

Design Document

ExploreXpert



Luis Bonilla
Grace Metallo
Jennifer LeViness

luigibonilla2003@gmail.com
Gracemetallogg@gmail.com
JJLeViness@student.fullsail.edu

Contents

1. **Introduction**
 - 1.1. [Purpose](#)
 - 1.2. [Goal](#)
 - 1.3. [Definitions](#)
2. **Overall Description**
 - 2.1. [Product Perspective](#)
 - 2.1.1. [User Interface Hardware](#)
 - 2.1.2. [Hardware interfaces & Memory Constraints](#)
 - 2.1.3. [Software interfaces](#)
 - 2.1.4. [Library Dependencies](#)
 - 2.1.5. [3rd Party Data and APIs](#)
 - 2.2.
3. **Specific Requirements**
 - 3.1. [MVP](#)
 - 3.2. [Alpha](#)
 - 3.3. [Beta](#)
4. **Use Cases**
 - 4.1. [User characteristics](#)
 - 4.2. [User interface](#)
5. **Development Environment**
 - 5.1. [Developer Tools](#)
 - 5.2. [Content-Generation Tools](#)
 - 5.3. [Data Tools](#)
 - 5.4. [3rd-Party Dependencies and APIs](#)
6. **System Architecture**
 - 6.1. [ERD Class Diagram](#)
7. **Production Plans**
 - 7.1. [Code Review Plan](#)
 - 7.2. [Integration Plan](#)
 - 7.3. [Testing Plans](#)
8. **Folder Hierarchy**

1. Introduction

Purpose:

ExploreXpert aims to provide users with an interactive way to discover and explore new restaurants, historical landmarks and unique attractions in the city they are visiting or live in. Using features such as scavenger hunts, augmented reality and contributions from fellow users, ExploreXpert will help visitors see the best of where they are and locals to discover something new.

Goal: What problem is this solving?

The goal of ExploreXpert in the simplest of terms is to help users discover places around cities that they may not even know exist. Often the only places we see when traveling or just even locally to our own cities are only seen on Social Media and go “viral”. ExploreXpert will help and even reward users for visiting and learning about places around their cities and they may discover something along the way.

Existing Solution:

While this project is not based on a previous project it is inspired by frustration with the problem this app is being designed to solve. There are similar applications that do exist “Geocaching” for one can be described as a global hide and seek; and while it is an excellent way to explore those are usually a ‘One and done’ hunt.

Atlas Obscura an application which helps its users find as the name suggests obscure places. While informational it is simply just that, a source of information. ExploreXpert will combine that hide and seek aspect with the Scavenger Hunts and help users have a more interactive experience to discover something new that can stick with them beyond a singular day.

Geocaching

Curious and Wondrous Travel Destinations - Atlas Obscura

Definitions:

Scavenger Hunt - A game where players must find a specific item or location. Often follows a route or a theme when completing tasks.

Points of Interest- Specific locations including landmarks, historical sites, restaurants and other attractions that may be of interest to users.

Augmented Reality - Technology that overlays information onto a real world view seen through the device's camera

Google Places API:- Service that provides information about various locations including names and addresses.

Google Maps API- Service that provides interactive maps and location based navigation

Firebase - Platform developed by Google for creating applications. Service that provides real-time databases, user authentication, and other cloud functions.

2. Overall Description

2.1. Product Perspective

2.1.1. User interface Hardware
Touch Screen

2.1.2. Hardware interfaces & Memory Constraints
Android Smartphone or Tablet
Device Camera
GPS
Internet Connectivity

<Memory Constraints to be added>

2.1.3. Software interfaces
Android OS 7.0 and above

2.1.4. Library Dependencies
Android SDK with Java - Framework
Firebase SDK
Google Cloud - Google Maps, Places
Android Jetpack Components

2.1.5. Database
Local-SQLite, Cloud- Firebase

2.1.6. 3rd Party Data and APIs
Google Maps API, Google Places API, Firebase, ARCore API

3. Specific Requirements

3.1. Milestones

3.1.1. Minimum Viable Product (A Features)

The user will be able to log in to the app using an email and password.
The user will be able to register using an email and selected password
The user will be able to retrieve a forgotten password
The user will be able to log out of their account
The user will be able to view a map showing their current location.
The user will be able to navigate to certain points using the map.
The User will be able to select a scavenger hunt from a list.
The user will be able to view tasks and checkpoints for scavenger hunts.

