

## Test Plan and Report

**Product Name:** Hike Review

**Team name:** Hike Review

**Release date:** 03/11/25

- A. **User Story 1.1:** As a hiker, I want to see a map of my area so that I know where I'm at.
- B. **User Story 1.2:** As a hiker, I want to know of nearby hikes so that I can find one that I would like to explore.
- C. **User Story 2.1:** As a hiker, I want to see a path of my hike (the start, end and route of the hike) so that I can preview a hike I am interested in.
- D. **User Story 2.2:** As a hiker, I want to see a description of a hike so that I can learn more about the hike before attempting it.
- E. **User Story 3.1:** As a hiker, I want to have an account to be able to provide a difficulty or scenery rating to share my opinion on a hike.
- F. **User Story 3.2:** As a hiker, I want to know the total distance of the hike to gauge how intense it will be.
- G. **User Story 3.3:** As a hiker, I want to know the estimated time it takes to finish the hike so I can plan ahead.
- H. **User Story 4.1:** As a hiker, I want to get directions to a hike so I can find my way to the start of the path.
- I. **User Story 4.2:** As a hiker, I want to coordinate hikes with other users so I can make friends.
- J. **User Story 4.3:** As a hiker, I want to be able to favorite hikes in case I want to return to the specific hike.

### **Scenario 1:** Map view (**Pass/Fail**)

- 1. Start Hike Review app
- 2. User should see the map on open

### **Scenario 2:** Nearby hikes (**Pass/Fail**)

- 1. Start Hike Review app
- 2. User should see nearby hikes on the map
- 3. User should see nearby hikes in bottom sheet list

### **Scenario 3:** Hike path (**Pass/Fail**)

- 1. Start Hike Review app; select hike from map or list
- 2. User view should focus on the hike; the route of the hike should be displayed

### **Scenario 4:** Hike description (**Pass/Fail**)

- 1. Start Hike Review app; select hike from map or list
- 2. User view should focus on the hike; description for hike should be displayed

### **Scenario 5:** User Account (**Pass/Fail**)

- 1. Start Hike Review app; select profile; select sign up; type  
username=<hiker13>  
email=<hiker132025@gmail.com>  
password=<Hhtrail112>  
Press sign up button
- 2. User should see verification message and profile logged in
- 3. Start Hike Review app; select profile; type  
email=<hiker132025@gmail.com>  
password=<Hhtrail112>  
Press sign in button

4. User should see verification message and profile

**Scenario 6: Total distance (Pass/Fail)**

1. Start Hike Review app; select a hike
2. User should be able to see the total distance of the hike

**Scenario 7: Time estimate (Pass/Fail)**

1. Start Hike Review app; select a hike
2. User should be able to see the time estimate of the hike

**Scenario 7: User groups (Pass/Fail)**

1. Start Hike Review app; select a hike; select groups
2. User should see groups available for that hike
3. User should be able to join groups
4. User should be able to create groups

**Scenario 7: Favorite hikes (Pass/Fail)**

1. Start Hike Review app; select a hike
2. User should be able to favorite and unfavorite hike
3. After logout, the user should see favorite hikes in their profile

**Scenario 8: Reviews (Pass/Fail)**

1. Start Hike Review app; select a hike; select reviews
2. User should see reviews left by others
3. User should be able to leave a review with verification message for successfully submitted reviews.

**Functional Tests:**

Frontend (Expo/React Native (TS)):

Manual testing with each feature maximizing branch coverage

Utilizing console.log to verify indirect input/outputs during software execution

Backend (Flask (Python) / MySQL):

Utilized Postman for manual testing of each server endpoint

Used unit tests for some endpoints

**Unit Tests:**

Unit tests for flask server located in the tests folder of MySQL repo.