

CatMap

Test Report

Group 2:

Austin M Sill, Chaker Baloch, Devlin Hamill,
Eli Holter, Krai Pongrapeeporn, Vandan Amin

Department of Computer Science, Central Washington University
CS481 - Capstone Project

Dr. Donald Davendra

March 19, 2025

Contents

| | | |
|----------|------------------------------------|-----------|
| 1 | Introduction | 2 |
| 1.1 | Purpose | 2 |
| 1.2 | Scope | 2 |
| 1.3 | Glossary | 2 |
| 2 | Test Overview | 3 |
| 2.1 | Scope | 3 |
| 2.2 | Testing Environment | 3 |
| 3 | Test Results | 4 |
| 3.1 | Viewing Schedule | 4 |
| 3.2 | Classes and Class Events | 12 |
| 3.3 | Other Event | 12 |
| 3.4 | Directions | 27 |
| 3.5 | Locations | 30 |
| 3.6 | User Management | 35 |
| 3.7 | User Account | 39 |
| 3.8 | Settings | 42 |
| 3.9 | Future Fixes | 43 |
| 3.10 | Metrics | 43 |
| 4 | Conclusion | 43 |

1 Introduction

1.1 Purpose

The purpose of this project is to create a schedule-based campus navigational aid Android app for the Central Washington University (CWU) campus. This app can be used by students to map out where they need to go to attend classes per their schedule and can be used to navigate to other locations like retail stores and food courts.

1.2 Scope

There are a wide range of things that we can and cannot incorporate in this application based on given requirements and time restrictions. To stay on task and avoid feature creep, the minimum requirements and known restrictions are listed below:

1.2.1 Things that will be addressed in the app

- An interactive graphical user interface (GUI)
- Map of the CWU campus
- User profile functionality
- Sign-in and sign-up pages
- Schedule management
- Location view page and favorites managements
- Notifications for upcoming classes
- Directions from the user's location to the next scheduled event
- Daily, Weekly, and Monthly schedule viewing

1.2.2 Out of the scope of the app

- Map viewing and navigation outside of the CWU campus
- Google street view functionality.

1.3 Glossary

- Event: a singular period of time that represents scheduled reoccurring classes or singular events
- Application Programming Interface (API): A set of functions and commands provided by a software provider that allows connection and interaction with an external software component.
- Database: This will serve as shorthand for Firebase Cloud Firestore database.
- Inactivity: when something is not being interacted with by the user.
- Class: internally, classes will be represented as reoccurring scheduled events.

2 Test Overview

The purpose of this document is to summarize tests conducted to test the reliability of app components. Each component has a dedicated section displaying tests ran and their results. Metrics are provided for each group of tests per component, and a summary of metrics is provided in the conclusion.

2.1 Scope

The majority of the app components are developed by our team and can be tweaked to correct for errors and to suit our needs. However, there are components that we do not have direct control over, so they will not be tested and evaluated in this report.

2.1.1 Components In Scope of Testing

- Events: Creation, editing, deletion, pushing to and retrieving from the database
- Directions: Creation and life-cycle management of displayed routes from the user's location to their chosen destination
- Locations: Viewing location information and starting directions to that location
- Favorite Locations: Displaying, adding and removing locations to/from their favorite locations, and pushing changes to/receiving from the database
- User Management: creating and modifying user data and settings. Everything besides location data is stored in association to the user's account on the database

2.1.2 Components Out of Scope of Testing

- Non Buildings locations: the user cannot add a building name outside of CWU to the schedule.
- Hiding outside campus locations: The user cannot see locations outside the campus or are defocused
- Google login: the user will have the option to sign in with google
- Retrieving images: the user can attach images to their profile and locations will have images attached to specific buildings or areas.
- Addressing: addresses cannot be attached to the schedule
- Google street view: Street view cannot be accessible with google maps api.

2.2 Testing Environment

2.2.0.1 Emulator

Most of the testing was done through an emulator in Android Studio. the specifications for the virtual device are as follows.

- Model Name: sdk_gphone64_x86_64
- Android Version: 16

2.2.0.2 Phone

Some of the testing had to be done on a physical phone. This was necessary for interacting with GPS systems as the emulated version has no such hardware.

- Product Name: Galaxy S20 FE 5G
- Model Name: SM-G781U
- Software Version: 359823611567209
- Android Version: 13

3 Test Results

3.1 Viewing Schedule

3.1.1 Tests and Results

This will be viewing the schedule with specific events for daily, monthly, and recommending events.

3.1.1.1 Weekly View: All events we're displayed within the given week chronologically, the calendar was hidden and the add button is hidden to prevent updating errors.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|---|-------------|--------------------|---------------|
| 1.1 | Displaying events for the current week | passed | passed | Devlin Hamill |
| 1.2 | Displaying events for the selected week | passed | passed | Devlin Hamill |
| 1.3 | Not displaying events for the past end date | passed | passed | Devlin Hamill |
| 1.4 | Transition to monthly view working | passed | passed | Devlin Hamill |
| 1.5 | Repeating events not showing extra day | fail | fail | Devlin Hamill |

Table 1: Week View display Data

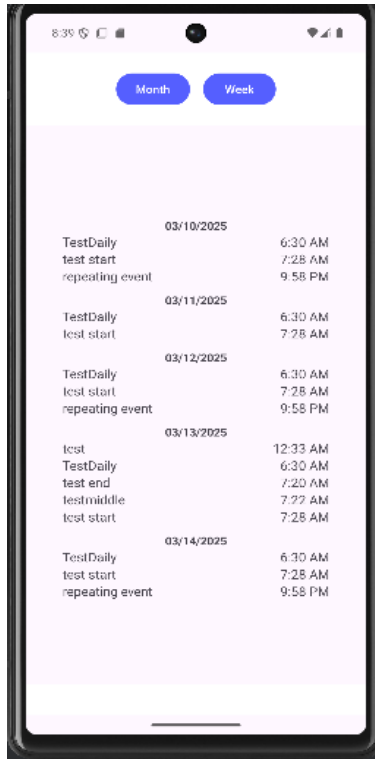


Figure 1: Weekly View displaying current week

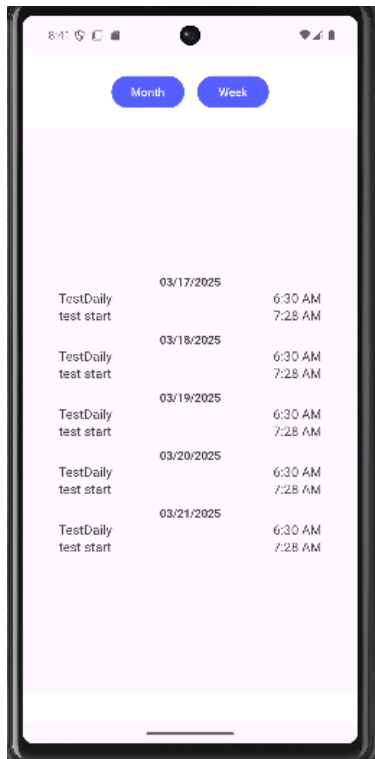


Figure 2: Weekly View displaying selected week

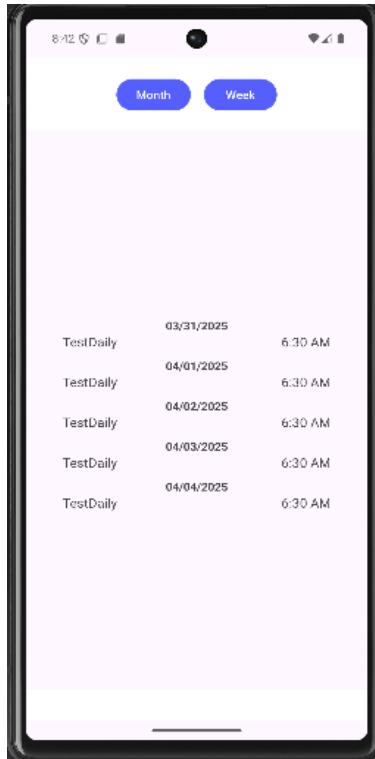


Figure 3: Not displaying events past end date

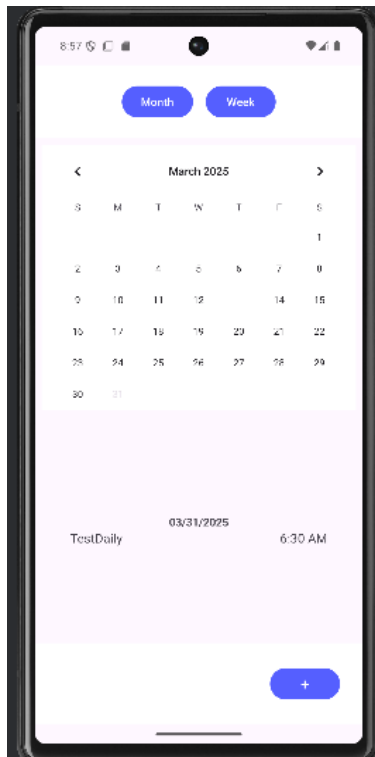


Figure 4: Transition to monthly view

3.1.1.2 Monthly View: Will be displaying the calender and show events for the current selected date.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|---|-------------|--------------------|---------------|
| 2.1 | Displaying an event for a current day | passed | passed | Devlin Hamill |
| 2.2 | Displaying events for the selected day | passed | passed | Devlin Hamill |
| 2.3 | Not displaying an event pasted end date | passed | passed | Devlin Hamill |
| 2.4 | Transition back to weekly view | passed | passed | Devlin Hamill |
| 2.5 | Repeating events not showing extra day | fail | fail | Devlin Hamill |

