Test Plan and Report

Bathroom Locator - 11/29/25

System Test Scenarios

User Story 1: As a user who travels a lot, I want to be able to find public bathrooms easily on a map

→ Scenario 1a: View nearby bathrooms with location access turned on

1. Start Bathroom Locator website
    a. If prompted, give the browser access to current location
2. A blue marker should indicate the user's current location, which the map should automatically be zoomed in on
3. Brown pins, representing bathroom locations, should pop up on the map
4. Drag around the map to look at other areas
5. Click on the white recenter button in the bottom right corner to move the map back to the user's current location

→ Scenario 1b: View nearby bathrooms with location access turned off

1. Make sure location access is turned off
2. The map should automatically zoom in on the Santa Cruz area
3. The recenter button will not be rendered

We currently don't have automated tests for the frontend of this, as we were having issues rendering the map while testing. However automated tests do exist for the backend HTTP GET request for bathrooms that is made to our server pass.

User Story 2: As someone unfamiliar with the area, it would be helpful to be given walking directions to the bathroom that I've selected, so that I can find the bathroom quickly

→ Scenario 2: View walking directions of a selected bathroom

1. Start Bathroom Locator website
2. Click on the bathroom you want to navigate to
3. Click on the 'Navigate' button
4. If the user is on mobile, it should open the Google Maps App if downloaded, otherwise it should open a new tab in their default browser with Google Maps; if the user is on desktop, it should open a new tab with Google Maps. In both cases, it should show walking directions to the selected bathroom.

No automated tests exist for this as react testing library renders the components in a virtual dom, but we verified that this user flow works manually.

User Story 3: As someone who's non-binary it would be helpful to see bathrooms that have gender neutral stalls so that I can find a bathroom that I can use

→ Scenario 3a: Creating new bathrooms - only one gender selected

1. Start the Bathroom Locator website
2. Login as a user if not already logged in (see login test scenario)
3. Click on the "Add Bathroom" button
4. Select a location for the new bathroom and fill out the name and description (see create new bathroom flow)
5. The user should see Male, Female, and Gender Neutral as possible Gender options to select.
6. Click on the <Male> chip (initial aria label of "Select Male") under Gender in Additional Details
7. The chip should change colors to be green and have an aria label of "Unselect Male"
8. Save the new bathroom
9. Click on the pin on the map for the new bathroom
10. The bathroom details should have an additional properties section with a "Male" chip next to Gender to indicate male bathroom stalls exist in this bathroom.

→ Scenario 3b: Creating new bathrooms - multiple gender options selected

1. Start the Bathroom Locator website
2. Login as a user if not already logged in (see login test scenario)
3. Click on the "Add Bathroom" button
4. Select a location for the new bathroom and fill out the name and description (see create new bathroom flow)
5. The user should see Male, Female, and Gender Neutral as possible Gender options to select.
6. Click on the <Male> and <Gender Neutral> chips under Gender in Additional Details
7. Both chips should change their background colors to be green & update their respective aria label to be "Unselect (chip value)"
8. Save the new bathroom
9. Click on the pin on the map for the new bathroom
10. The bathroom details should have an additional properties section with "Male" and "Gender Neutral" chips next to Gender to indicate male and gender neutral bathroom stalls exist in this bathroom.

→ Scenario 3c: Viewing existing bathrooms

1. Click on an existing bathroom pin <E2 Bathroom Second Floor Bathroom>
2. Bathroom details should open

3. Female and Gender Neutral chips should be present under additional details to indicate both Gender Neutral and Female bathroom stalls exist for this bathroom.

All automated frontend and backend unit tests pass.

User Story 5: As a user of this app, I want to be able to tell if a bathroom actually exists before navigating to it

→ Scenario 5a: Liking a bathroom

1. Start the bathroom locator app, and click on the <Namaste Lounge Bathroom> pin
1. A details page should slide up from the bottom, and next to the bathroom title, the user should see the number of likes a bathroom has
2. Click the like button
3. The user should see that the number of likes has increased by 1, and the heart icon is filled red.

