Practice Problem App

Software Requirements Specification

CSCE 247: Software Engineering

Steven Cole, Simeon Hearrell, Austin Bramlett

January 23rd, 2025

Version 1.0

Table of Contents

1. Introduction

Purpose

The Computer Science and Engineering program at the University of South Carolina places a strong emphasis on programming expertise, algorithmic thinking, and problem-solving. Throughout their academic careers, students are expected to maintain and improve upon the fundamental principles introduced in courses like CSCE 145/146, CSCE 215, and other core topics. However, finding organized and structured practice problems that closely relate to the content covered in these courses is a challenge for many students. The goal of this project is to provide a platform for coding practice and conversation that is designed for University of South Carolina Computer Science and Engineering students. In addition to helping students repeat topics learned in class, this application will include course-aligned coding assignments, explanations, and discussion features that will help them prepare for technical interviews in an ethical and organized way.

Scope

This document will cover:

- The personas of potential users and stakeholders invested in this project.
- Any constraints that have been applied to this project.
- A description of the website/app along with its business use cases.
- Shows both the functional and non-functional requirements of the app.
- A competitive analysis to outline the purpose of the projects.

2. Stakeholders

- University of South Carolina Students
    - Computer Science Students
    - Computer Engineering Students
    - Computer Information Systems Students
    - Upper-level students
    - Teaching Assistants
- University of South Carolina
- Faculty and Academic Staff
    - Professors
    - Staff
- Developers

Personas

JACOB JONES

PROFILE

Age	:	19
Hometown	:	Charleston, SC
Student	:	Undergraduate
Work	:	Part-time Dining Employee
Character	:	Learner

BIOGRAPHY

Alex is enrolled in CSCE 146 as a sophomore at the University of South Carolina studying computer science. After finishing CSCE 145, Alex is uncertain about his ability to remember fundamental data structures, loops, arrays, and recursion. Alex finds the current code practice platforms intimidating and ill-suited to UofSC education, despite his desire to excel academically and land an internship.

GOALS

- Practice Coding problems
- Strengthen understanding of programming concepts
- Prepare for technical interviews
- Track progress and improvement of coding over time

FUSTRATIONS

- Problems with finding problems that follow what is taught in class
- Online answers do not explain solutions
- Does not know which topics are most important to review

PERSONALITY

Introvert — Extrovert

Thinking — Feeling

Judging — Perceiving

Sensing — Intuition

TECHNOLOGY

Software

Social Media

Mobile App

CASSIE RIVERS

PROFILE

Age	:	22
Hometown	:	Raleigh, NC
Student	:	Undergraduate
Work	:	Undergraduate Teaching Assistant
Character	:	Mentor

BIOGRAPHY

After completing most of the CSCE curriculum, Jordan, a senior in computer engineering, works as a teaching assistant for CSCE 145. Jordan values helping underclassmen understand difficult programming concepts and believes that teaching others enhances one's own development. Jordan wants a structured system that enables students to study effectively and efficiently without compromising their academic integrity.

GOALS

- Share accurate solutions and explanations
- Help students understand concepts of coding
- Encourage students to use ethical learning practices
- Contribute to student success

FUSTRATIONS

- Concerns about students cheating using unethical practices
- Repeating explanations
- Lack of practice resources

PERSONALITY

Introvert — Extrovert

Thinking — Feeling

Judging — Perceiving

Sensing — Intuition

TECHNOLOGY

Software

Social Media

Mobile App

DR. TROY BENNET

PROFILE

Age	:	40
Hometown	:	Columbia, SC
Student	:	Professor
Work	:	Computer Science & Engineering
Character	:	Educator

BIOGRAPHY

Dr. Reynolds teaches at the University of South Carolina's Department of Computer Science and Engineering. They are committed to the academic and extracurricular success of their students and provide a variety of undergraduate CSCE courses. Although Dr. Reynolds acknowledges that students use outside resources, he favors platforms that uphold academic integrity while reinforcing learning.

GOALS

- Improve student understanding of core CS concepts
- Encourage practice outside of class
- Reduce use of unapproved online resources
- Maintain academic honesty/integrity

FUSTRATIONS

- Students using unapproved online resources
- Limited ability to see where students struggle conceptually
- Practice tools that violate course guidelines

PERSONALITY

Introvert — Extrovert

Thinking — Feeling

Judging — Perceiving

Sensing — Intuition

TECHNOLOGY

Software

Social Media

Mobile App

3. Constraints

Time Constraints

- This project needs to be completed within a span of a few weeks.

