**Digital Collections SPMP**

**Neal Bhalodia**

1.1

Project Overview: The app has the following requirements: it must provide a simple way for users to add items from their physical collections to Digital Collections, must work on mobile and desktop browsers, and must be suitable for people from all age groups and also for people who aren’t good with technology. In order to make it easy to use for everyone the user interface will be simple and easy to navigate. It should allow users to register and login from any browser and store the user’s data on a database.

The all portions of the development of this project will be done by Neal Bhalodia who will communicate with the client (the professor) about the project along the way.

1.2

Project Deliverables:

* Requirements
* Use Cases
* Functional release.
  + The application at this point should be usable with some minor bugs. The bugs will be fixed as a part of the regular maintenance of the application.

1.3

Evolution: All issues that take place will be discussed between the client and myself. I will be communicating with the client along the development process to get inputs and correct issues.

The software project management plan is under version control. The repository will be updated as the project progresses. Each commit will have information on what was changed during the update.

https://github.com/ibneal/DigitalCollections

1.4

Reference Materials: I currently use a system called Shop Keep to manage inventory for my store but it is meant for a retail brick and mortar business. It has many of the functions I want like being able to search items, being able to track quantity, or other attributes like which supplier you bought it from. I am using

Graphical user interface

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

I will be getting a lot of the inspiration for the layout from Shop Keep. I really like their navigation menu, but I don’t like that there are no images for each of the items. I want there to be multiple layouts the user can pick but definite give the option to view images of the items as well.

I also like that the add item function that the system uses. It takes the user to new page where they can input all the relevant information for their item.

1.5

Definitions and Acronyms:

**Collection**: any set of similar items.

**Item**: A specific product from a collection.

**Views**: Different ways a user can view the items in a collection when they are browsing.

2.1

Process Model: The application will allow users to create an account where they store digital versions of their physical collection. Once a user signs up they will be able to create a collection with unique attributes. Once the collection is created the user can begin to add items to the collection. Users will be able to add items manually and eventually be able to add items using a CSV that will allow for bulk imports. Once the collections are complete users can show people specific items by searching.

The waterfall model will be used as the process model for this project.



2.2

Organizational Model: Since this project is expected to be done alone, I will be the only person working on the software and all aspects of the project. I have created a table with roles that need to be done throughout the project.

|  |  |  |
| --- | --- | --- |
| Roles | Due Date | Assignee |
| Project Specification (Requirements, SPMP, Use Cases) | Nov 18 | Neal |
| Build Database | Nov 20 | Neal |
| Create Code functions | Nov 25 | Neal |
| Build UI | Nov 18 | Neal |
| Connect Database to UI | Nov 18 | Neal |
| Test Site | Nov 28 | Users |
| Create Reference Material | Nov 30 | Neal |
| Deploy site | Nov 31 | Neal |

2.3

Organizational Interfaces:

**Heroku** is a domain and hosting service many multiple features that allow you to deploy updates to the project much more easily. I will also be able to

**Github** will be used to manage the work done on this application and for version control. It will also be connected to Heroku where the project will be deployed.

**JawDB** will be the database service used during this project. It will connect to Heroku where the whole project will be deployed. I will also be able to manage the database remotely.

**MySQL WorkBench** will be used to remotely manage the database. This application will primarily be used to create tables and set up the initial structure.

2.4

Project Responsibilities:

General high-level responsibilities:

The purpose of this project is to create an application that will allow users to manage their physical item collections. It will be a web-based application that will work on browsers so nothing will need to be installed for users. The application needs to be secure and be able to store user’s information so that they can login from any browser.

Specific responsibilities:

Create a sample UX without functionality to show the client.

Create methods for each piece of function.

Use MongoDB as the database to store user data.

Create a functional version of the application for test users.

Get feedback from test users on major flaws and changes required.

Create a plan to address the issues laid out by test users and fix them.

Continue to fix bugs and updates over the life of Digital Collections

3.1

Management Objective and Priorities: Responsibilities regarding developing the application will be on me. I will also be communicating with the client along the way. The priorities in development are achieving a functional prototype of the application in which all use cases run smoothly by early December and a technical report in November.

Important features client requires:

* Must have the ability to have images for each item in the collection.
* User must be able to use application on a browser without any installations.
* Login feature
* Information must be stored securely for users.

3.2

Assumptions, Dependencies and Constraints: The cooperation and communication between myself and the client (professor) is very important to the completion of this project.

Assumptions – There are parts of the project that I cannot proceed with unless it is approved by the client (professor).

Constrains – The application is being developed solely by one person under a time constraint. The application is also restricted to browsers. There is definitely some constraint in the development of this application since only one person is working on it.

Dependencies – The application is running Node JS on the backend and React for the front end. These dependencies will automatically be installed on Heroku with a package.json file.

3.3

Risk Management: We will check that all data is properly secured when testing the functionality of the application.

There is always a threat of user data being stolen. To protect user security the hosting of the website will be on a secure server and user data will also be secured.

The application will be dependent on funding from users in the form of donations. There is a risk that users will not donate which will lead to the application not being maintained.

The code will not be scalable to allow for new features. The only real contingency for this is to inform the client of the functionality available and to decide whether there is time enough to alter the code so that it is scalable with a fair amount of ease.

3.4

Monitoring and Controlling Mechanism:

Github will be used as the primary version control mechanism for the project.

3.5

Staffing Plan: One developer will handle all aspects of the development process. The client will provide input along the way and ask for any specific changes that they would like to see.

4.1

Methods, Tools, and Techniques

Javascript (React/ Node JS)

HTML/CSS

MongoDB/JawsDB

WebStorm IDE

JavaScript, HTML, and CSS will be the languages used to program the application. I will use WebStorm as my IDE. MongoDB is a nonrelational cloud database that will store all the user’s data.

4.2

Software Documentation

All lines of code will have comments that explain what the line’s purpose is, so that all future developers can clearly understand the methods and previously written code. There will also be documents with information about what each method does.

4.3

Project Support Functions

To ensure quality assurance all aspects of the application will be tested thoroughly with specific consideration for the user experience.

Configuration management

Verification

5

Work Breakdown Structure

Diagram

Description automatically generated

Timeline

Description automatically generated