The user will be able to earn and view points for completing tasks in scavenger hunts.

The user will be able to view information about a location, name, address, and description.

The user will have their scavenger hunt progress stored locally on their device.

3.1.2. Alpha (B Features)

The user will be able to rate and read reviews about a place.

The user will be able to create and edit a user profile.

The user will be able to view and share past achievements in their profile.

The user will see updated content dynamically.

(tentative) The user will be able to use their device camera to view and interact with AR overlays at specific locations.

3.1.3. Beta (C Feature)

The user will receive recommendations for new places based on preferences or history.

The user will be able to suggest new locations and scavenger hunts for others.

The user will be able to see promotions and discounts for local businesses.

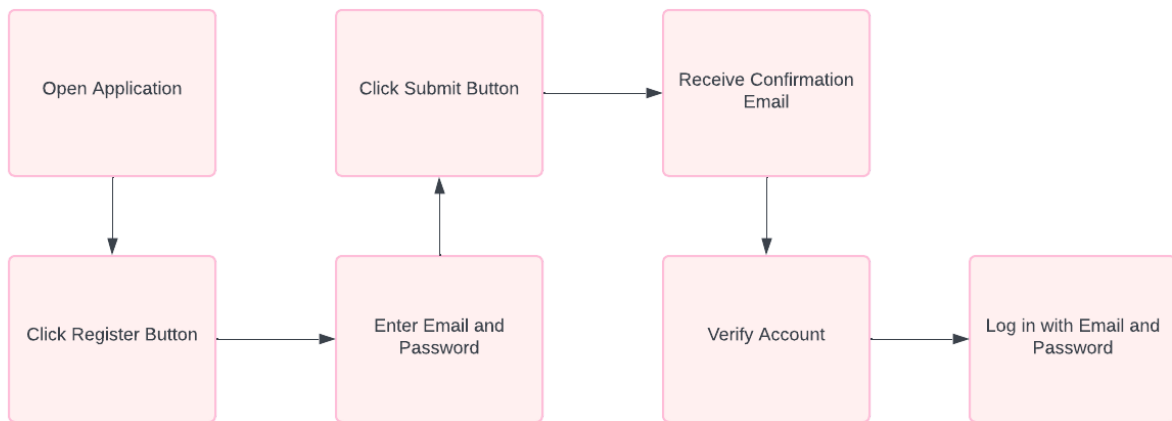
The user will be able to earn rewards from partnered local businesses.

4. User Experience

4.1. Use Cases

Registration: User will be able to create an account in order to track their progress, earn points and rewards.

1. User opens application
2. User clicks on Register button
3. User enters email and desired password
4. User clicks on Submit Button
5. User receives a confirmation email and verifies their account
6. User is able to log in with a registered email and password.