Monetary Constraints

- The project's development has a budget of $0.

Technical Constraints

- The code will be programmed in Java.
- The system must be able to run on multiple different devices.

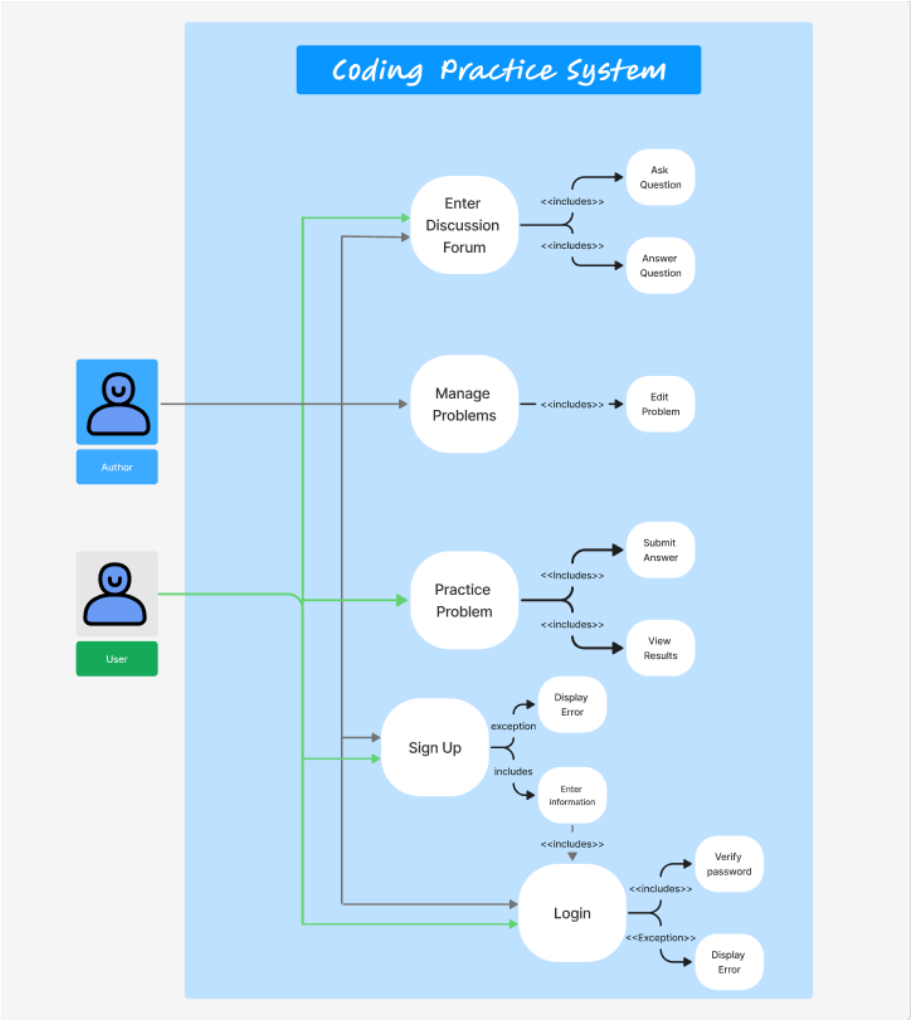
4. Overall Description

This system will be available to University of South Carolina Computer Science and Engineering students. Users will be able to access their account through a personal login.

The system will provide coding practice problems related to courses such as CSCE 145, 146 and others. Students will be able to view problems and then attempt them with help if needed. General explanations will be provided.

The system will store feedback and other inquiries in order to improve over time. The system will be simple to use in order to promote frequency and easy access to learning.

5. Business Use Cases



6. Functional Requirements

Functional Requirements

7. Non-Functional Requirements

Look and Feel Requirements

- The system shall provide a simple interface that all students can easily understand when practicing coding problems.

Usability Requirements

- The system shall be easy to navigate for users with different levels of experience.

Performance Requirements

- The system shall load problems and features quickly on any device connected to the internet.

Maintainability and Support Requirements

- The system shall be able to run on Windows, Linux, and Mac machines and allow it to update problems easily.

