

System and Unit Test Report

CMPS 115

Homie

Ben Pao, Mary Chern, Keith Ho, Andy Wong

Dec 2, 2018

System Test Scenario Format

[User Story]

1. ~~[Finished Task]~~
2. ~~[Finished Task]~~
3. [Unfinished Task]

[SCENARIO]:

1. *[Set up the context]*
2. *[User actions]*
3. *[Output (from the user's perspective)]*

Sprint 1

As a developer, I want to learn react native, so that I have a 6/10 proficiency

- ~~1. Complete a tutorial (2hrs)~~
- ~~2. Decide which database (1.5 hrs)~~
- ~~3. Set up development environment (1.5hrs)~~

No system test scenarios for this user story, since there is no way to input values to decide if the story has been completed.

As a end-user, I want to see a log in or sign up button, to create an account

- ~~4. Creating a Database to store info (1 hr)~~
- ~~5. Create Login page (0.5 hr)~~
- ~~6. pass information to the database (1 hr)~~

SCENARIO 1:

4. Start expo application; select 'Sign up'; type
 - a. First Name = <Bob>
 - b. Last Name = <by>
 - c. Email = <Bobby@expo.com>
 - d. Password = <Password>
 - e. Confirm password = <Password>
5. Press 'Sign up' Button
6. The account information should be present in the database

SCENARIO 2:

1. Start expo application; enter the information below for log in
 - a. Email: <Bob>
 - b. Password: <Password>
 2. Press 'Login' Button
 3. User should see a "Welcome to Homie" Page
-

As a developer, I want to be able to test my login page for functionality

- ~~1. Write unit tests for login page~~

No system test scenarios for this user story, since there is no way to input values to decide if the story has been completed.

Sprint 2

Design UI (Every logical paths possible--where does clicking this button take you?).

- ~~1. Design UI Logic path for login process(3 hrs)~~

No system test scenarios for this user story, since there is no way to input values to decide if the story has been completed.

As a developer, I want to be able to set up a database to store information.

- ~~1. Set up the database(3 hrs)~~

No system test scenarios for this user story, since there is no way to input values to decide if the story has been completed.

As a end-user, I want to be able to input my account information.

- ~~1. Sign up redirect to log in page/automatically sign in --> go to homepage (1 hr)~~
- ~~2. Get rid of Back option after user logs in/signs up (2 hrs)~~
- ~~3. Sign up testing (6 hrs)~~
- ~~4. Sign up can't throw expo errors when email is already used (1 hr)~~
- ~~5. Log in can't throw expo errors when email is badly formatted (1 hr)~~
- ~~6. Log in/Sign up can't throw any errors in general (3 hrs)~~
- ~~7. Add Back/cancel button to sign up page(3 hrs)~~

SCENARIO 1:

1. Start expo application; select 'Sign up'; type
 - a. First Name = <Bob>
 - b. Last Name = <by>
 - c. Email = <Bobby@expo.com>
 - d. Password = <Password>
 - e. Confirm password = <Password>
2. Press 'Sign up' Button
3. User should be redirected to a "Welcome to Homie" Page

SCENARIO 2:

1. Start expo application; select 'Sign up';
2. Press 'Cancel' Button
3. User should be redirected to the Login page

SCENARIO 3:

1. Start expo application; select 'Sign up'; type
 - a. First Name = <Bob>
 - b. Last Name = <by>
 - c. Email = { an email that is already in use by another account }
 - d. Password = <Password>
 - e. Confirm password = <Password>
2. Press 'Sign up' Button
3. User should see an alert that says "The email is already in use by another account."

SCENARIO 4:

1. Start expo application; select 'Sign up'; type
 - a. First Name = <Bob>
 - b. Last Name = <by>
 - c. Email = <Bobby@exp>
 - d. Password = <Password>
 - e. Confirm password = <Password>
2. Press 'Sign up' Button
3. User should see an alert that says "The email address is badly formatted."