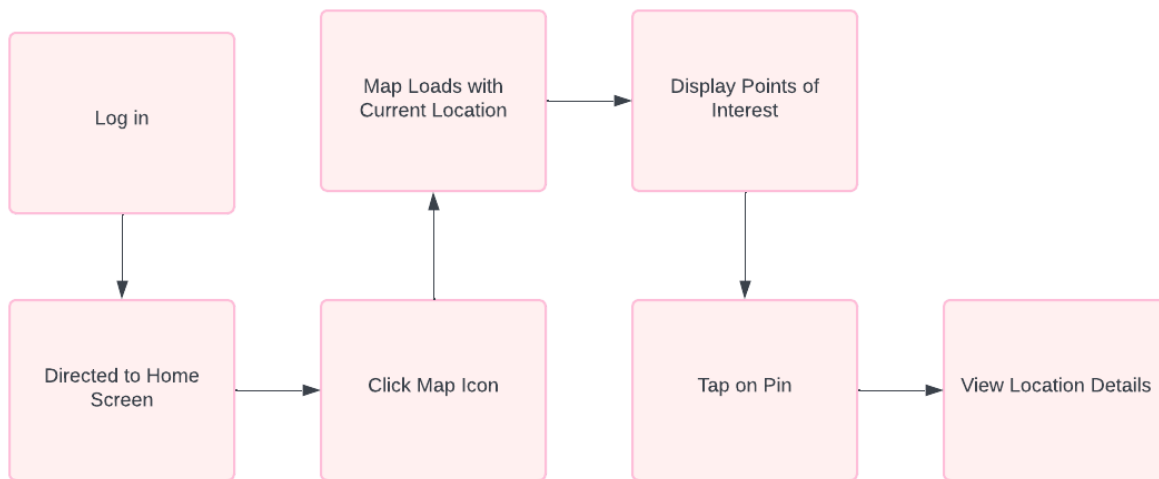
Login: Returning users may login to access their progress within the app.

1. User opens application
2. User clicks on Login button
3. User enters email and password
4. User is directed to Home Screen



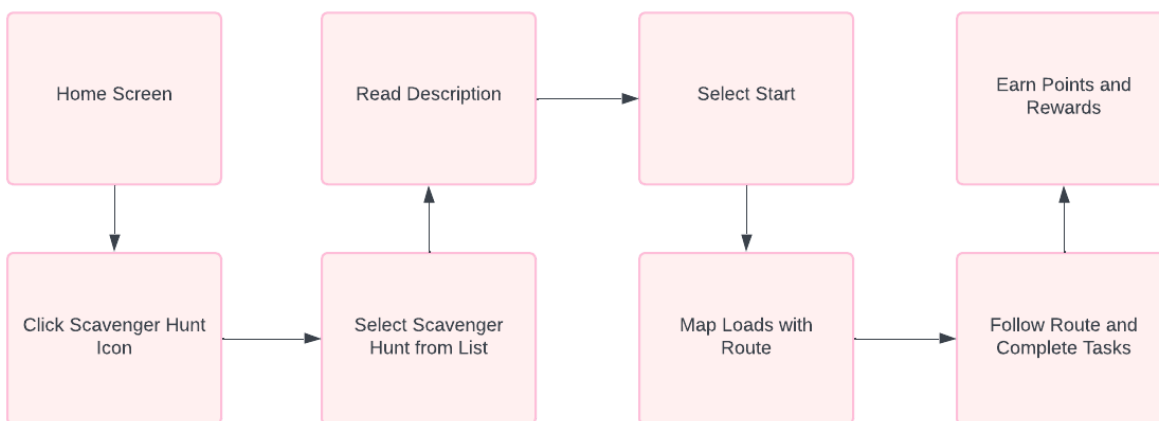
Using the Map: Users will view a map of the current location and see surrounding points of interest.

1. User logs in with their email and password and is directed to home
2. User clicks map icon
3. Map loads displaying their current location.
4. Nearby points of interest are displayed with pins or markers on map
5. User taps on pin to see more details about selected location



Participating in a Scavenger Hunt: User will be able to take part in scavenger hunts and complete tasks.