Table 2: Week View display Data

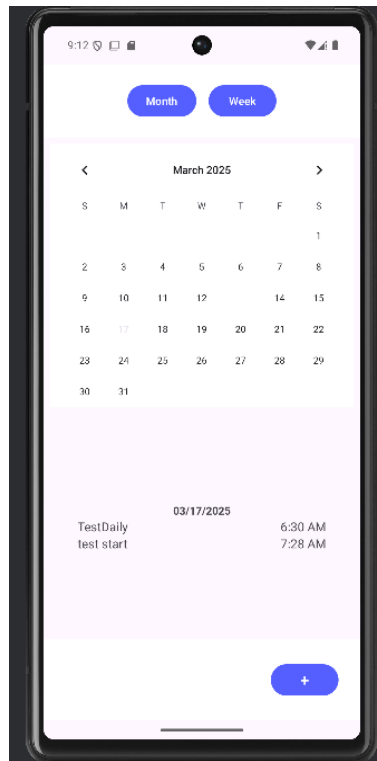


Figure 5: Monthly View displaying current day

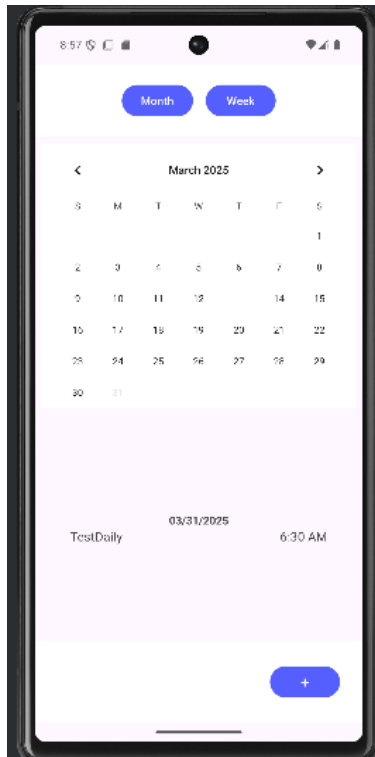


Figure 6: Monthly View displaying selected day



Figure 7: Not displaying events past end date

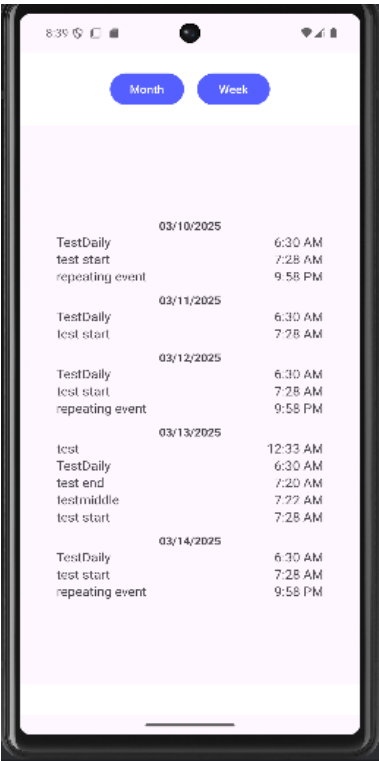


Figure 8: Transition to weekly view

3.1.1.3 Suggestions This will be testing the suggestions to make sure dates are being suggested for the week correctly.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|--|-------------|--------------------|---------------|
| 3.1 | Displaying suggestions for the week | passed | passed | Devlin Hamill |
| 3.2 | Displaying that there are no events | passed | passed | Devlin Hamill |
| 3.3 | Repeating events not showing extra day | fail | fail | Devlin Hamill |

Table 3: Week View display Data



Figure 9: Suggesting events within this week



Figure 10: Not suggesting events within this week

3.1.2 Future Fixes

One future fix we may include the color preference to the recycle view for each event to display the users color preference, but this is relatively small.

3.1.3 Metrics

| Test completion percentage | Test failure percentage | Test Execution Percentage | Total success rate |
|----------------------------|-------------------------|---------------------------|--------------------|
| 100 | 0 | 100 | 100 |

Table 4: View Metrics Data

3.2 Classes and Class Events

At the time of testing, this feature is still in development. Classes are groupings of events that allow a single class to contain multiple events, like repeating events for meetings on Mondays-Wednesdays at a specific room, meetings on Thursdays in a different room, and a one time event for the class final. Action will be taken to fast-track the development of this feature so it is ready for demonstration during the final presentation.

3.3 Other Event

3.3.1 Test and Results

This will contain tests on both the edit functionality and the the add functionality for non class events.

3.3.1.1 Add Making sure if all error checks are working properly and the user can enter a valid event.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|---|-------------|--------------------|---------------|
| 4.1 | Stoping entering no room number | passed | passed | Devlin Hamill |
| 4.2 | Stoping entering no event title | passed | passed | Devlin Hamill |
| 4.3 | Stoping entering no building name | passed | passed | Devlin Hamill |
| 4.4 | Stoping entering a invalid building name | passed | passed | Devlin Hamill |
| 4.5 | Checking if the user picked a time | passed | passed | Devlin Hamill |
| 4.6 | Checking if the user picked a color | passed | passed | Devlin Hamill |
| 4.7 | Checking if repeating events added successfully | passed | passed | Devlin Hamill |
| 4.8 | Adding a non class event | passed | passed | Devlin Hamill |
| 4.9 | Checking for smooth transitions | Fixed | passed | Devlin Hamill |
| 4.10 | Checking for colorpicker functionality | passed | passed | Devlin Hamill |
| 4.11 | Checking for timepicker functionality | passed | passed | Devlin Hamill |
| 4.12 | Checking for datepicker functionality | passed | passed | Devlin Hamill |
| 4.13 | Repeating events not showing extra day | fail | fail | Devlin Hamill |

Table 5: Adding non class event Data

A mobile application interface for creating an event. The screen has a white background with a light blue header bar containing a back arrow and the title "Event". Below the header, there are several input fields: "Event Title", "Building Name", and "Room Number". The "Room Number" field is empty. Below these fields is a "Repeating Event" toggle switch, which is currently turned off. Further down are date and time pickers showing "03/19/2025" and "Select Time". At the bottom of the form are two blue buttons: "Pick Color" and "Confirm". A green error message bubble at the bottom center says "Please enter a room number". The status bar at the top shows the time "12:48" and various system icons.

Figure 11: Not entering a room number

A mobile application interface for creating an event, similar to Figure 11. The "Room Number" field now contains the text "123". The "Event Title" field is empty. The "Repeating Event" toggle is still off. The date and time pickers remain the same. The blue "Pick Color" and "Confirm" buttons are still present. A green error message bubble at the bottom center now says "Please enter an event title". A vertical menu is visible on the left side of the screen, containing icons for a close button (X), a checkmark, a smiley face, and a hamburger menu. The status bar at the top shows the time "12:48" and various system icons.