SCENARIO 5:

1. Start expo application; select 'Sign up'; type
 - a. First Name = <Bob>
 - b. Last Name = <by>
 - c. Email = <Bobby@expo.com>
 - d. Password = <Password>
 - e. Confirm password = { something != "Password" }
2. Press 'Sign up' Button
3. User should see an alert that says "Passwords don't match"

SCENARIO 6:

1. Start expo application; select 'Sign up'; type
 - a. Leave first name and/or last name field empty
 - b. Email = <Bobby@expo.com>
 - c. Password = <Password>
 - d. Confirm password = <Password>
2. Press 'Sign up' Button
3. User should see an alert that says "Please enter your first name"

SCENARIO 7:

1. Start expo application; enter the below information for log in
 - a. Email: <Bobby@expo>
 - b. Password: <Password>
2. Press 'Log In' Button
3. User should see an alert that says "Error: The email address is badly formatted"

SCENARIO 8:

1. Start expo application; enter the below information for log in
 - a. Email: <Bobby@expo.com>
 - b. Password: { wrong password or null password }
 2. Press 'Log In' Button
 3. User should see an alert that says "Error: The password is invalid or the user does not have a password."
-

As a end-user, I want to be able to create a House.

- ~~1. Make a welcome page that has options: Create a House (2 hrs)~~
- ~~2. Be able to create a house, store house information in the database, and generate a house ID (5 hrs)~~

SCENARIO 1:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. User should see a Welcome page that has a 'Create a House' button

SCENARIO 2:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press 'Create a House' button; type
 - a. Enter a name for Your House: <Bobby'sHouse>
4. User should see the application Home page

Sprint 3

As a end-user, I want to be able to add people to my house (originated from sprint 2)

- ~~1. Be able to add others to a house by email (in app) without verification from the recipient (5 hrs)~~
- ~~2. Be able to join a house by house ID (4 hrs)~~
- ~~3. Be able to join a house by the email address of a user in the house (5 hrs)~~

SCENARIO 1:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the far right (settings) button
4. Press 'Add People'
5. Type in <blah@gmail.com>
6. User should see an alert that says "User with email <blah@gmail.com> already has a house with us!"

SCENARIO 2:

1. Create a new user account; type:
 - a. First Name = <John>
 - b. Last Name = <ny>
 - c. Email = <Johnny@expo.com>
 - d. Password = <Password>
 - e. Confirm password = <Password>
 2. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
 3. Press 'Login' button
 4. Press the far right (settings) button
 5. Press 'Add People'
 6. Type in <Johnny@expo.com>
 7. Press on 'Settings' button
 8. Press on 'Log out' button
 9. Log into other account; type:
 - a. Email: <Johnny@expo.com>
 - b. Password: Password
 10. You should now see the same data presented in Bob's House
-

As a end-user, I want to be able to see shared content within my house. (originated from sprint 2)

- ~~1. Create tables for chores, bills, and grocery list in firebase (4 hrs)~~
- ~~2. Be able to display only content associated with the user's house (7 hrs)~~
 - ~~a. Grocery/supply list: actual data~~

SCENARIO 1:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
 2. Press 'Login' button
 3. Press the shopping cart icon to navigate to grocery screen
 4. User should see a list of grocery items that the user's housemates' have added
-

As a end-user, I want to be able to navigate the app. (originated in sprint 1)

- ~~1. Redirect user to App if user has a house, redirect to Welcome otherwise (3 hrs)~~
- ~~2. Button to navigate between screens (home, grocery, etc.) (4 hrs)~~
- ~~3. Create a header (3 hrs)~~
- ~~4. Create screen templates (pages) (3 hrs)~~
- ~~5. Write unit test for each screen (5 hrs)~~
- ~~6. Add links to different pages on the bar and be able to navigate to a page (5 hrs)~~
- ~~7. Replace names with icons in the nav bar (1 hr)~~

SCENARIO 1

1. Start expo application (in order to log in)
 - a. If the user already has an account and is in a home, they will be redirected to the Home page
 - i. Enter the information below
 - ii. Username: <Bob>
 - iii. Password<Password>
 - b. If the user has an account but is not in a home, they will be directed to a Join house page
 - i. Enter the information below
 - ii. Username: <Bob>
 - iii. Password<Password>

As a end-user, I want to be able to add/remove items from the master grocery list

- ~~1. Create add/delete buttons (3 hrs)~~
- ~~2. Have them connected to the db (3 hrs)~~

SCENARIO 1:

7. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
8. Press 'Login' button
9. Press the 'Grocery' (Shopping cart) button
10. Press on 'Add Item' card; type:
 - a. 'Lettuce'
11. Press on the '+' Button
12. You should see the 'Lettuce' Card show up in the grocery page

SCENARIO 2:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Grocery' (Shopping cart) button)
4. You should see the 'Lettuce' Card show up in the grocery page
5. Press the 'Trash Can' (Delete) Icon
6. You should see the 'Lettuce' Card delete from the grocery page

AB testing for database (originated from sprint 3)

- ~~a. Research how to do it (4 hrs)~~

No system test scenarios for this user story, since there is no way to input values to decide if the story has been completed.