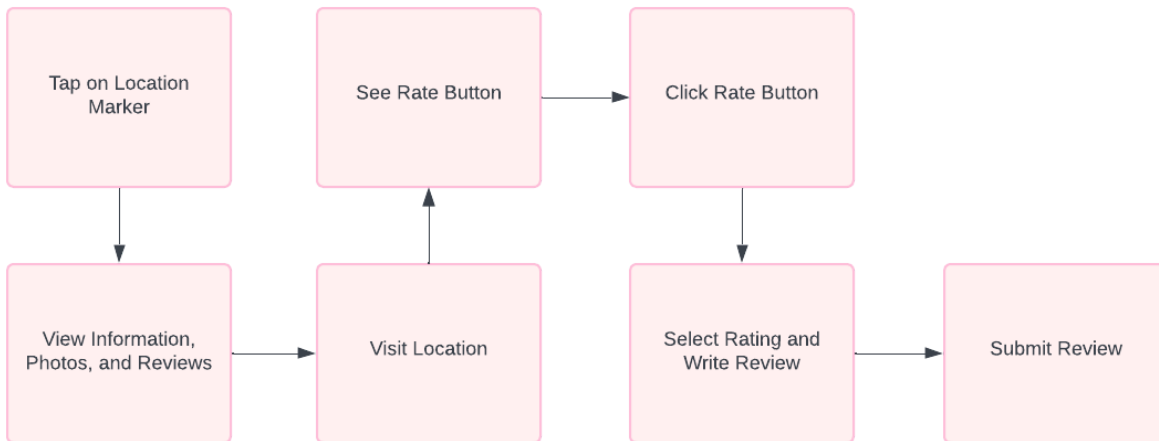
1. From the Home Screen user will click on Scavenger Hunt icon
2. User selects a scavenger hunt from a list
3. User can read the description and select Start
4. Map will load with the route on the map and show locations
5. user follows the selected route and complete the tasks
6. User earns points and rewards for each completed task and hunt.



Viewing and Rating Locations: User will be able to view detailed information about a location before visiting.

Users will be able to rate a location after visiting to share their experience with others in the app.

1. In the map a user will tap on a location marker to view the details
2. User will see information, photos and reviews
3. After visiting User will see a rate button
4. User clicks rate button
5. User is able to select a rating and write a review
6. User submits their review with Submit button



Using Augmented Reality* : User will be able to use their camera to view AR specific content in certain locations to allow for a more interactive experience.

1. User arrives to a Point of Interest
2. In the map the user selects AR button
3. Camera view opens with the AR overlay showing the name of the location
4. User is able to interact with on screen buttons to see more information about location.



4.2. User characteristics

Our ideal users are teens and older who want to see more places around them. People who love to travel and explore areas they didn't know about. And locals who want to see something new in their area.

4.3. User interface



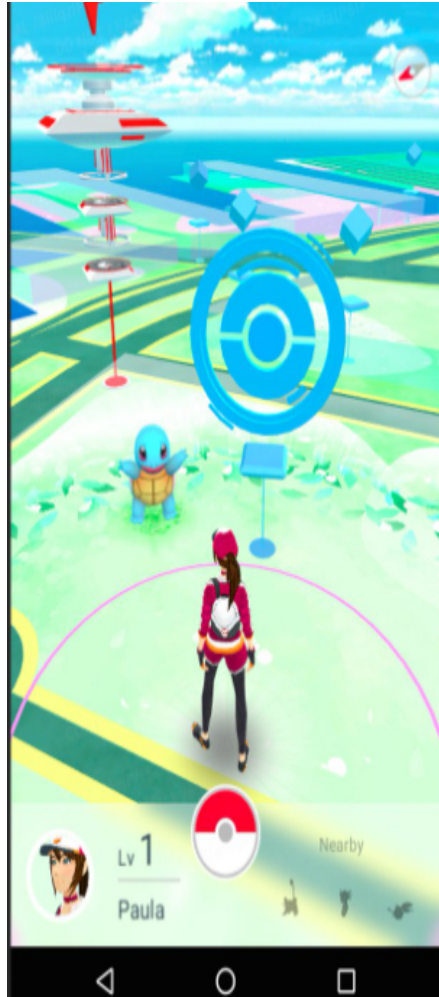
Google Field Trip

Layout could potentially be inspiration for how users will be able to see points of interest on the map in the city. This also shows an example of how a user may be able to view a filtered list of locations within the application.