→ Scenario 5b: Ensuring that a liked bathroom stays liked

1. Start the bathroom locator app, and click on the <Namaste Lounge Bathroom> pin
2. Click the like button
3. The user should see that the number of likes has increased by 1, and the heart icon is filled red.
4. Close the bathroom details page
5. Click the <Namaste Lounge Bathroom> pin on the map again.
6. Bathroom details should pop up, and the bathroom should still be liked

→ Scenario 5c: Unliking a bathroom

Assumption for this scenario the user has previously liked the <E2 Bathroom Second Floor Bathroom>

1. Start the bathroom locator app, and click on the <E2 Bathroom Second Floor Bathroom> pin
2. The bathroom details should open, and the user should see that they've already liked this bathroom
3. Click the like button again
4. The user should see that the number of likes should decrease by one and the heart icon should go back to being outlined

→ Scenario 5d: Ensuring that a unliked bathroom stays unliked

1. Start the bathroom locator app, and click on the <E2 Bathroom Second Floor Bathroom> pin
2. The bathroom details should open, and the user should see that they've already liked this bathroom
3. Click the like button

4. The user should see that the number of likes should decrease by one and the heart icon should go back to being outlined
5. Close the bathroom details page
6. Click the <E2 Bathroom Second Floor Bathroom> pin on the map again.
7. Bathroom details should pop up, and the bathroom should still be unliked

→ Scenario 5e: Bathroom becomes verified

1. Start the bathroom locator app, and click on the <E2 Bathroom Second Floor Bathroom> pin
2. The user should see this bathroom currently has 4 likes (with no changes made to the data), and no “Verified Bathroom” chip is visible
3. Click the like button, and see the number of like go up to 5
4. The bathroom is now considered a verified bathroom and the user should be able to see a “Verified Bathroom” under the name of the bathroom in the bathroom details page.

→ Scenario 5f: Opening up a verified

1. Start the bathroom locator app, and click on the <Namaste Lounge Bathroom> pin
2. The bathroom details page should open, and the user should see that the bathroom has 10 likes, and the “Verified Bathroom” chip exists (if no changes were made to the data) under the bathroom name

All automated frontend and backend unit tests pass.

User Story 6: As a user, I want to be able to create an account that I can use later.

→ Scenario 6a: Successfully creating a new account

1. Start the Bathroom Locator app and click the Login button
2. The user should be navigated to the Login screen
3. Click the Sign up Button
4. The user should be navigated to the Sign up screen and see that the Sign up button is currently disabled
5. In the signup form type:
   Email: <sample@email.com> (note this can be anything that looks like a correctly formatted email address as we don’t have email verification in our application)
   Password: <SamplePass123>
   Confirm password: <SamplePass123>
6. The user should see that the Sign up button is now enabled
7. Click the Sign up button, and the page should navigate back to the home page
8. The user should see a placeholder avatar icon in the place of the login button on the home page to show they are logged in

→ Scenario 6b: Unable to sign up due to the entered Password and Confirm passwords not matching

1. Start the Bathroom Locator app and click the Login button
2. The user should be navigated to the Login screen
3. Click the Sign up Button
4. The user should be navigated to the Sign up button and see that the Sign up button is currently disabled
5. In the signup form type:
    a. Email: <sample@email.com> (note this can be anything that looks like a correctly formatted email address as we don't have email verification in our application)
    b. Password: <SamplePass123>
    c. Confirm password: <Sample123>
6. Click the now enabled Sign up button
7. The user should see an alert popup saying "'Passwords do not match"
8. The user can then then attempt to sign up again with matching passwords

→ Scenario 6c: Unable to sign up due to invalid email

1. Start the Bathroom Locator app and click the Login button
2. The user should be navigated to the Login screen
3. Click the Sign up Button
4. The user should be navigated to the Sign up button and see that the Sign up button is currently disabled
5. In the signup form type:
    a. Email: <not an email>
    b. Password: <SamplePass123>
    c. Confirm password: <SamplePass123>
6. Click the now enabled Sign up button
7. The user should see an alert popup saying "Invalid email format"
8. The user can then then attempt to sign up again with a well formatted email address

All automated frontend and backend unit tests pass.