Figure 12: Entering no event title

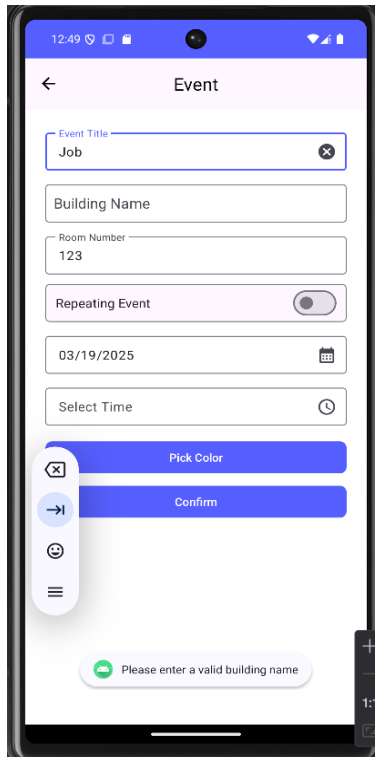


Figure 13: Entering no building name

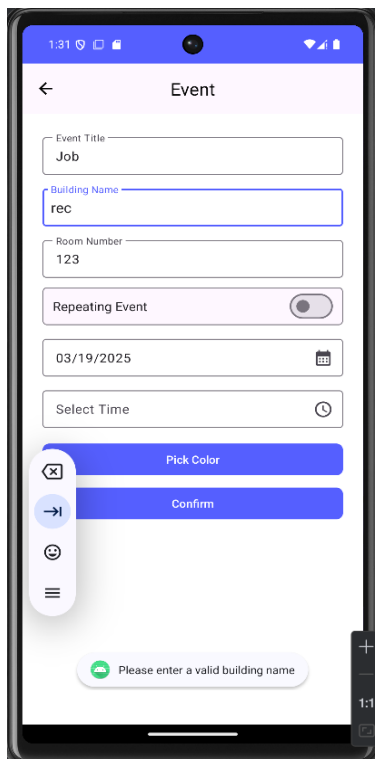


Figure 14: Entering a invalid building name

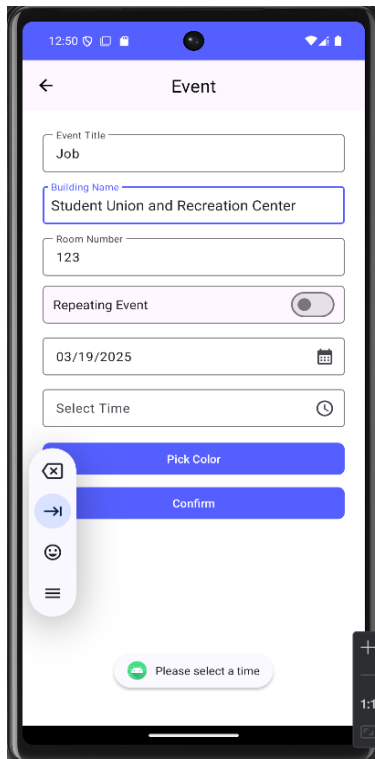


Figure 15: Check if time was picked

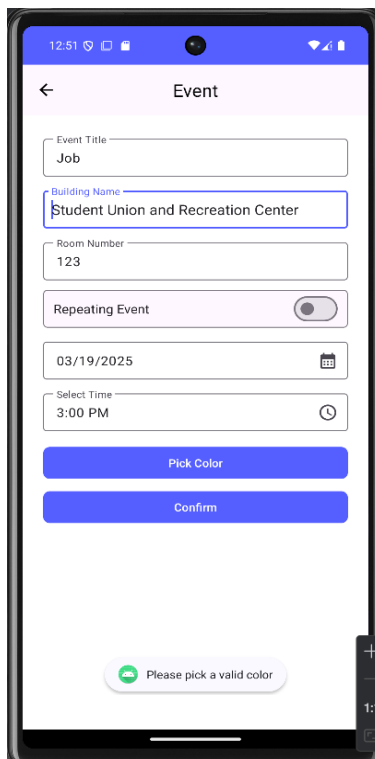


Figure 16: Checking a color was selected

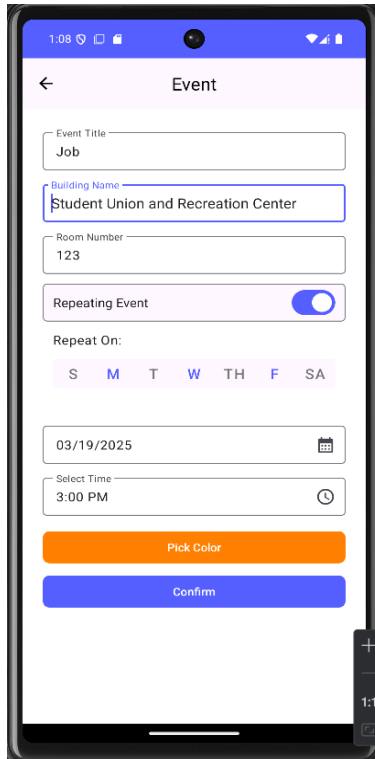


Figure 17: Adding event successfully

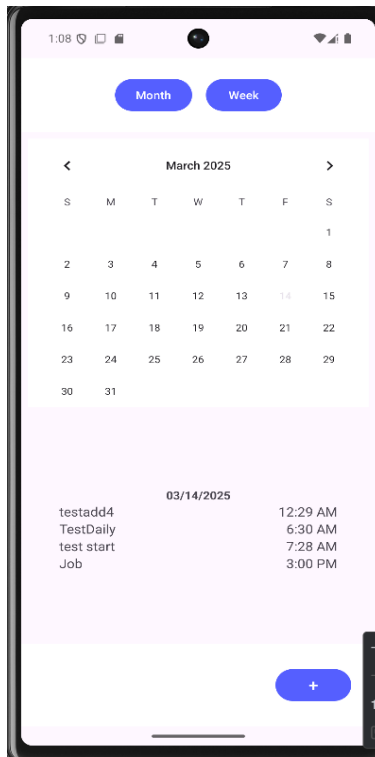


Figure 18: Transition to weekly view

3.3.1.2 Edit This will test the editing features displayed on the app,

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|--|-------------|--------------------|---------------|
| 5.1 | Stoping entering no room number | passed | passed | Devlin Hamill |
| 5.2 | Stoping entering no event title | passed | passed | Devlin Hamill |
| 5.3 | Stoping entering no building name | passed | passed | Devlin Hamill |
| 5.4 | Stoping entering a invalid building name | passed | passed | Devlin Hamill |
| 5.5 | Checking if the user picked a time | passed | passed | Devlin Hamill |
| 5.6 | Checking if the user picked a color | passed | passed | Devlin Hamill |
| 5.7 | Checking if repeating events edited successfully | passed | passed | Devlin Hamill |
| 5.8 | editing a non class event | passed | passed | Devlin Hamill |
| 5.9 | Checking for smooth between views transitions | passed | passed | Devlin Hamill |
| 5.10 | Checking for colorpicker functionality | passed | passed | Devlin Hamill |
| 5.11 | Checking for timepicker functionality | passed | passed | Devlin Hamill |
| 5.12 | Checking for datepicker functionality | passed | passed | Devlin Hamill |
| 5.13 | Repeating events not showing extra day | fail | fail | Devlin Hamill |