Geocaching

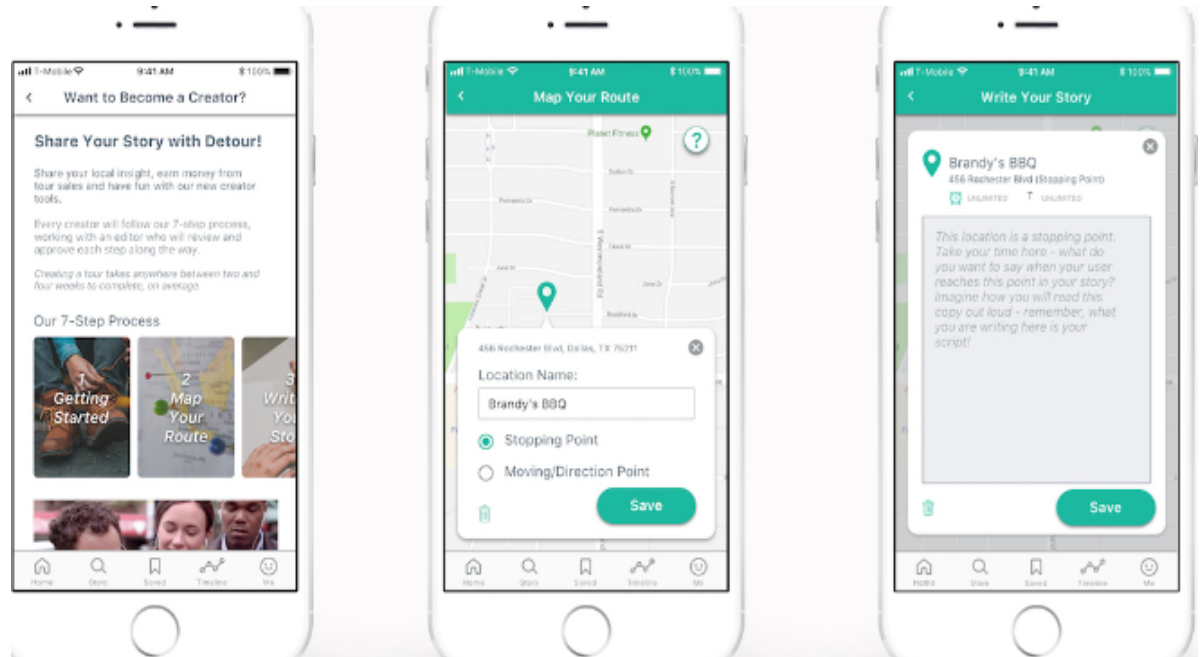
Potential view for a way a user may be able to find locations in a scavenger hunt. Users may be able to see how close they are to a location instead of being able to see it just on a map to add a more immersive aspect to the Scavenger Hunts.