Sprint 4

Be able to display only content associated with the user's house

- ~~1. Grocery list: actual data~~
- ~~2. Chore list: actual data~~
- ~~3. Payments: actual data~~

SCENARIO 1: (assuming person "Johnny" is in your house)

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Grocery' (Shopping cart) button
4. Press on 'Add Item' card; type:
 - a. 'Lettuce'
5. Press on the '+' Button
6. Press 'Create Payment' Button; type:
 - a. <Groceries> in 'Enter a Payment' Section
 - b. <49.00> in 'Enter the Payment Quantity'
 - c. Rotate to <Johnny> in recipient
7. Hit 'Submit'
8. Press on 'Log out' button
9. Log into other account; type:
 - a. Email: <Johnny@expo.com>
 - b. Password: Password
10. You should see the 'Lettuce' Card show up in the grocery page and 'Grocery' payment card

As a end-user, I want to be able to add/remove items from the master grocery list.

- ~~1. Fix delete button to work with cards (3 hrs)~~
- ~~2. Implement a "clear list" button (3 hrs)~~
 - a. (Note to us): Only works for the person who hit the clear-list

SCENARIO 1:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Grocery' (Shopping cart) button
4. Press delete button
5. The item that is pressed should not appear anymore

- a. *This happens on every house dynamically*
 - 6. *Press clear list*
 - a. *User who pressed grocery list will be empty*
 - i. *Other housemates will still have uncleared list unless they refresh the page*
-

As a end-user, I want to see a master grocery list for the house. (originated from sprint 2)

- ~~1. Be able to add items to the grocery list (just input fields that get stored to the db) (3 hrs)~~
- ~~2. Be able to store added items by house ID and user ID (3 hrs)~~
- ~~3. Be able to display every added item associated to the user's house with name of the user who created it (5 hrs)~~
- ~~4. Fix the layout of username on the cards (3 hrs)~~

SCENARIO 1:

- 1. *Start expo application; type*
 - a. *Email <Bobby@expo.com>*
 - b. *Password: <Password>*
 - 2. *Press 'Login' button*
 - 3. *Press the 'Grocery' (Shopping cart) button*
 - 4. *Press text input example <Soap>*
 - 5. *Items stored in your house will display only to your house*
 - a. *Soap and all previous items will be visible*
 - b. *The name of the user who created soap in this example Bobby will be portrayed below*
-

As a end-user, I want to be able to create/see chores.

- ~~1. Create add/delete buttons on chores screen (3 hrs)~~
- ~~2. Implement cards for each chore (3 hrs)~~

SCENARIO 1:

- 1. *Start expo application; type*
 - a. *Email <Bobby@expo.com>*
 - b. *Password: <Password>*
- 2. *Press 'Login' button*
- 3. *Press the 'Chores' (List) button*
- 4. *User should be able to see all the chores in cards*

SCENARIO 2:

- 1. *Start expo application; type*
 - a. *Email <Bobby@expo.com>*
 - b. *Password: <Password>*

2. Press 'Login' button
 3. Press the 'Chores' (List) button
 4. User should be able to see add and delete buttons for chores
-

As a end-user, I want to be able to see my payments so that I know what I owe and what I am requesting.

- ~~1. Add Requests and Charges screens under payments screen (create a stack) (5 hrs)~~
2. Display each payment with the username of the housemate being charged and the housemate charging the current user (10 hrs)
- ~~3. Create buttons for create/cancel requests (4 hrs)~~

SCENARIO 1:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
 2. Press 'Login' button
 3. Press the 'Payments' (List) button
 4. User should be able to press Create Payment
 - a. Add payment Name
 - b. Add payment Amount
 - c. Pick who owes you
 5. Press Submit
 6. Automatic reroute to Request Payment Page
 7. On each request card there is a cancel button "paid"
-

Unit testing (originated from sprint 3)

- ~~1. For every screen created so far (10 hrs)~~
- ~~2. For the main functionalities implemented so far (8 hrs)~~
- ~~3. Fix broken unit tests (30 hrs)~~

No system test scenarios for this user story from the user's perspective, but it was verified by running "npm test"

SCENARIO 1:

1. Type in "npm test" in command line
 2. All 32 tests (snapshot + unit tests) should pass
-

Add CI tools (originated from sprint 3)

