Project Plan Documentation

Cringe Coders

Austin Bailey
Mya Bell
Nolan Gregory
Lindsey Langdon

Comprehensive Project Overview
for the CSLC Portal

Dr. Harvey Siy; College IS&T
University of Nebraska - Omaha
Capstone Course
August 31st, 2023

Contents

CHAPTER 1

Client Information

1.1 Organization

The Computer Science Learning Center (CSLC), is a student service hosted through the College of Information Science & Technology (IS&T) for academic assistance and tutoring. The CSLC accepts tutoring by appointment or walk-ins Monday through Saturdays, and distance tutoring over Zoom. Student-tutors are paid for their services, and the University has administrators to oversee and manage CSLC staffing levels, work shifts, and payroll.

1.2 Point of Contact

Kyle Reestman is the director of the Computer Science Learning Center. Apart from his directorial position, he also serves as an instructor for CIST1400, an introductory computer science course. This course is designed to provide students with a comprehensive overview of programming basics utilizing the Python programming language. Furthermore, as the director of the CSLC, Kyle oversees the orchestration and management of the tutoring center's operations. He coordinates the tutors schedules, supervises the tutors, and ensures that the center's tutors are well-equipped with the necessary teaching materials. Through his capacities as both an educator and a director, Kyle Reestman guarantees the smooth operation of the CSLC while fostering students' development in computer science.

For the duration of this capstone, Kyle Reestman will be the primary point of contact. Secondaries, subject matter expert, and other consultation contacts will be at the delegation of Kyle Reestman, to be determined as necessary throughout the course of the project.

CHAPTER 2

Motivation

2.1 Context

The Computer Science Tutoring Portal is a comprehensive web application designed to streamline and enhance the tutoring experience within the Computer Science Learning Center. This portal serves as a centralized platform for students, tutors, and administrators to manage tutoring requests, track progress, and facilitate efficient communication. The application's key features include ticket management, tutor and course administration, and seamless interaction between students, tutors, and administrators.

2.2 System Needs

The existing CSLC portal has incomplete or missing features, functionality, and services. The current ticket submission system functions through a Google Form tied to a spreadsheet with the resulting student information.

2.3 Proposed Solution

To better meet the needs of the CSLC, the Cringe Coders proposes a full stack refresh of the existing technology stack. The ticketing system at present will be rebuilt to provide tutors and students with a more efficient way for exchanging essential information. This encompasses sharing specifics about the class assistance needed, articulating the present comprehension of the problem, identifying the respective course instructor, and the level of priority. The new portal will be able to facilitate this seamless sharing of this information between tutors and students.

2.4 Client Role

The client holds the important role of qualifying goals, and acceptance criteria. Throughout this project the client will be the source of truth and final authority for product quality and completeness. This power requires that the client provides periodic input into goals and the scope of work. To prevent redundant or extraneous work, a scheduled periodic meeting and an ad hoc communication channel are required during the continuous delivery process.

CHAPTER 3

Project Description

3.1 Project Overview

The initial conception of the scope of work involves a complete overhaul of front-end and back-end frameworks. Feature augmentations will be added as part of the refresh.

3.2 Features and Services

Concurrent with the replacement the current portal with a full refresh of the CSLC portal, Cringe Coders will add database tie-ins to the portal to replace the current Google Form solution.

3.3 Languages

For this project, we anticipate using the following languages: Python, JavaScript, HTML, and CSS. We will also use the following frameworks and libraries: Django, React, and Tailwind.

3.4 Technology Stack

The front-end will use the React library and Tailwind CSS to enable better UI/UX. While our back-end will be written using the Django ORM. We will treat it as a REST back-end by using the Django REST framework. This will allow us to populate the webpage with real time database information via API calls. We will use Postman to test our API, and we will use sqllite as our database. Additionally, we will deploy our application using docker for containerization, Kubernetes for automating deployment/management of containerized applications, and we will host our application on Azure, pending any client objections.

3.5 Documentation

Lastly, we will use a combination of Sphinx and Autodoc for our technical documentation. We will also use LaTeX to write our documentation (such as this document you are reading).