Pokemon Go

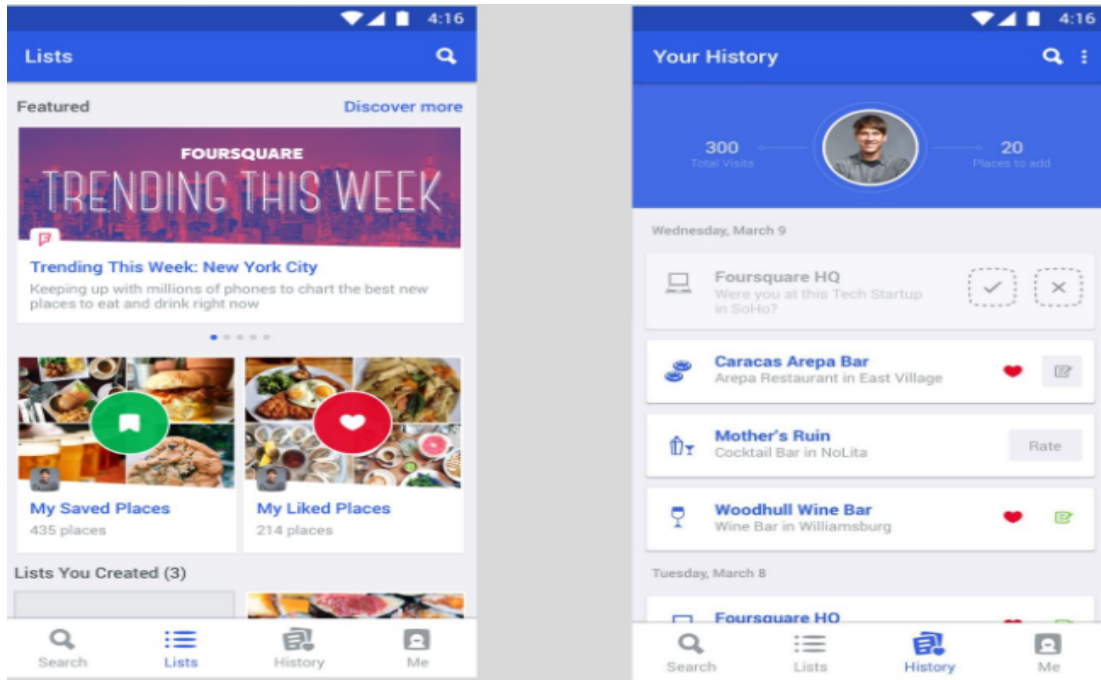
In one screen shot we see another potential option for Scavenger hunts, the places the user would need to go is also done based on vicinity but with a visual representation on the map to keep with an immersive experience but still be able to see the location displayed in the device.

In the following screenshot we see an example of how the AR may function. We see the physical location through the camera lens but also have the overlay where location name and information may be visible. The buttons along the bottom could be used in the case of ExploreXpert to display the information or possibly in relation to the scavenger hunts to progress further.



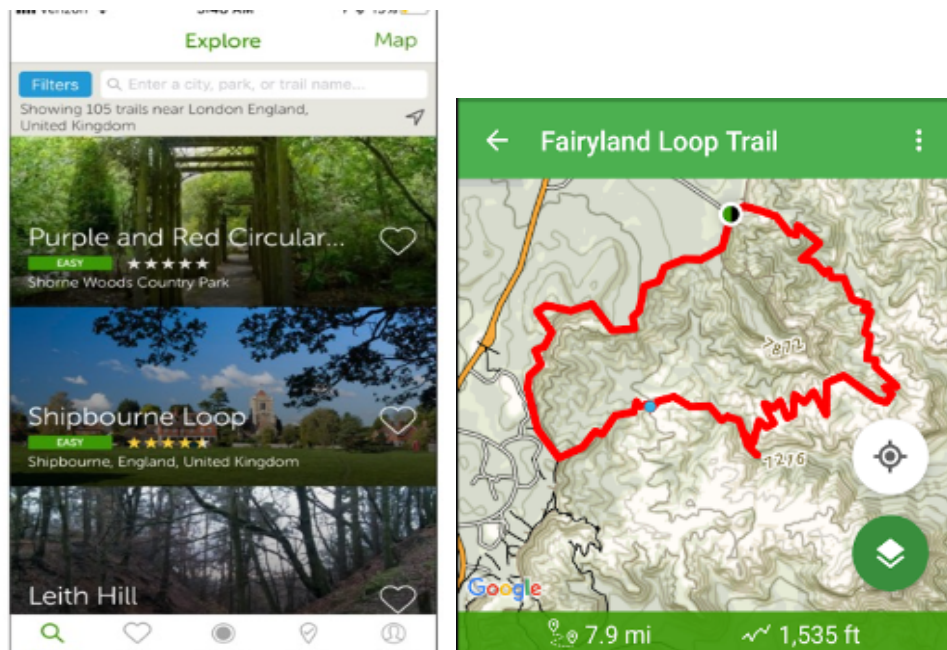
Detour

Here we see an example of a display for how users may contribute to the application. They could see the location they want to add a review for on the map and be able to select and add in their text from there. This would be able to be utilized directly in the map itself without having to move to a completely different screen to do so.