Table 6: Adding non class event Data

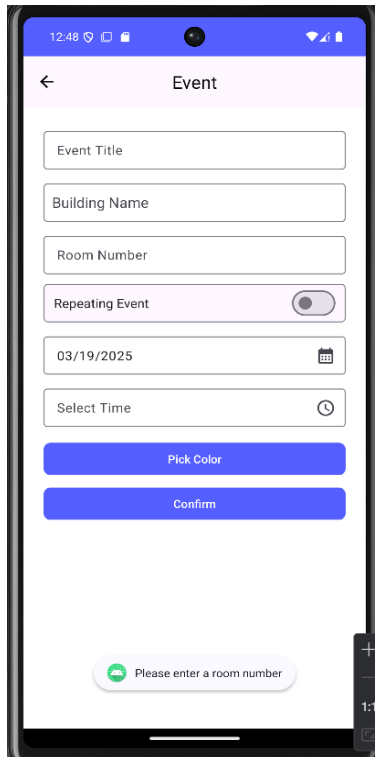


Figure 19: Not entering a room number

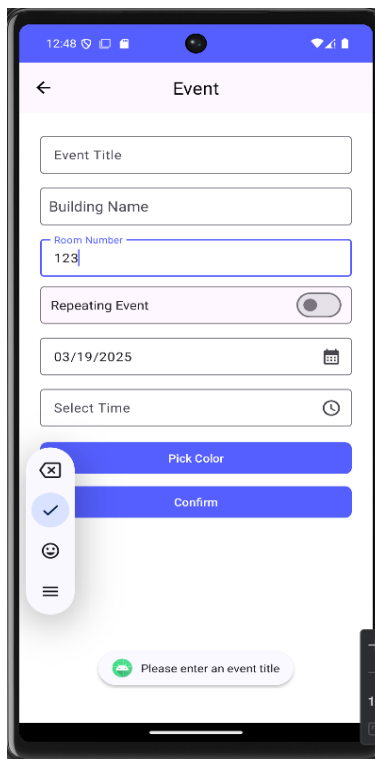


Figure 20: Entering no event title

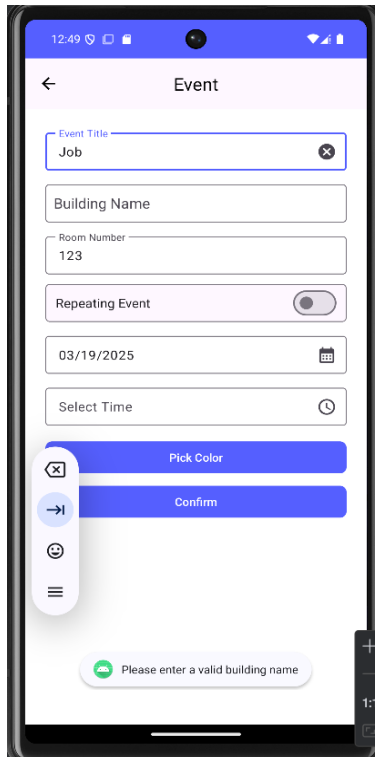


Figure 21: Entering no building name

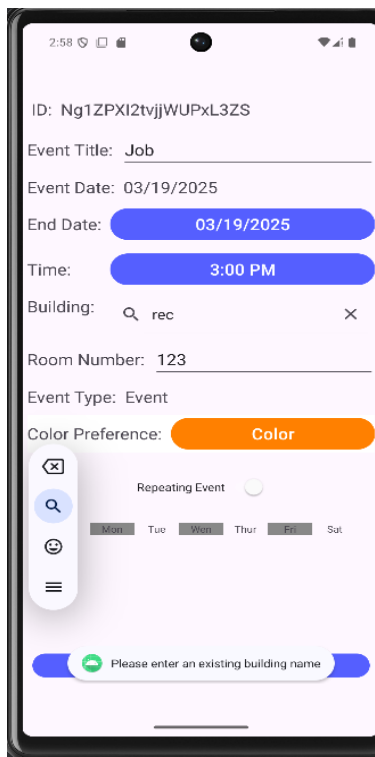


Figure 22: Entering a invalid building name

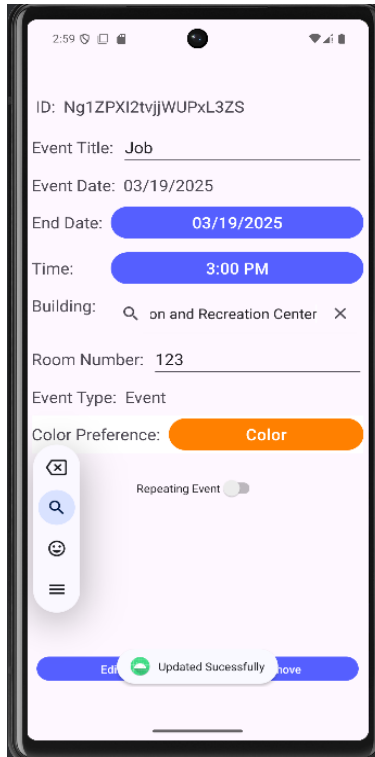


Figure 23: Updating repeating events successfully

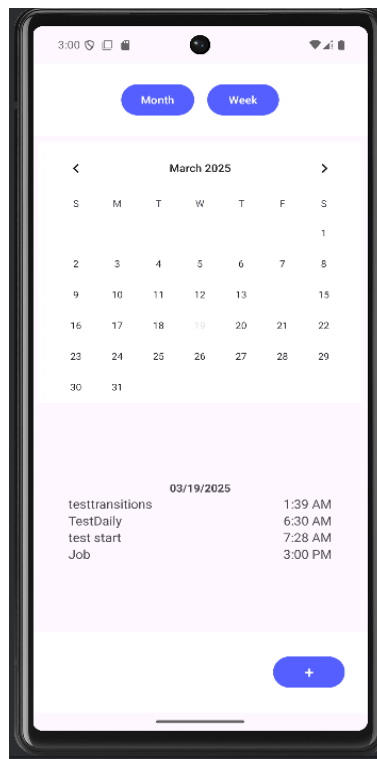


Figure 24: showing event on specific date as a one time event

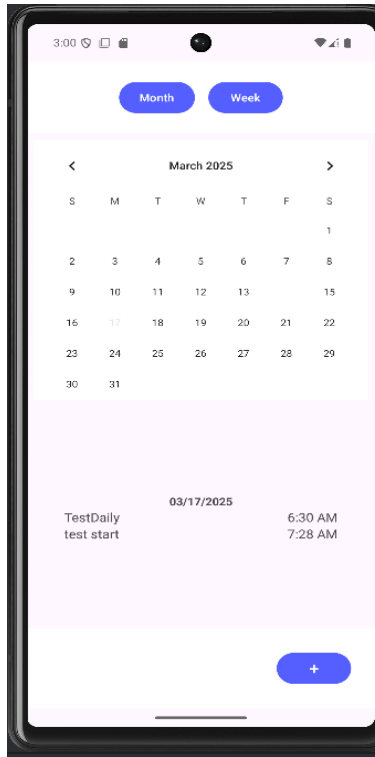


Figure 25: Event now missing cause its no longer a repeating event

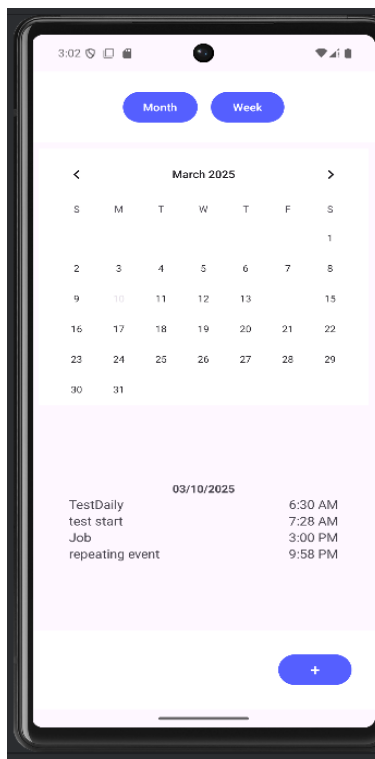


Figure 26: Updating back to repeating event

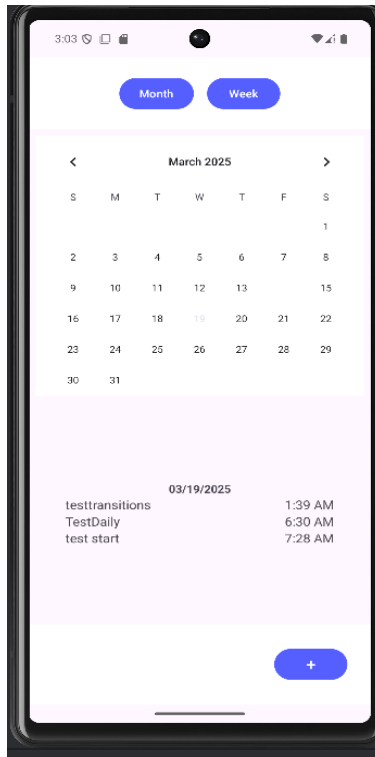


Figure 27: Removing event successfully

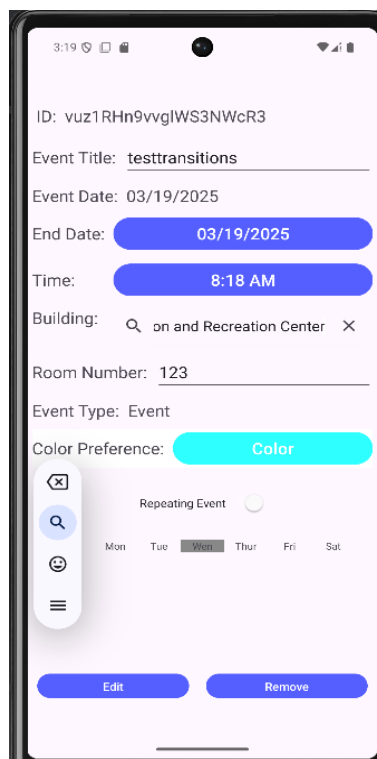


Figure 28: Initial event data

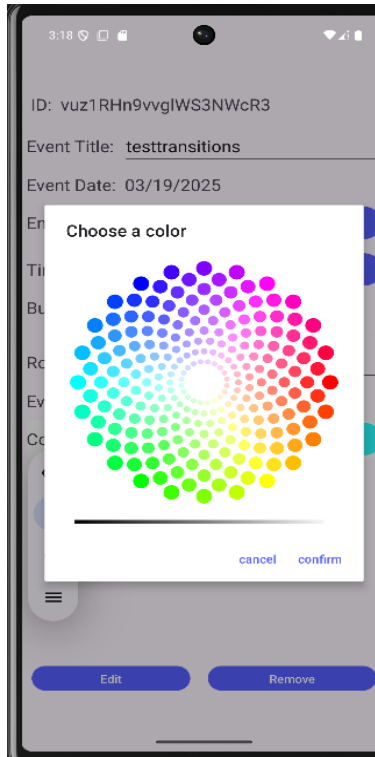


Figure 29: Color picker displaying successfully

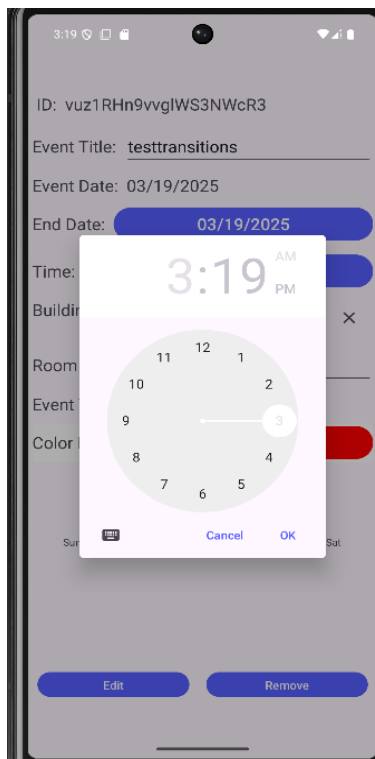


Figure 30: timepicker displaying successfully

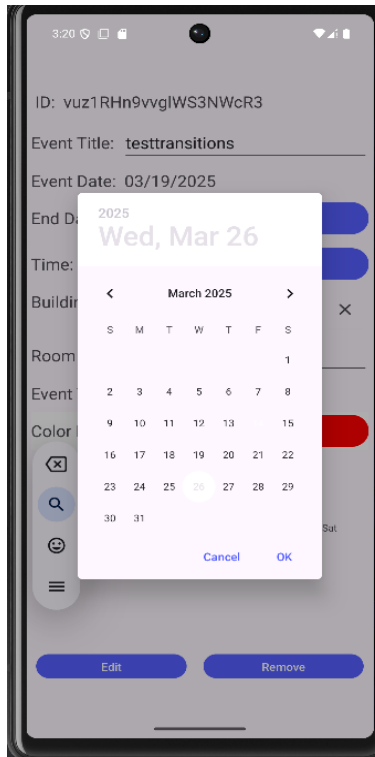


Figure 31: Datepicker displaying successfully

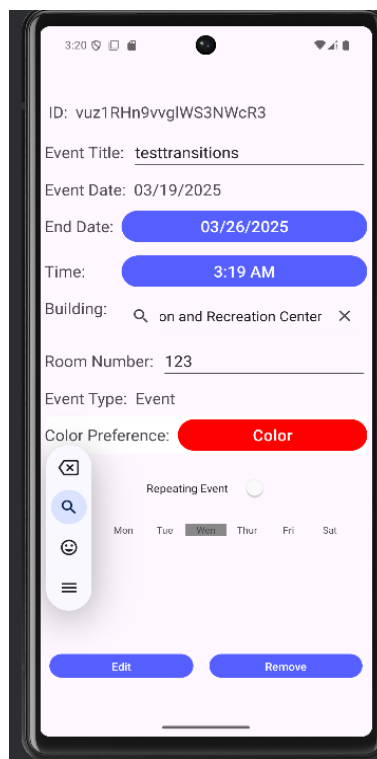


Figure 32: New Data

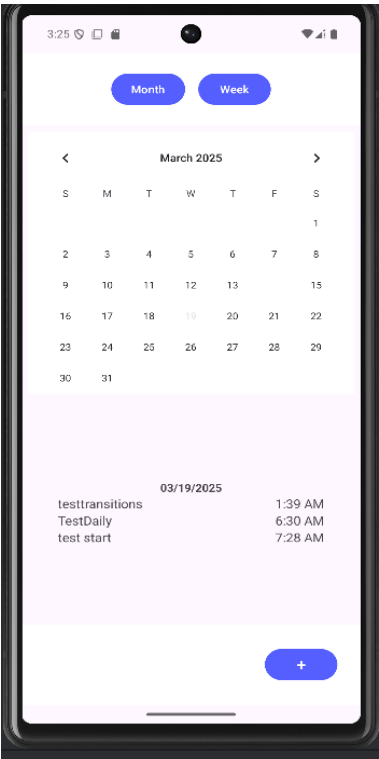


Figure 33: Event updating successfully

3.3.2 Future Fixes

We will be updating the UI of the edit page to make it more in line with the rest of the app.

3.3.3 Metrics

| | | | |
|----------------------------|-------------------------|---------------------------|--------------------|
| Test completion percentage | Test failure precentage | Test Execution Percentage | Total success rate |
| 96 | 4 | 100 | 98 |

Table 7: View Metrics Data

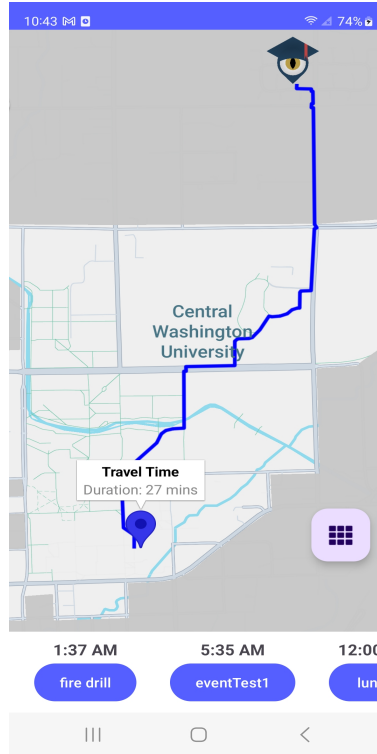


Figure 34: Overview of existing homepage

3.4 Directions

This section is designed to document the successes and failures in testing the direction finding acuity of the home page menu. It should be noted that this is a separate functionality from the ability to ask for directions from the locations menu. Here is a background routine that automatically routes the user to the next event. Because of this this function has a dual purpose of managing daily events.

3.4.1 Tests and Results

Below is a list of test that were run on the current operational model.

3.4.1.1 Display Map

Expected Outcome: Map is visible as soon as the app is opened.

Outcome: Map can be seen immediately upon the launch of the app. If the user is not logged in they will request the user log in or create an account before displaying map. This can be done with or without location permissions,

Pass/Fail status: Passed

Notes: None

3.4.1.2 Campus View

Expected Outcome: The campus will be visible from a top down view.

Outcome: Map displays a top down view of the CWU campus with no evident problems even without location permissions.

pass/Fail status: Passed

Notes: None

3.4.1.3 Show User Location

Expected Outcome: user location will be visible immediately upon app launch.

Outcome: Map displays user location as a unique pin so the user can differentiate it from locations. To enable this it is required that location permissions are turned on.

Pass/Fail status: Partial

Notes: Success will always be dependent on GPS permissions and user location.

3.4.1.4 Does Not Display off campus locations

Expected Outcome: Off campus locations are not displayed on the map.

Outcome: University property is the only place on the map that is not grayed out. Additionally it is impossible for the user to select non-designated locations. It should be noted that if you zoom in, white building outlines can be seen from outside of the campus property in the grayed out area.