3.6 Potential Features and Services

At present no further work is scoped. This will be revisited and addressed according to client needs.

CHAPTER 4

Implementation Strategy

4.1 Website Refresh

This transformation starts with a redesign of the website's user interface. This shift requires moving away from the CSLC's current reliance on Google Forms. It involves constructing a new front-end using react, a widely recognized Javascript framework, and Tailwind CSS. This front-end will be intricately linked to a back-end using the Django REST framework and a database server.

4.2 Back-End Database & New Features

Later continuous delivery cycles will focus on creating a sqllite database and integrating it with the refreshed portal website.

4.3 Security Controls

At all stages of delivery, security controls will be implemented at an architectural level, and periodically audited for validity and completeness.

CHAPTER 5

Computer Science Challenges

5.1 Challenges

We anticipate only trivial computer science challenges will arise in the scope of work. The only challenge revolves around the synchronization of the front-end and back-end for real-time updates. Successfully establishing seamless communication between these elements, alongside the portal's relational sqllite database, is mentioned because the implementation is novel to the Cringe Coders team.

CHAPTER 6

Computer Engineering Challenges

6.1 New Frameworks & Technologies

The success of this project hinges on everyone's familiarity with the technology stack in use. For some team members, and potentially the entire team, this means dedicating time to become acquainted with new frameworks and programming languages. The capacity to acquire new skills is essential in the world of software engineering. Also, given that not all team members have worked together before, getting an understanding on each other's work processes presents an additional challenge. In essence, the success of this project not only hinges on the team's technical expertise but also equally relies on the team's collaboration and communication.

CHAPTER 7

Team Members

7.1 Background on the Cringe Coders

The Cringe Coders consists of four members with a range of hard skills; soft skills; professional computer science experience; and, experiences in adjacent fields of information assurance and mathematics. Based on internal deliberation among team members, we have aligned ourselves to the roles most apt for individual skill sets.

7.2 Austin Bailey

Client Liaison
Information Security Analyst
Quality Assurance Tester

Austin Bailey is responsible for thoroughly testing the application to identify vulnerabilities, usability issues, and performance bottlenecks. Austin will create and execute test cases, provide feedback to the development team, and ensure a high-quality final product. Likewise, Austin will simulate real-world cyberattacks to identify vulnerabilities and weaknesses. Austin will help our team improve our security posture and protect sensitive information by uncovering any potential security flaws.

7.3 Mya Bell

Requirements Analyst
Quality Assurance Tester
Client Liaison

Mya Bell will oversee the development process, and ensure that the project stays on track, objectives are met, and communication flows smoothly between team members. Likewise, Mya will facilitate communication and collaboration between leadership and team players to ensure a successful outcome. Lastly, Mya will create and execute test cases to ensure a high-quality final product.

7.4 Nolan Gregory

Architect
Technical Lead
Back-End Developer
Dev-Ops and Database

Nolan Gregory is responsible for making technical decisions and guiding the development process. Nolan will provide technical expertise, review code, and ensure that the architecture and technologies chosen align with the project's goals. Nolan will handle all server-side logic, database management, and API calls. Nolan will build the core functionalities of the application, including ticket management, user authentication, and communication features. Nolan will design and maintain the database structure, ensuring efficient data storage, retrieval, and integrity. Lastly, Nolan will set up the deployment infrastructure, manage continuous integration/continuous deployment (CI/CD) pipelines, and ensures smooth deployment and scaling of the ticketing portal.

7.5 Lindsey Langdon

UI/UX Lead
Front-End Developer
Mobile Design Lead

Lindsey Langdon is responsible for creating an intuitive and visually appealing user interface. Lindsey will design wireframes, mockups, and prototypes, ensuring a user-friendly experience and consistent branding. Lindsey will be responsible for implementing the user interface and ensuring that the application is responsive and visually engaging. They work closely with the UI/UX Designer to translate design concepts into functional front-end components. Lindsey will ensure that the front-end can run on mobile or tablet devices, as well as on desktop systems, with a focus on mobile-first design.