Foursquare

These screenshots provide examples of what it may look like to see a user profile where we may see items like points or achievements. Another thing to note is there is a user history displayed. In the case of ExploreXpert we may be able to see past scavenger hunts and places visited as well as we can see in the other image recommendations or lists based on what the user likes or places they have saved for later.



All Trails

The screenshots provide examples on how a user may be able to view a list of the preexisting Scavenger Hunts in an area and be able to view the route overtop a map after selecting. Number of stops may be included on the map for our purposes as well as adding pins where the stops are along the outline of the path. Ratings of the Scavenger Hunt path may be included like the screenshot as well as the like or favorite button may be able to be incorporated to pair with the saved lists like the previous image above.

5. Development Environment

Developer Tools:

- Android Studio Koala | 2024.1.1

Content Generation Tools:

- Adobe Photoshop/ Illustrator
- Maya
- 3DS Max

Data Tools:

- Firebase/Firestore
- MySQL Workbench
- Local Data Storage

3rd Party Dependencies and APIs:

- Android
- Android SDK with Java - Framework
- Firebase SDK
- Google Cloud - Google Maps, Places
- Android Jetpack Components
- Google Maps API
- Google Places API
- ARCore API

<TODO Create a diagram to show how your data will be organized in the database. If you are working with a NoSQL db consider reading this article to direct how you would approach this diagram as you will need to create a series of text JSON endpoints rather than an ER diagram shown below.>



6. Production Plans

Code Review Plan

Requisites for Review

- All functionality is complete and working
 - Has been tested on end device
 - Needs no further revision
- Meets all coding standard requirements
- Functions are properly commented
- You will need to test as a team BEFORE Build Reviews.
- All completed work needs to be present on the Development branch unless there is a good reason for otherwise.

Integration Plan

Source Control

- Git to be used for Version Control
- Major Branches:
 - Main: Will contain the production ready code that has been tested and ready for further development.
 - Development: Will contain integrated features/fixes before . Will have the latest changes before a release.
 - Feature Branches: Multiple branches used to develop specific features, will be branched off the development branch and merged back when completed.
 - Release Branches: Multiple branches used to prep for a new release. Used for last minute changes. Will be merged into main and development branches.
 - Bug Fix: branch to make quick patches to releases. Branched off of Main and merged back once patched.
 - Staging: Branch for testing integration of features and fixes before being merged back into Main. Will allow for testing before releasing.
 - Branch Creation and Merging
 - Feature branches will be created when a new feature is being developed.
 - Branches off Development branch and merged back into Development
 - Strategy: Will be merged into Development branch using a Pull Request. Must be approved by another developer before merging.
 - Release Branches will be created when the development branch is stable and is ready for release, for last minute testing and fixes.
 - Branches off from Development branch and merges into Development and Main branches.
 - Will be tagged with release number when merged
 - Bug Fix branches will be created when an urgent patch is required.
 - Branches off from Main and merged back into Main and Development Branches.
 - Strategy: Will be merged into Main and Development branches after the fix has been implemented and tested Pull Request is created and branch is reviewed before merging.
 - Staging Branches are an intermediate branch between Development and Main branches. It will allow for testing as multiple features are completed and tested.
 - Branches off of Development branch and merges back into Main.
 - Strategy: To be merged into Main after branch has been thoroughly tested and reviewed by development

team. Once all issues have been addressed and is stable branch will be pushed to Main with a release number attached. i.e 1.0.0

Testing Plans

Weekly Code Reviews

- The Day before Build Reviews with Mentor/Producer:
 - 1-2hrs - priority one
 - All team members will test all code completed during the work week
 - 15-30 min complete/incomplete discussion

Bug Database

- Maintained and updated by the team members
- Software
 - Jira

Outside Testing

- Month 4 of Capstone
- 5 Testers selected
- Crosstesting will be expected on a monthly basis with other teams in CS and GD
- A Testing Report Compiled
 - Scenarios planned
 - Findings
 - Surprises
 - Planned Solutions

7. Folder Hierarchy