Pass/Fail status: Partial

Notes: this may be able to be fixed but may also be an artifact of how the google maps API interacts with polygons.

3.4.1.5 Prevents off campus navigation

Expected Outcome: The user will be unable to navigate to off campus locations.

Outcome: The user is unable to navigate to non-designated locations. Some locations are not close to campus such as the air field. these are still navigable locations and are technically out of bounds.

Pass/Fail status: Pass

Notes: The issue of off campus location data might not be able to be resolved.

3.4.1.6 Auto Path Generation

Expected Outcome: The app will automatically generate a path to the user's next event.

Outcome: Map can immediately produce a path to the user's next event. This is taken from the users own date and time so if the user does not have their time set correctly it cannot predict the next path.

Pass/Fail status: Pass

Notes: the issue of users having incorrect time information on their phone will likely persist.

3.4.1.7 Daily Schedule

Expected Outcome: The daily schedule can be seen at the bottom of the page as soon as the app starts.

Outcome: Upon the launch of the app the daily schedule will be filled with all daily events. This information is presented based on the phone's date and time setting and may be incorrect if they are not set properly.

Pass/Fail Status: Pass

Notes: None

3.4.1.8 Event Order

Expected Outcome: the events will be ordered in accordance to when they occur that day.

Outcome: The daily scheduled events are ordered from earliest to latest for the convenience of the user. This ordering is unaffected by the phones date and time setting though it may not show the correct events despite having the correct order.

Pass/Fail status; Pass

Notes: None

3.4.1.9 Show event name and time

Expected Outcome: Event name and time are clearly visible

Outcome: The daily schedule easily displays both the name and time of each event that day.

Pass/Fail status: Pass

Notes: None

3.4.1.10 Remove past events

Expected Outcome: the daily schedule will automatically remove events as they are completed.

Outcome: Currently the schedule is unable to remove events from the daily schedule once they have passed.

Pass/Fail Status: Fail

Notes: None

3.4.1.11 Horizontal Scroll

Expected Outcome: The user should be able to scroll horizontally over their events that day.

Outcome: The list of daily events can scroll horizontally to accommodate days with more events. there appears to be no cap on the amount of events in a day that can fit on the daily timeline.

Pass/Fail status: Pass

Notes: None

3.4.1.12 Dynamic daily view

Expected Outcome: The daily view will be adjusted automatically as events are added or deleted.

Outcome: This function of the daily schedule is currently non-functional. the daily schedule does not automatically update as events are added or removed and the only way to make it update is to close the app and restart it.

Pass/fail status: Fail

Notes: None

3.4.1.13 Event Detail

Expected Outcome: The user should be able to click on an event name and be given more details about that event.

Outcome: Clicking on the Events listed in the daily schedule in it's current state does nothing. it ought to show a more in depth informative window about the event itself.

Pass/Fail status: Fail

Notes: None

3.4.1.14 Event Colors

Expected Outcome: All events shall display with the preferred color that the user selected upon event creation.

Outcome: Currently all events have the same color no matter which is selected when the event is created. This makes it hard to differentiate the events.