Security Requirements

- The system shall protect user accounts and data from misuse.
- The system shall provide a safe way for users to recover their own lost account data.

Disciplinary Requirements

- The system shall provide certain trusted users with administrative roles to enforce the rules of the site.

Legal Requirements

- The system shall provide users the option to delete their account and remove their data from the website entirely.
- The rules will ensure that the social aspect of the system will be a suitably safe environment.
- All data collected without the user's express consent (if any) will be covered by the user consenting to a privacy policy that details the data as collected.

9. Competitive Analysis

LeetCode	
Strengths	Leetcode problems are recognized as reliable proof of proficiency by many employers. The site has existed for a long time, so there are a lot

	of problems available. The problems that do exist are divided into difficulties for user and employer ease of understanding. The site hosts contests where users may compete with others to prove their skills. The site hosts a forum where users may interact with each other. They host mock assessments so users can take them in anticipation for what they may encounter during a real interview. They have crash courses where users may learn important coding skills by themselves. They also have a built-in compiler.
Weaknesses	They have a subscription service, so not all services are free. The website only focuses on code/programming.
Audience/Focus	Anybody who is looking to build experience with coding for a resume.
StackOverflow	
Strengths	The site is in the form of a forum, allowing for a system where users can answer questions about almost any type of code and be rated based on their accuracy. The site also comes with a job searcher powered by Indeed. StackOverflow has existed for so long there's a good chance that if there's a programming issue in your code, you can find an answer to that problem through StackOverflow. They have this feature called 'Challenges' where people can post problems to be solved for practice.
Weaknesses	No practice problems, only questions and answers. The site is entirely focused on programming, not engineering or other subjects. 'Challenges' cannot be put on resumes or job listings, according to the website itself. Alleged unaddressed toxicity within the community leads to people who are new at coding getting unnecessarily negative feedback for asking easy-to-solve questions. User interaction with the site is declining.
Audience/Focus	Any student who needs to code. Anybody who needs to code as a job. People who code as a hobby or are learning to code.

GeeksForGeeks	
Strengths	Lots of documentation for code. Some programming courses available, as well as tutorials and interview prep material. They have a built-in compiler, and are very similar to LeetCode (with the exception of practice problems and resume building.) The classes they offer are industry-recognized. They have practice problems, but they don't seem to have as much recognition as LeetCode.
Weaknesses	Similarly to LeetCode, services must be paid for, but at least its greatest strength (documentation for beginners) is free. No forums, unlike the other two. Layout of the website is not very intuitive outside of the documentation they have for coding languages. Things are pretty hard to find without deliberately searching for them with the search bar. The website is focused entirely on programming.
Audience/Focus	People who are trying to learn how to code. People who need a refresh on documentation. People who are looking to prepare for coding interviews.

Summary	**Strengths**	**Weaknesses**	**Audience/Focus**
Leetcode	+ Practice problems valued by employers + Coding courses + Built-in compiler + Forum with upvoting system + Aged and reputable + Interview prep materials	- Certain features require a subscription - Entirely focused on coding	Anybody who is looking to build experience with coding for a resume, whether they be students or graduates.
StackOverflow	+ Forum environment + Practice problems + Job searching built into the website + Very old and still supported	- Potentially unfriendly community - Usage declining - Practice	Anyone who has problems with their code and wants to solve them.

	+ Virtually unlimited supported coding languages	problems hold no value on a resume - Focused on coding	
GeeksForGeeks	+ Industry recognized coding classes on website + Contains a lot of helpful documentation for people looking to get started + Coding practice problems	- Unintuitive layout - Classes must be paid for - Focused on coding - No forums	People who are new to coding, and people who are looking to prepare for interviews.

The competitive analysis above offered us a lot of insight into what similar websites have historically achieved. This should greatly help us in our abilities to improve our product. To make our product stand out amongst its competitors, we have concluded that our product must include a forum where students may post questions and answers, along with an entire section of our product devoted towards industry-standard coding problems. We need to have an intuitive layout, as well as a functional disciplinary system to ensure the community remains friendly in the event bad actors appear. We need to include links to code documentation for people who are struggling to understand their code. We also need to expand our reach beyond code, dividing the framework of our website into sections devoted to helping each discipline we choose to support. With just these features at minimum, our product will stand out among others. While other strengths listed in the summary chart above are possible to implement, they are highly ambitious and not all of them are realistic to add with our numbers and current timeframe.