Technical Scope Document

for

Apprentice

Version 1.0

Group DNWS

March 5th, 2022

Table of Contents

1. Business Requirements & Technical Scope

1.1. Background

Schools offer a plethora of services that are readily available to their students and more often than not, students are also able to get tons of discounts and rewards for just being a student. Many students do not know what they are paying for and what perks they have access to until much time has passed. Instead of having to remember a list of services that are offered to you by your school, with Apprentice you have access to all that information and how to utilize it as well as a forum where students can converse with other students if they have any questions regarding any of the services.

1.2. Technical Objectives

Our technical objectives for apprentice are:

- Develop a website that is easily accessible through cross platform devices.
- Store user-generated content from deals and features as a static placement.
- Develop a database of accounts that are user generated.
- Have user generated posts organized into tags that are a part of specific genres.
- Have basic user customization capabilities when it comes to their profiles.

Cross Platform

We want to build a platform that is accessible through all platforms (Desktop and Mobile) in an organized, modern UI/UX design. We want to accommodate a majority of students accessing our platform. We believe that developing our platform to be web app based, will be the best scenario.

Streamline Storing of User Data

We want to develop a platform that establishes efficient storing techniques for user data when they post listings and comment to listings on the platform. We believe that the best method for this is using Web Storage. WebStorage API allows the storing of more well defined information locally rather than using cookies.

Account Database Integration

We want our platform to store account information with reliability, speed, and security. We believe that MySQL will be the best platform to use for storing account information in a secure format. MySQL is an open-source database management system (RDBMS) that supplies ample documentation for DNWS to use.

User generated posts tagged and organized

In order for our platform to be convenient for users, posts need to be organized and tagged. We will be using JS to organize and tag posts in specific categories. JS can be used to allow the capabilities of tagging data and organize it into a neat modern design.

Basic User Customizations

In a modern platform that involves the use of social media techniques, it's assumed that the user will have some form of customization. This is important because the user needs to feel that they can make something on the platform their own. We will provide basic customizations that will include:

- user-generated username
- user-generated profile picture
- user-generated bio on profile page.
- user-generated feeds

Using JS intertwined with MySQL.

1.3. Customer Requirements

- Users require a platform that is free for users to access. There will be a subscription tier list later on for users to have additional features to better help acquire that information.
- Users require that the platform has a modern UI/UX design that is up to today's standards.
- Users require that the platform has reasonable functionality and reliability.

2. Scope and Limitations

2.1. Scope

Our goal is to provide a platform that is reliable, functional, and modern UX/UI design that aggregates deals and hidden features for college students associated with their student account. The scope of the Apprentice project encompasses tasks and/or functions typically associated with the development and management of a modern web application including but not limited to the areas of:

- Planning and blueprints
- Web Application Design
- Copywriting and Labeling
- Testing and Launch
- App maintenance

Planning and blueprint

In this stage we are planning out who our customers are and what is the problem we are trying to solve. In this process we are also building out a general idea of what is needed for the project in our solution to the overall problem. We will have an overview of the goal of the web application. Apprentice is a web application that is designed to help college students by aggregating deals and hidden features associated with student accounts. This web application will be hosted on a web server where users can access through desktop or mobile devices.

Web Application Design

Apprentice will be developed as a JavaScript Web Application. This Web Application will be designed to have fluid and modern interactions. The color palette for this application will be cheerful with the base color being blue. The web application will be storing data and information using WebStorage API. WebStorage API is flexible and can be used with MySQL to provide a fast flow of data retention and data transfers for users and user-generated content.

Web Application Programming

Apprentice application will be set up on a linux webserver. Linux allows ease for web developers compared to other Operating systems. The linux server will have an easy install and set up the server for the apprentice application. Majority of the web programming will be done in these three areas:

- setting up the packages for installation of the server.
- Configuring MySQL server to work with the web application
- Developing the Web Application in JavaScript.

Copywriting and Labeling

Documentation on how an apprentice application needs to be set up and used is of the utmost importance. We will have documentation detailing what needs to be done for installing the server on a linux system. There will also be documentation for getting started with customer experience.

Testing and Launch

After the application has a minimum amount of code in order to be functional, we will begin testing the app for bugs or issues that can go wrong with the app.

Application Maintenance

After the apprentice has been released, we continue to test any unintended problems that may occur. The process of building this app runs itself all over again because we are following the spiral method.

2.2. Limitations and Exclusions

For this project, we have various Limitations and Exclusions due to us implementing a spiral development process. With this for our initial release we are excluding:

- AI Based Moderation
- Subscription Based Features
- Chat forum for news feed

With our initial release, our limitations are:

- Providing basic features for themes and profile pictures.
- Establishing a distributed platform that maintains data across multiple servers.
- Limited verification processes of student accounts.
- Limit the posts to be text based with hyperlinks.
- Limit user-generated post to 500 words
- Limit user's feed to 5 listings per refresh.
- Moderation will be handled by administrator accounts on the platform.
- Users cannot delete posts, but can edit or add.