Pass/Fail status: Fail

Notes: None

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|-------------------------|-------------|--------------------|-------------|
| 3.4.1.1 | Display Map | Pass | Pass | Austin Sill |
| 3.4.1.2 | Campus View | Pass | Pass | Austin Sill |
| 3.4.1.3 | Show User Location | Partial | Partial | Austin Sill |
| 3.4.1.4 | Off Campus Hidden | Partial | Pass | Austin Sill |
| 3.4.1.5 | Off Campus Nav | Pass | Pass | Austin Sill |
| 3.4.1.6 | Auto Path Generation | Partial | Partial | Austin Sill |
| 3.4.1.7 | Daily Schedule | Pass | Pass | Austin Sill |
| 3.4.1.8 | Event Order | Pass | Pass | Austin Sill |
| 3.4.1.9 | Display Event Name/Time | Pass | Pass | Austin Sill |
| 3.4.1.10 | Remove Past Events | Fail | Pass | Austin Sill |
| 3.4.1.11 | Horizontal Scroll | Pass | Pass | Austin Sill |
| 3.4.1.12 | Dynamic Daily View | Fail | Pass | Austin Sill |
| 3.4.1.13 | Event Details | Fail | Pass | Austin Sill |
| 3.4.1.14 | Event Colors | Fail | Pass | Austin Sill |

Table 8: Direction tests

3.4.2 Future Fixes

In the future we intend to make sure that past events are removed from the daily view. This feature may be able to be made inaccurate by a user changing their date or time settings in their phone. Along the same line the daily view will be updated on a more regular basis the exact interval will need to be determined but it should be frequent enough to not cause the user any issues. the functionality will be further extended by allowing the user to click on the daily events to gain more information, in addition the events themselves will display the color the user chose on creation of the event.

3.5 Locations

To enable the use of path-finding with google maps, the app needed a way to track and associate location names with coordinates. The easiest option was to associate the two within the app using a json file to store location information like name and coordinate . This has some advantages and disadvantages but was the best option in this application.

| Advantages | Disadvantages |
|--|--|
| Navigation is locked to only locations we define beforehand. | All location information has to be gathered by hand and entered directly into the file. |
| Fewer and shorter database calls for location information. | Storing and working with large files can hog system resources and slow things down if not handled carefully. |

3.5.1 Tests and Results

3.5.1.1 Search Locations View a scroll-able, sectioned list of locations and identify locations that the user has added to their favorites. The list should be filterable using a search bar and location information should be accessible by clicking on a location in the list.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|---|-------------|--------------------|------------|
| 12.1 | User can see a list of all locations in the app as well as their favorite locations | passed | passed | Eli Holter |
| 12.2 | User can use search bar to find specific locations | passed | passed | Eli Holter |
| 12.3 | Clicking the 'X' clears the search bar and stops filtering | passed | passed | Eli Holter |
| 12.4 | Matching locations display as the user types | passed | passed | Eli Holter |
| 12.5 | Clicking a location takes navigates to that locations information page | passed | passed | Eli Holter |

Table 9: Testing location search page functionality

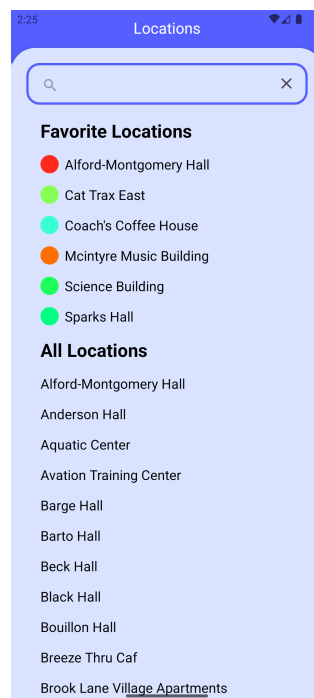


Figure 35: Viewing all locations and favorites

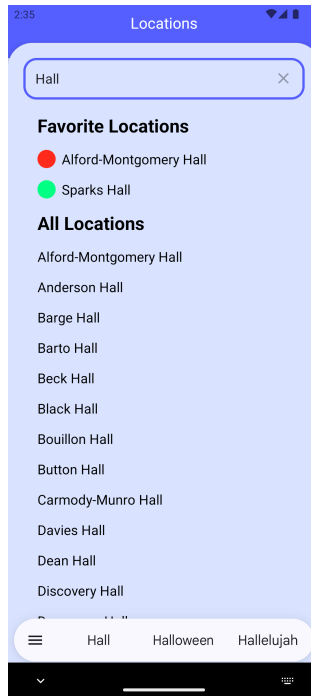


Figure 36: Filtering shown locations by name

3.5.1.2 Location Information View a scroll-able information page containing relevant information for any given location. This page contains the location name, it's address, it's main coordinate, a picture of the location, and a short description of it's purpose. From this page the user can navigate to this location and add/remove it to/from the favorites.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|--|-------------|--------------------|------------|
| 11.1 | User can view detailed location information | passed | passed | Eli Holter |
| 11.2 | User can view an image of the location | passed | passed | Eli Holter |
| 11.3 | Navigate to location button is visible and clickable | passed | passed | Eli Holter |
| 11.4 | Add to/remove from favorite location button is visible and clickable | passed | passed | Eli Holter |

Table 10: Viewing location data and buttons

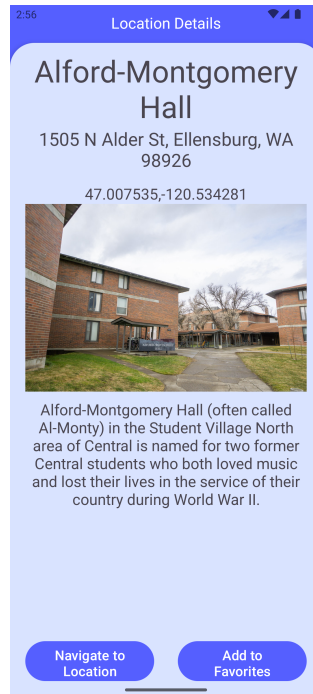


Figure 37: Displaying location information and picture, as well as navigation and favorites buttons

3.5.1.3 Favorite Locations Add and remove location to and from the users favorite locations. Favorite locations will be displayed at the top of the location search page. The user can add/remove locations to/from their favorite locations by clicking the corresponding button on the location information page. The buttons text will change to reflect the locations presence in favorite locations.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|---|-------------|--------------------|------------|
| 10.1 | Favorite locations are shown at the top of the location search page | passed | passed | Eli Holter |
| 10.2 | Location can be added to favorite locations using a button | passed | passed | Eli Holter |
| 10.3 | Location can be removed from favorite locations using a button | passed | passed | Eli Holter |

Table 11: Displaying favorite locations and favorite management

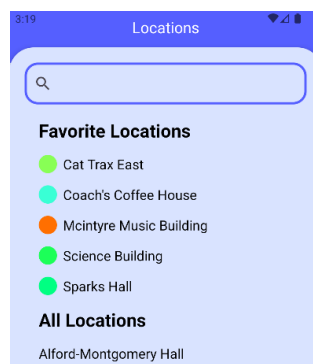


Figure 38: Displaying favorite locations on location search screen



Figure 39: Favorite button when location is not a favorite

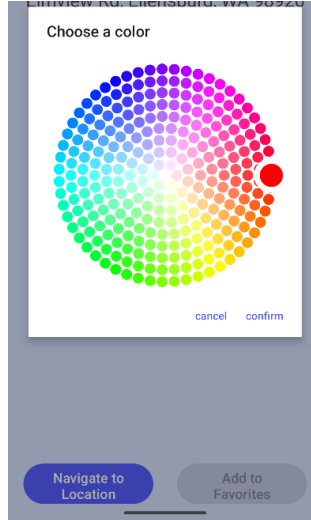


Figure 40: Select a color to add location to favorites



Figure 41: Favorite button when location is a favorite

3.5.1.4 Navigate to Location When the 'navigate to location' button is clicked, the user will be taken to the main map page and a route will be drawn from their current location to the location. The user can end navigation by clicking the 'end navigation' button.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|--|-------------|--------------------|------------|
| 13.1 | Clicking 'Navigate to Location' button launches the map screen with route to location | failed | passed | Eli Holter |
| 13.2 | Clicking 'End Navigation' removes route to location, hides itself, and begins navigation to next scheduled event | failed | passed | Eli Holter |

Table 12: Navigation to selected location

3.5.2 Metrics

- Total number of tests: 14
- Number of tests failed: 2
- Number of tests passed: 12

| Test Completion % | Test failure % | Test Success % | Result vs Expectation % |
|-------------------|----------------|----------------|-------------------------|
| 100.0% | 14.3% | 85.7% | 100.0% |

Table 13: View Metrics Data

3.5.3 Future Fixes

At the time of testing, the navigate to location feature is still in development. Action will be taken to fast-track the development of this feature so it will be ready for demonstration during the final presentation.

3.6 User Management

This section will cover tests for the following features: sign in, sign up, profile page, change password, and change username. User settings/preference will not be included in this section and will be in the "Settings" section instead.

