you think you have a lot of work this semester, consider dropping some of your courses.”
- “There are many aspects to consider when you plan for the primary tasks, more than you expected when writing down your SRS, and the design decision to implement them should be clever for both enhancing the user interaction and the efficiency of system. Don’t stick to the initial plan, but think more flexible to prioritize the quality of the product.”
- “Make plans, know what you are doing, start early.”

Some advice from prior students

- “Please start and make plan ahead. If you are not familiar with your teammates, please gather up, have some nice dinner, or go out and play bowling to get close with each other. A good atmosphere within the team makes it easier to communicate and work together as a team!”
- “Even though the project seems easy and simple when you just think about it (before starting to make the actual project), it gets complicated and takes quite a long time to fix major and minor parts of the code. So, be simple and be sure what application you want to make.”
- “I just want to say that students should manage their time very well to do all the things. And they need to know what they actually can finish in time. We thought that we can finish all of our use cases in time, but we couldn’t.”
- “Your group members are very important and learn to use the frameworks even though the initial upfront cost may be high.”

Introductions

First tasks

1. Create GitHub account & apply for GitHub Education benefits
   - https://education.github.com/discount_requests/new
   - Register with SBU email. On how you plan to use GitHub write “I plan to use GitHub in my Computer Science Software Engineering course to work on a group project with my teammates.”
2. Fill out online intro survey (posted on schedule):
3. Read “A Brief History of Software Engineering” by Amy Ko (posted on schedule)
   - Note down 1-2 points that were interesting or surprising to you for next class
4. Come up with one potential project idea to present next class. You will state the problem and how your proposed system will solve it (~30 second pitch)

Covid class guidelines . . .

- If the instructor gets COVID: this class will go online for a week
- If you have COVID symptoms and/or are confirmed with COVID, tell your instructors immediately – no later than the start of class
    - If confirmed with COVID: you must not attend classes for a week – instructors will provide you with course materials and support
    - If you have COVID symptoms: you must not attend class until you have been tested for COVID and get a negative result

Tips on using Zoom

- Will use Zoom when necessary
- Can send text messages
  - Please feel free to write questions at any time
- Can mute / unmute self
- Just re-join if connection drops off
- Please send me feedback if you have any issues or suggestions