User Story 8: As a user, I want to see bathroom amenities so I know what to expect.

→ Scenario 8a: Creating new bathrooms - only one amenity selected

1. Start the Bathroom Locator website
2. Login as a user if not already logged in (see login test scenario)
3. Click on the "Add Bathroom" button
4. Select a location for the new bathroom and fill out the name and description (see create new bathroom flow)

5. The user should see Toilet Paper, Soap, Paper Towel, Hand Dryer, Menstrual Products, and Mirror as possible Amenity options to select.
6. Click on the <Toilet Paper> chip (initial aria label of “Select Toilet Paper”) under Amenities in Additional Details section
7. The chip should change colors to be green and have an aria label of “Unselect Toilet Paper”
8. Save the new bathroom
9. Click on the pin on the map for the new bathroom
10. The bathroom details should have an additional properties section with a “Toilet Paper” chip next to Amenities to indicate this bathroom has toilet paper available.

→ Scenario 8b: Creating new bathrooms - multiple amenity options selected

1. Start the Bathroom Locator website
2. Login as a user if not already logged in (see login test scenario)
3. Click on the “Add Bathroom” button
4. Select a location for the new bathroom and fill out the name and description (see create new bathroom flow)
5. The user should see Toilet Paper, Soap, Paper Towel, Hand Dryer, Menstrual Products, and Mirror as possible Amenity options to select.
6. Click on the <Soap> and <Menstrual Products> chips under Amenities in Additional Details
7. Both chips should change their background colors to be green & update their respective aria label to be “Unselect (chip value)”
8. Save the new bathroom
9. Click on the pin on the map for the new bathroom
10. The bathroom details should have an additional properties section with “Soap” and “Menstrual Products” chips next to Amenities to indicate this bathroom has soap and menstrual products available for use.

→ Scenario 8c: Viewing existing bathrooms

1. Click on an existing bathroom pin <E2 Bathroom Second Floor Bathroom>
2. Bathroom details should open
3. The amenities section show display the following chips indicating they are available at the bathroom: Toilet Paper, Soap, Paper Towel, Menstrual Products, and Mirror

All automated frontend and backend unit tests pass. We also have manual testing we did to check that the bathroom details, Scenario 8c, works as expected due to issues rendering the map and the bathroom markers.

User Story 9: As a community member, I want to be able to add a new bathroom to the map to help others using this website

→ Scenario 9a: Successfully add a new bathroom

1. Load the app and log into an account
2. The “Add bathroom” button should appear in the bottom right screen. Click on it
3. A banner at the top, saying “Choose a location for the bathroom” should appear. Click where you’d like to place the new bathroom
    a. Note: as of this release the website does not automatically select your current location as the default location for the new bathroom
4. A pin should appear where you clicked, and a form titled “New Bathroom” should pull up from the bottom
5. Type the following in the respective inputs
    a. Bathroom Name: <Test bathroom> (An actual bathroom name can also be used if creating a real bathroom)
    b. Bathroom Description: <Enter via the main doors closest to the elevator and take a left into the hallway. The bathrooms are the first entrance on your right>
6. Select any additional details (eg gender, amenities) that fit the bathroom
7. Hit the “Save” button
8. The New Bathroom form should close, and the new bathroom pin should be visible on the map. When you click on the bathroom pin, it should display the information you just inputted into the New Bathroom form

This scenario was tested automatically with our backend and frontend unit tests and both sets passed. We also manually verified that the new bathroom pin is displayed on the map after successful creation of a bathroom.