3.6.1 Tests and Results

3.6.1.1 Sign up

3.6.1.1.1 Sign Up button visibility

Expected outcome: Sign Up button is visible on the Launcher page

Test outcome: Sign Up button is obstructed behind the phone's software buttons on the bottom

Pass/fail status: Partial

Notes: This test was done on a Pixel 8 (API 30, emulator).

3.6.1.1.2 Sign Up page accessibility

Expected outcome: Sign Up page is accessible from the Launcher page

Test outcome: Sign Up page can be accessed from the Launcher page by a button

Pass/fail status: Pass

Notes: This relies on the button, which is obstructed.

3.6.1.1.3 Sign Up with valid inputs

Expected outcome: Sign Up is successful with valid inputs. Valid e-mails formats must contain an at sign, have an address prefix (such as .com), and must only have letters or numbers. Valid password formats are the following: at least 6 characters long, 0-9, A-Z, special characters from a chosen set.

Test outcome: Sign Up was successful with the e-mails: "test@3621.com" and "john@test.com" and the passwords: "Password1!" and "P@ssword1".

Pass/fail status: Pass

Notes: None.

3.6.1.1.4 Sign Up with a profile picture

Expected outcome: The user can choose to use a profile picture when they sign up, and it will be displayed in their profile. During sign up, they can press on the default profile picture and their Android Gallery app will open, prompting them to choose a picture on their local storage.

Test outcome: A sign up attempt with valid inputs was made, with the e-mail "kraitest@test.com", then pressing the default profile picture on the sign up page prompted Android Gallery to open, showing all pictures on the storage. Pressing on a picture will close Android Gallery and the user returned to the app, with the chosen photo now selected as their profile picture, which is automatically scaled and fitted into a round display.

Pass/fail status: Pass

Notes: None.

3.6.1.1.5 Sign Up with invalid e-mail and/or password

Expected outcome: Sign Up fails with invalid inputs. Refer to test case 3.6.1.1.3 for what constitutes valid inputs, anything outside of that is considered invalid. An error message should show and explain to the user how to fix the problem.

Test outcome: Sign Up with invalid inputs failed, invalid e-mails tested were: "aa!!!@gmail.com", "test@gmail", "AA@gmail.com", "testtest", and an empty e-mail field. Invalid passwords tested were: "password", "Password", "Password.\$\$\$22", and an empty password field. An error message displayed for invalid e-mails, and invalid passwords.

Pass/fail status: Pass

Notes: Error messages do not update properly, if invalid inputs were given, the errors remain even after fixing them if a new invalid input was given.

3.6.1.1.6 Sign Up with an existing e-mail

Expected outcome: Sign Up fails when the user tries to sign up with a duplicate e-mail, in other words an e-mail that has already been registered with the system. An error message should show and explain to the user how to fix the problem.

Test outcome: An account with the e-mail "kraitest@test.com" was created, and another attempt to register with the same e-mail failed. No error message appears, and nothing happens when pressing "Create Account" with the duplicate e-mail.

Pass/fail status: Partial

Notes: Partial status because there was no error message displayed. FirebaseAuth automatically handled and denied the duplicate e-mail registration, the current frontend error catcher failed.

3.6.1.1.7 Sign Up with mismatched passwords

Expected outcome: Sign Up fails when the user doesn't confirm their password correctly. An error message should show and explain to the user how to fix the problem.

Test outcome: A valid e-mail and username was provided, a valid password "Password1!" with an invalid password confirmation "Password1!2" was inputted and failed. An error message displayed indicating that the password confirmation failed.

Pass/fail status: Pass

Notes: None.

3.6.1.1.8 Sign Up with missing fields

Expected outcome: Sign Up fails if the user hasn't entered all of the required fields: username, e-mail, password, and password confirmation.

Test outcome: The inputs username "test", e-mail "testkrai2@testkrai.com", password "Password1!", confirmation password "Password1!" were typed into the fields, and then removed one by one starting from username to confirmation password. In each step, a registration was attempted, none of them succeeded and proper error messages indicate what fields are missing.

Pass/fail status: Pass

Notes: None.

3.6.1.1.9 Navigation back to Launcher from Sign Up

Expected outcome: When the user gets to the Sign Up page, they can navigate back to the Launcher if they accidentally navigated to the Sign Up page.

Test outcome: A button at the bottom of the Sign Up page is highlighted and clicking it returns the user to the Launcher page.

Pass/fail status: Pass

Notes: The physical Android Back button also works.

3.6.1.1.10 Signing up using Google Accounts

Expected outcome: The user is able to sign up using a Google Account, if they've logged into one on Google Play Services. If they have not logged into Google Play Services, they will be prompted to. Pressing "Sign in with Google" will automatically register a Google Account on the CatMap app.

Test outcome: A phone without a Google account logged in was, after pressing "Sign in with Google" the user is redirected to the Google Play Services login flow. The Gmail account "wimcd1996@gmail.com" was used to login to Google Play Services, and registration with CatMap was successful. After registration, the user was redirected to the main page.

Pass/fail status: Pass

Notes: None.

3.6.1.1.11 Signing up using Google Accounts with a profile picture

Expected outcome: The user is able to sign up using a Google Account that has a profile picture, and the CatMap app will automatically retrieve that picture and set it as their CatMap profile picture.

Test outcome: The Gmail account "wimcd1996@gmail.com" was used to login to Google Play Services, and registration with CatMap was successful. After registration, the user was redirected to the main page. Navigating to the Profile page, the profile picture shown in the Profile page is the same one as the Google Account picture.

Pass/fail status: Pass

Notes: None.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|--------------------------------------|-------------|--------------------|---------------------|
| 3.6.1.1 | Sign Up button visibility | Partial | Pass | Krai Pon-grapeeporn |
| 3.6.1.2 | Sign Up page accessibility | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.3 | Sign Up with valid inputs | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.4 | Sign Up with a profile picture | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.5 | Sign Up with invalid input | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.6 | Sign Up with an existing e-mail | Partial | Pass | Krai Pon-grapeeporn |
| 3.6.1.7 | Sign Up with mismatched passwords | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.8 | Sign Up with missing fields | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.9 | Back navigation from Sign Up | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.10 | Signing up using Google Accounts | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.1.11 | Sign up with Google with profile pic | Pass | Pass | Krai Pon-grapeeporn |

Table 14: Sign up tests

3.6.1.2 E-mail Sign In

3.6.1.2.1 E-mail Sign In from the Launcher page

Expected outcome: On the Launcher page, the user can sign in if they already have an existing account that they created with the Sign Up page. The user will be redirected to the main page after inputting a valid e-mail and password, and pressing "Sign In"

Test outcome: An account "test3621@gmail.com" with the password "Password1!" was created and used to successfully login, and the user was redirected to the main page.

Pass/fail status: Pass

Notes: None.

3.6.1.2.2 E-mail Sign In with invalid e-mail and/or password

Expected outcome: Sign In fails when the user enters an invalid e-mail, one that was not registered with the service, or an invalid password with either a valid or invalid e-mail.

Test outcome: Attempts were made with the following combination: "hellohello@hello.com" (unregistered e-mail) with "Password1!", empty e-mail with empty password, "test@test.com" (existing e-mail) with password "password", and "test@test.com" with empty password. Error messages were given for an invalid input (wrong e-mail, wrong password) or missing input.

Pass/fail status: Pass

Notes: None.

3.6.1.2.3 Google Account Sign In

Expected outcome: The user is able to sign in on a Google Account, if they've registered one with the CatMap app beforehand. Pressing on "Sign in with Google" while logged in on Google Play Services with a valid Google account will automatically log them in and navigate them to the main page.

Test outcome: The Gmail account "wimcd1996@gmail.com" was registered successfully beforehand on the CatMap app. "wimcd1996@gmail.com" was used to log into Google Play Services and pressing "Sign in with Google" on the Launcher page navigated the user to the main page.

Pass/fail status: Pass

Notes: None.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|-----------------------------------|-------------|--------------------|---------------------|
| 3.6.2.1 | E-mail Sign In | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.2.2 | E-mail Sign In with invalid input | Pass | Pass | Krai Pon-grapeeporn |
| 3.6.2.3 | Google Account Sign In | Pass | Pass | Krai Pon-grapeeporn |

Table 15: Sign in tests

3.6.2 Future Fixes

- The Sign Up button will be repositioned so that non of the button is obstructed by other UI elements.
- An error message will display when the user attempts to sign up with a duplicate e-mail, and tell the user how to correct the issue.

3.6.3 Metrics

Test Summary Report

- Total Tests Conducted: 13
- Tests Passed: 11
- Tests with Partial Pass: 2

| | | | |
|----------------------------|-------------------------|---------------------------|--------------------|
| Test completion percentage | Test failure percentage | Test Execution Percentage | Total success rate |
| 84% | 16% | 100% | 92% |

Table 16: Sign In Metrics Data

3.7 User Account

This section will cover tests for the following features: profile page, change password, and change username. These functionalities are crucial for user management, ensuring proper authentication, secure data handling, and seamless UI interaction.

3.7.1 Tests and Results

3.7.1.1 Profile Page

3.7.1.1.1 Profile Page Accessibility

Expected outcome: The user should be able to access the profile page after signing in from main page.