- ~~1. Research how to do it (4 hrs)~~
- ~~2. Implement it (4 hrs)~~

No system test scenarios for this user story, since this story has not been completed

As a end-user, I want to be able to add chores into a chore wheel (originated from sprint 1)

- ~~1. Create a Chores page (0.5 hr)~~
- ~~2. Create a chores chart (1 hr)~~
- ~~3. add/delete buttons on chores page (0.5 hr)~~
- ~~4. Be able to store information into database (1 hr)~~
- ~~5. Link chores to user in database and portray the user's chores (1 hr)~~
- ~~6. As a end-user, I want the chore wheel to be able to manually rotate on a weekly basis (3hr)~~

SCENARIO 1:

5. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
6. Press 'Login' button
7. Press the 'Chores' (List) button at the bottom
8. User should be able to see a chores list

SCENARIO 2:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Chores' (List) button
4. User should be able to press the add button to add a new chore
5. User should be able to see the updated chores list

SCENARIO 3:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Chores' (List) button
4. User should be able to press the delete button to delete a chore
5. User should be able to see the updated chores list

SCENARIO 4:

1. Start expo application; type

- a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Chores' (List) button
4. User should be able to press the rotate button to rotate the chore assignment to another housemate
5. User should be able to see the updated housemate name on the chore

SCENARIO 5:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Chores' (List) button
4. User should be able to press the rotate all button to rotate all of the chore assignments to other housemates
5. User should be able to see the updated housemate name on all the chores

SCENARIO 6:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Chores' (List) button
4. Press the 'Add a Chore' button
5. User should be able to press the back button on the header to return to the chore list

As a end-user, I want to be able to see who added which items on the grocery list (originated from sprint 3)

- ~~1. Pull user data from db and display them (3 hrs)~~
- ~~2. Display each grocery/supply item nicely (6 hrs)~~

SCENARIO 1:

1. Start expo application; type
 - a. Email <Bobby@expo.com>
 - b. Password: <Password>
2. Press 'Login' button
3. Press the 'Grocery' (Shopping cart) button
 - a. 'Lettuce'
4. Press on the '+' Button

5. *The name of the person who added the item should appear at the bottom of the card that was just created*
-

As a end-user, I want to be able to receive automatic notification in any changes to my chores/grocery list originated from sprint 3)

1. Set up a notification system (10 hrs)
2. Have a notification sent to everyone in the house when changes are made in the chores/grocery list (4 hrs)

No system test scenarios for this user story, since this story has not been completed

As a end-user who gets invited to join a house, I want to be able to get a notification in app and join. (originated from sprint 3)

1. Have a notification sent to the person who is invited (2 hrs)
2. Allows the recipient to join house with notification (without knowing a house id/sender's email) (8 hrs)

No system test scenarios for this user story, since this story has not been completed

Unit Tests (Jest)

AboutScreen:

- Snapshot testing

AddPeopleScreen:

- Snapshot testing
- emailNotNull()
 - Checks non-null email passes
 - Checks null email is detected and user is alerted
 - Checks empty email is detected and user is alerted

ChargesPaymentsScreen:

- Snapshot testing

ChoresScreen:

- Snapshot testing

AddChoresScreen: (Will do after Dec 2 because this screen is new)

- Snapshot testing
- getTodayDate()
 - Checks date is fetched

CreateHouseScreen:

- Snapshot testing

CreatePaymentsScreen:

- Snapshot testing

GroceryScreen:

- Snapshot testing

HomeScreen:

- Snapshot testing

JoinHouseScreen:

- Snapshot testing
- badInput()
 - Should pass if user tries to join house by just houseID
 - Should pass if user tries to join house by just housemate's email
 - Should detect if user tries to join house without neither houseID or housemate's email and alert user

- Should detect if user tries to join house with both houseID and housemate's email and alert user

LoginScreen:

- Snapshot testing

PaymentsScreen:

- Snapshot testing

ProfileScreen:

- Snapshot testing

RequestPaymentsScreen:

- Snapshot testing

SettingsScreen:

- Snapshot testing

SignupScreen:

- Snapshot testing
- nameIsNull()
 - Should pass if user provides both first and last name
 - Should detect first name is null and alert user
 - Should detect first name is empty and alert user
 - Should detect last name is null and alert user
 - Should detect last name is empty and alert user
- passwordDontMatch()
 - Should pass if password and confirm_password match
 - Should detect and alert user if password and confirm_password don't match

WelcomeScreen:

- Snapshot testing