COS318 - Web Programming  
An Industry Perspective  
Fall 2022  
Prerequisite: COS217

Instructor: Steven Yackel Email: yacste@bethel.edu  
T TH 2:05 – 3:20 BAC344 Office Hours: By appointment

Books

* (Optional) Cracking the Coding Interview: 189 Programming Questions and Solutions, Gayle Laakmann McDowell, 2015; ISBN-13: 978-0984782857; ISBN-10: 0984782850

Course Overview

This course covers the rapidly changing field of web programming. Most of the class will cover web programming from the server side, with REST, server architecture, and common web service problem spaces. Client-side programming will be in javascript and the React framework. Other topics that are strongly related to web programming such as queues, dependency injection, and workers will also be covered.   
  
Why this Course is in our Curriculum

While there are still many traditional software platforms and applications in development, more and more companies are moving their technologies to the web with services, web sites, and cloud deployments. It is important for students graduating with a computer science degree to be fluent in web development and the problems that arise when creating web services.

This class’s subtitle is “An Industry Perspective.” There are several things about this class that will be different than more traditional classes. Examples include frequent real-world code reviews, software engineer interview practice, and reputation as a part of the final grade. The intention is to give a more accurate picture of what it will mean to be a software engineer as a full-time job.  
  
Class Format

Class time will be varying combinations of lecture, live coding, and class collaboration. Class participation is expected, which includes questions and answers during live coding as well as discussion in groups when appropriate. All live coding done during class as well as any code necessary for assignments will be available at https://github.com/spazard1/Web-Programming-FA2022.

Reputation

Many courses have a portion of the grade allocated to participation, but participation doesn’t model effectively enough what a software engineer career will entail. Elements such as timeliness of code, interaction in meetings and groups, going above and beyond what is expected, and showing passion will all raise reputation in a company. Those concepts will be rewarded in this class; examples of what it that means practically are participation in discussions, doing more than assignments require through stretch levels, and quality and timeliness of your code.

Reputation values throughout the semester do not directly correspond to a grade percentage. When the class is complete, reputation values will be converted into a grade from an average of all the reputation values in the class. See the Grade Breakdown for more details.  
  
Assignments

Assignments will be introduced on the Tuesday of each week and will be due a week after the following Thursday. This means each assignment must be completed in a maximum of nine days. All assignments will primarily be creating new code projects or adding additional code to existing projects. Late assignments are accepted for full credit up to two weeks after their due date, but being repeatedly late will negatively affect your reputation.

Each assignment will have stretch levels for those students who wish to push themselves beyond the baseline level of the class. Often these stretch levels will include topics not covered in class and will require outside investigation and learning to complete. Students who succeed at these stretch levels will be given extra reputation.

Final Exam

The final exam will involve each student being given a fully functioning web service program. Each student will independently code review this program and will be graded on their ability to make good comments on necessary changes to the code based on the topics covered in this course.  
  
Institutional Policies

Violation of honesty standards can result in denial of credit (U or F) in a course, as well as dismissal from the university. Penalties are given at the discretion of the faculty member, and offenders will be referred to the associate provost of the College of Arts and Sciences. (See Bethel University’s full policy on Academic Honesty in the catalog: (<http://catalog.bethel.edu/arts-sciences/general-information/academic-honesty/>).

Accessibility

Bethel University is committed to accessibility for students with disabilities and the Office of Accessibility Resources & Services (OARS) is a resource to ensure students experience access. Reasonable accommodations are approved after an interactive process with the student and OARS. The instructor will provide accommodations, but the student is required to initiate the process.

Students with a documented disability may contact OARS to learn more about how to register for accommodations. Reasonable accommodations are approved after an interactive process with the student and OARS.

Students registered with OARS are responsible for logging in to their AIM Accessibility Accommodation portal (via MyBethel) each term to request their Faculty Notification Letter of Accommodations. Accommodations cannot be applied prior to the faculty’s receipt of the letter.

OARS recommends the student and faculty discuss how accommodations may apply in the specific course. Accommodations cannot modify essential requirements or fundamentally alter the nature of the course. Consultation with OARS may be necessary to clarify reasonable accommodations based on the course. If there are any questions or concerns, connect with OARS at accessibility-serivces@bethel.edu or 651.638.6833.

Grade Breakdown

The average reputation of all students will be converted into a B- (80%), with higher and lower reputations receiving proportion grades.

Reputation 25%  
Assignments 60%  
Final Exam 15%

|  |  |  |
| --- | --- | --- |
| ≥93 – 100 % A ≥90 – <93 % A- ≥87 – <90 % B+ ≥83 – <87 % B | ≥80 – <83 % B-  ≥76 – <80 % C+  ≥65 – <76 % C | ≥60 – <65 % C-  ≥57 – <60 % D+  ≥50 – <57 % D |

Topics Overview

|  |  |
| --- | --- |
| HTML, CSS Javascript, JSON, XML HTTP (Headers, Verbs, Parameters, Status Codes) REST (Controllers/Routes)  MVC/WebApi/Asp.Net  Filters/Middleware, CORS Dependency Injection Transient Fault Handling | Async/Sync (async/await, Promises)  Cloud Deployment, Logging, Monitoring  Workers, Queues Optimistic Concurrency Control  API Versioning JWT Service Configuration  Logging |