

System and Unit Test Report

CMPS 115

Homie

Andy Wong, Alexis Mabugat, James Chen, Danny Phan, Jerod Zheng

March 12 2019

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 familiarize myself with the technology, so that I can build the app.**

- ~~1. Complete a tutorial (6hrs)~~
- ~~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 I want to be able to have a account, so that I can keep my information safe and keep track of my bills.**

- ~~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 application; select 'Sign up'; type
 - a. First Name = <Bob>
 - b. Last Name = <by>
 - c. Email = <Bobby@google.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 "activities page" Page
-

- **As a user I want to be able to be able to be able to split up the bill, so that I can charge people.**

SCENARIO 1:

7. *Input total price*
8. *Input split amount*
9. *Input tax/tip*
10. *Press calculate*
11. *Total after tax/ tip calculates*
12. *See total split between people*

Sprint 2

- **As a user I want to be able to navigate the app, so that I can use all the features.**

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

- **As a user I want to have a profile, so that I can let people find me.**

- ~~1. Set up the database (3 hrs)~~
 - ~~2. Fetch from database~~
-

1. **[3]As a user I want to see a history of all my payments/charges, so that I can verify if I paid someone yet**
 - Store the payments in DB
 - Store all charges too
 - Keep tab of which one was paid/not paid

SCENARIO 1:

1. *Go to home screen see charges*
2. *Press on requests screen*

3. *Data should display*

Sprint 3

- **As a user I want to be able to make in app payments (Venmo), so that I don't have to waste time and just keep staying on one app to do all my business.**

- ~~1. Be able to open paypal~~
- ~~2. Pass information to paypal~~
- ~~3. Get confirmation~~

SCENARIO 1:

1. *On charges press pay with paypal*
2. *Login to paypal*
3. *Pay amount charged*
4. *See charged post go return to history page*

-
- **[3]As a user I want to be able to charge someone, so that I can get my money back.**

- ~~1. Get values from calculation screen~~
- ~~2. Find user id of charging person~~
- ~~3. Charge person with those values~~

SCENARIO 1:

1. *Do calculation from uneven or even split*
2. *Press charge*
3. *Information should be passed over*
4. *Type email and find*
5. *List with user should appear*
6. *Press charge*
7. *Request should now show in your request page*

-
- **As a user I want to be able to split bills unevenly, because not everyone orders the same thing**

- ~~1. Choose uneven split~~
- ~~2. Input item with price how ever many you want~~
- ~~3. It calculates total for that one person~~

SCENARIO 1

- 1. Choose unevenly split*
- 2. Input item with price*
- 3. Calculate*
- 4. Press charge and go to charge screen with the total*
- 5. Can remove items and add if wanted to*

Sprint 4

- **As a user I want to be able to use the app smoothly so that I can access what I want when I want.**

- ~~1. Make sure nav works~~
- ~~2. Charges is workign~~
- ~~3. Displaying request and charges is correct~~
- ~~4. Logic is also correct~~

-
- **As a user I want to be able to make in app payments (Venmo), so that I don't have to waste time and just keep staying on one app to do all my business.**

- ~~1. Fix part where we link a charges request to paypal~~

- ~~2. Pass vars to paypal~~
- ~~3. Return and charges database tables~~
- ~~4. Moves payments to history table~~

SCENARIO 1:

1. *On charges screen*
 2. *Press pay*
 3. *Enter paypal info*
 4. *Pay*
 5. *When return charge should dissapear and show up in history page*
-

Unit Tests (Xcode)

LoginScreen:

- Xcode testing