Test outcome: The profile page is accessible from the main screen.

Pass/fail status: Pass

Notes: None.

3.7.1.1.2 Display User Information

Expected outcome: The user's name, email, and profile picture should be correctly displayed.

Test outcome: User's name and email were retrieved and displayed correctly. Profile picture was loaded from Firestore.

Pass/fail status: Pass

Notes: Verified by comparing UI elements with Firestore data.

3.7.1.1.3 Profile Picture Update

Expected outcome: User can change their profile picture via gallery selection, and the updated image should be stored in Firestore.

Test outcome: Profile picture was updated and displayed correctly. The new image was successfully stored in Firestore.

Pass/fail status: Pass

Notes: The edit functionality replaces the existing image in firebase, only JPEGs allowed.

3.7.1.1.4 Profile Listener for Updates

Expected outcome: If the profile is updated from another session, the changes should reflect in real-time.

Test outcome: The profile information and picture were updated in real-time when modified from another device.

Pass/fail status: Pass

Notes: None.

3.7.1.1.5 Logout Functionality

Expected outcome: User should be signed out and redirected to the login page.

Test outcome: User was logged out successfully, session cleared, and login page appeared.

Pass/fail status: Pass

Notes: None.

3.7.1.2 Change Password Button

3.7.1.2.1 Navigate to Change Password Page

Expected outcome: The profile page button, 'Update Password' navigates the user to Change Password page.

Test outcome: The button correctly works and navigates user into change password page.

Pass/fail status: Pass

Notes: None.

3.7.1.3 Change Password

3.7.1.3.1 Validate Password Strength

Expected outcome: Weak passwords should be rejected with appropriate error messages.

Test outcome: Weak passwords (less than 6 characters, missing numbers, missing special characters) were correctly rejected.

Pass/fail status: Pass

Notes: None.

3.7.1.3.2 Mismatched Password Confirmation

Expected outcome: If the confirmed password does not match the new password, an error message should be displayed.

Test outcome: Mismatched password entries triggered an appropriate error message.

Pass/fail status: Pass

Notes: None.

3.7.1.3.3 Re-authenticate Before Password Change

Expected outcome: The system should prompt the user for re-authentication before updating the password.

Test outcome: The user was required to re-authenticate. Incorrect credentials resulted in an error message, 'Incorrect current password'.

Pass/fail status: Pass

Notes: None.

3.7.1.3.4 Successful Password Update

Expected outcome: After passing authentication, the new password should be stored in Firebase, and the user is navigated back to the profile page.

Test outcome: Password updated successfully in Firebase, and the user is directed to profile page without any confirmation message

Pass/fail status: partial

Notes: No confirmation message displayed for password change.

3.7.1.4 Change username Button

3.7.1.4.1 Navigate to Edit Username Page

Expected outcome: The profile page button, 'Update Username' navigates the user to Edit username page.

Test outcome: The button correctly works and navigates user into Edit username page.

Pass/fail status: Pass

Notes: None.

3.7.1.5 Change Username

3.7.1.5.1 Validate Empty Username Input

Expected outcome: The system should prevent submission of an empty username and display an error message.

Test outcome: An error message "username cannot be empty" was displayed when trying to submit an empty username.

Pass/fail status: Pass

Notes: None.

3.7.1.5.2 Update Username Successfully

Expected outcome: The new username should be updated in Firestore and reflect in the UI and the user is directed to profile page where the new username is shown once the confirmation button is clicked.

Test outcome: The username was updated and reflected correctly in Firestore and in profile page.

Pass/fail status: Pass

Notes: None.

3.7.1.5.3 Handle Firestore Update Failure

Expected outcome: If Firestore update fails, the error should be logged and displayed to the user.

Test outcome: No error message is displayed.

Pass/fail status: Fail

Notes: None.

| Test number | Test Description | Test Status | Expectation Status | Tester |
|-------------|--|-------------|----------------------------------|---------------|
| 3.6.2.1 | Profile page accessibility | passed | passed | Chakar Baloch |
| 3.6.2.2 | User details display correctly | passed | passed | Chakar Baloch |
| 3.6.2.3 | Profile picture updates correctly | passed | passed | Chakar Baloch |
| 3.6.2.4 | Profile listener updates in real-time | passed | passed | Chakar Baloch |
| 3.6.2.5 | Logout redirects user to login screen | passed | passed | Chakar Baloch |
| 3.6.2.6 | Navigate to Change Password page | passed | passed | Chakar Baloch |
| 3.6.2.7 | Password validation rejects weak passwords | passed | passed | Chakar Baloch |
| 3.6.2.8 | Password confirmation must match | passed | passed | Chakar Baloch |
| 3.6.2.9 | User re-authentication required for password change | passed | passed | Chakar Baloch |
| 3.6.2.10 | Password successfully updates in Firebase | partial | failed (no confirmation message) | Chakar Baloch |
| 3.6.2.11 | Navigate to Edit Username page | passed | passed | Chakar Baloch |
| 3.6.2.12 | Empty username submission blocked | passed | passed | Chakar Baloch |
| 3.6.2.13 | Username updates correctly in Firestore | passed | passed | Chakar Baloch |
| 3.6.2.14 | Firestore username update failure (no error message displayed) | failed | failed | Chakar Baloch |

Table 17: Profile, Change Password, and Change Username Tests

3.7.2 Future Fixes

- Improve UI feedback for Firestore failures by displaying appropriate error messages when updates fail.
- Implement confirmation messages for successful password updates to enhance user experience.
- Ensure Firestore update failures (e.g., due to network issues or permissions) provide proper user notifications.
- Conduct additional UI/UX testing to verify that navigation between profile and update pages is intuitive.

3.7.3 Metrics

3.7.3.1 Test Execution Results

- **Total Test Cases:** 14
- **Passed:** 13
- **Partial Pass:** 1
- **Failed:** 1

| Test completion percentage | Test failure percentage | Test Execution Percentage | Total success rate |
|----------------------------|-------------------------|---------------------------|--------------------|
| 92.86% | 7.14% | 100% | 96.43% |

Table 18: Sign In Metrics Data

3.7.3.2 Reliability and Stability

- **App Crashes During Testing:** 0
- **Firestore Read/Write Success Rate:** 93%
- **Network Dependency Failures:** 1 (Firestore Update Failure)
- **Permissions Handling Success Rate:** 0% (Firestore Update Failure Test Failed)

3.7.3.3 Improvement Goals

- Reduce Firestore failure handling time from unknown to less than 2s with proper UI notifications.
- Achieve 100% confirmation messages for all update actions.
- Improve Firestore update success rate with better retry mechanisms.

3.8 Settings

This section will cover tests for the following features: Dark Mode, High Contrast Mode, ADA Accessibility, and Notifications. These functionalities ensure a customizable user experience, supporting different accessibility needs and preferences.

3.8.1 Tests and Results

3.8.1.1 Appearance Settings

3.8.1.1.1 Dark Mode Toggle

Expected outcome: Enabling Dark Mode updates the application theme and persists across app restarts.

Test outcome: Dark Mode successfully updates the theme and remains active after restarting the app.

Pass/fail status: Partial

Notes: Need to clean up some UI components to ensure all colors are being pulled correctly.

3.8.1.1.2 High Contrast Mode Toggle

Expected outcome: Enabling High Contrast Mode applies high-contrast colors.

Test outcome: High Contrast Mode correctly applies.

Pass/fail status: Partial

Notes: Currently its working on Settings page only, need to work on applying across app .

3.8.1.2 Accessibility Settings

3.8.1.2.1 ADA Accessibility Mode

Expected outcome: Enabling ADA mode modifies UI elements for usability.

Test outcome: Doesn't work

Pass/fail status: Fail

Notes: Need to work on updating UI Elements .

3.8.1.3 Notification Settings

3.8.1.3.1 Enable Notifications Toggle

Expected outcome: Notifications enable alerts for scheduled events.

Test outcome: Notifications crash when setting new time.

Pass/fail status: Fail

Notes: None.

3.9 Future Fixes

- Improve UI transitions when switching themes to prevent flickering.
- Ensure immediate visual feedback when toggling settings without needing a restart.
- Optimize storage of settings in Firebase for efficiency.

3.10 Metrics

| Test Completion Percentage | Test Failure Percentage | Test Execution Percentage | Total Success Rate |
|----------------------------|-------------------------|---------------------------|--------------------|
| 100% | 50% | 100% | 50% |

Table 19: Settings Test Metrics

4 Conclusion

The testing phase for the CatMap campus navigation app is complete. Testing of the app was a success, with a total of 100 planned tests we were able to execute all 100, with 88 successful/partial tests and 12 failed tests. This puts the overall test success rate at 88.0%. This result is somewhat below expected levels and puts the app at a higher risk of throwing errors, resulting in overall unreliability, but some components are still in development at time of testing so it wasn't unexpected. Based on the current state of the software and the remaining time before the deadline, all major components will be implemented and function in time for the final demonstration and presentation.