→ Scenario 9b: Start adding a new bathroom, then stop midway

1. Load the app, login, click on the “Add bathroom” button, and click anywhere on the map to leave a pin
2. The create bathroom form should pop up
3. Click the “Cancel” button, or anywhere outside the form
4. The form should slide down, with “New Bathroom” still peeking out.
5. Click the “Cancel” button in the banner below the search bar
6. The New Bathroom form should slide away completely, and the pin that you put down earlier should go away
7. The next time that you create a new bathroom, the New Bathroom form should be reset for an empty bathroom

This scenario was tested manually and via automated frontend unit tests, and both passed. We manually verified that the New Bathroom form is reset if the user cancels creation of a bathroom midway, and later decides to create a new bathroom.

→ Scenario 9c: Move around the location of the new bathroom while creating it

1. Load the app, login, click on the “Add bathroom” button, and click anywhere on the map to leave a pin
2. Fill out the form accordingly
3. Click outside the add bathroom form or slide the form down
4. The map should now be visible, and click where you’d like to move the pin to
5. On the map, click where you’d like to move the pin to
6. The pin should move there, and the New Bathroom form should open up again, retaining all the information you inputted earlier

This scenario was tested manually following this user flow as we are currently unable to render the Map in our automated tests.

→ Scenario 9d: Try to create a bathroom, but leave out mandatory information like the bathroom title and description

1. Load the app, login, click on the “Add bathroom” button, and click anywhere on the map to leave a pin
2. Click on the “Save” button, without inputting the bathroom name or description
3. Nothing should happen.

All automated frontend and backend unit tests pass.

User Story 12: As a user, I want to be able to search for bathrooms on the map in specific cities even without allowing it to access my location.

→ Scenario 12: Search cities on a map, eg if location access is turned off, or for looking for bathrooms in a different area

1. Load the Bathroom Locator website and click on the search bar
2. Type <Los Angeles> in the search bar
    a. Any city or area can be used
3. A dropdown of five best matching suggestions should appear as the user types
4. Click on the first <Los Angeles> suggestion or hit enter on the keyboard to select the first option by default
5. The user should see the map move to Los Angeles and load any bathrooms in the area

Our automated frontend tests for the search bar pass, and we manually tested the function of the map panning to the selected location.

User Story 13: As an existing user, I want to be able to log into an account that I’ve already created

The following user exists in our database for test purposes:
Email: test@user.com
Password: password

→ Scenario 13a: Successful login

1. Load the Bathroom Locator website and click the Login button

2. The user should be navigated to the Login page
3. Type:
    a. Email: test@user.com
    b. Password: password
4. Click “Login”
5. It should navigate back to the map, and the Login button in the top right corner should be replaced with a user icon. Also, the “Add Bathroom” button in the bottom right should be visible

→ Scenario 13b: Missing credentials when logging in

1. Load the app and hit the “Login” button
2. The “Login” button on the login page should be disabled unless both the email and password field are filled in

→ Scenario 13c: Invalid email login credentials

1. Load the app and hit the “Login” button
2. Type:
    a. Email: nottest@user.com
    b. Password: password
3. Click “Login”
4. A red banner should appear, stating “Invalid login credentials”

→ Scenario 13c: Invalid password login credentials

1. Load the app and hit the “Login” button
2. Type:
    a. Email: test@user.com
    b. Password: notpassword
3. Click “Login”
4. A red banner should appear, stating “Invalid login credentials”

All automated frontend tests for this pass. We don’t have any backend tests as the authentication is handled by the Supabase auth library

Unit Tests

Frontend: bathroom-locator/frontend/src/__tests__/*
Backend: bathroom-locator/backend/src/test/*

All unit tests run successfully.

- Our frontend tests run automatically when a PR is made or a new commit is made to a branch with an open PR, and block the PR from being merged until all tests pass.
- We were unable to set up a CI/CD pipeline in github for our backend tests due to some issues initializing our postgres docker container. However, as a

workaround we have a precommit check to ensure that all backend tests must pass in order for a change to be committed.