

Thursday, October 26, 2023

## Project 2 CS 250

Hong Nhung Nguyen

### 1. Unit Test & Functional Test

- Def: Unit test maybe an individual function, method, procedure, module or object in the project
- Unit Test example: UserAccount Class. This class detailed basic users's account setting.
  - Tested Components: Username; Password
  - Tested Methods: setUsername(); setPassword(); addPlanner(); removePlanner(); addRecipe(); removeRecipe(); setBudget();getBudget()

<u>Unit Testing for Username &amp; Password</u>			
Test Case	Type	Description	Objectives
One piece	String	Accepted Input	Verify that an user could enter their user name and password
“ “	null	Error: Invalid Input. Username and password cannot be null	Verify that an user cannot leave their username and password blank
12578	Int	Accepted Input	Verify that an user could enter either string or int datatype
2200483647	Int	Error: out of limit of int data type	Verify what happen if input received out of upper bound

Functional Test Plan for methods in UserAccount Class			
setUserName()	setPassword()	addRecipe()	removeRecipe()
Enter valid user name: "Jenny123"	Enter valid password: "Pass123"	Enter content of recipe	Choosing an existing recipe
Expected output: Pass. when user enter valid input, setUserName() should add new user name to the database	Expected output: Pass. when user enter valid password , setPassword() should save password associated with the user account to the database	Expected output: Pass. when user choose add recipe option; addRecipe() should allow new recipe to be added to the database	Expected output: Pass. when user choose to remove an existing recipe in their account, removeRecipe() will enable a recipe to be removed from the database.

## 2. System Testing:

Testing the integrated system as a whole

- Login Page:

- Test case 1: verify user enter valid user name
- Test case 2: if user enter valid user name -> ask to enter password -> verify user enter valid password

- Recipe page:

- Test case 1: once user login successfully. Verify that user can add recipe in recipe page.
- Test case 2: once user login successfully. Verify that user can search for recipe under recipe page.
- Test case 3: once user login successfully. Verify that user can modify their recipe.

- Data manangement:

- Test case 1: verify that database is updated frequent once any changes happens
- Test case 2: verify that user can see their history on their account.

- App UI:

- Test case 1: verify that all app element are displayed correctly and user friendly. For example, when opening app user should see login page pop up first with sign-in button and “create account” button next to each other.
- Test case 2: verify that each app options is correctly interacted to the other. For example: after login successfully, user will able to see their homepage layout different app options.
- Performance testing:
  - Test case 1: measure app performance running on OS/Android mobile platform
  - Test case 2: measure website performance such as loading page speech when running on desktop.