

Tu Lam

CS 362 / Instructor Vijay Tadimeti

January 28th, 2021

Homework #3

1. *Software Testing*

i) *Which of the 3 systems would you apply unit testing to and why?*

Answer: Out of the 3 systems, **S/W system A** would apply to the unit testing. System A fit with this testing due to the unit testing focus on testing individual component in the whole system. As demonstrated in the drawing, the dotted line showing only that component is in focus and that the team is only focus on testing that unit.

ii) *Which of the 3 systems would you apply integration testing to and why?*

Answer: Out of the 3 systems, **S/W system C** would apply to the integration testing. System C fit in with this testing is due to the fact that the test is looking at each individual component that can be brought together and can perform the test without any problem. Basically, asking if these components can integrate and work with each other. System B shows that as the dotted line focus on bringing 3 individual components to test and see if they would integrate well with the whole system.

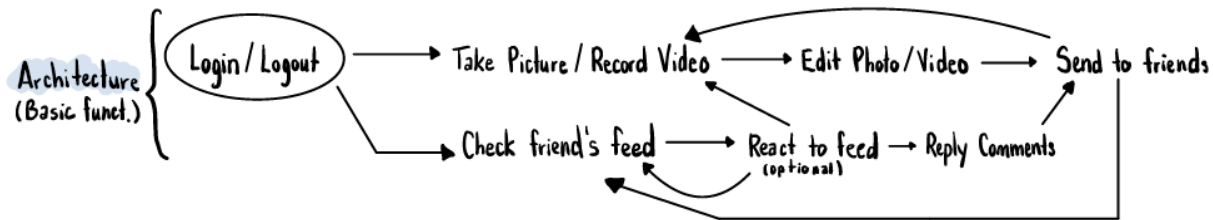
iii) *Which of the 3 systems would you apply system testing to and why?*

Answer: Out of the 3 systems, **S/W system B** would apply to the system testing. System B would be the best fit as the system testing is performing the test as a whole system, meaning testing everything in the system itself. System B shows this as the dotted line circle the entire system and every block of components are part of that testing phase.

2. In-Class Activity Revisit

i) Describe or draw your system

Answer: Below would be a drawing of the system from my in-class activity and this is revolved around the Snapchat SNS.



ii)

Verification	Validation
<ul style="list-style-type: none">- Display the login page if user access through app, if the data entry matches, let the user in.- When sending a photo to a friend, the user should see a red arrow icon that it sends successfully.	<ul style="list-style-type: none">- Test the boundary to see if the message pop up about the user sends more than 10 photo/videos to a friend at a time or would it continue.- Check to see the error handling when sending thing without internet/LTE should display to resend again.

iii)

Functional Testing	Non-Functional Testing
<ul style="list-style-type: none">- When a user decides to click "save" it will save to their device photo album.- User will also get notification if someone respond back to their text, or send a photo/videos to them directly.	<ul style="list-style-type: none">- Testing the speed performance of the app when use, should launch app and have everything load in about 5 seconds.- Testing if the texting in app can hold/support up to infinite word count.

3. Leap Year Program

i, ii)

Answer: All the part for this question is being provided with an URL link to the GitHub where the code for leap year with error handling, without error handling, and the flowchart to the README.md would be located at.

URL: <https://github.com/tulam1/CS362---Hw3>

4. Agile and Waterfall methodologies

i) Describe a project in a few words.

Answer: The project is a video game based on text adventure video game. The game will start and give user options to choose which path they want to go and the choice choose by user will have consequences in the adventure's storyline. Game is provided with number options, and text to display their result. Users can load/save their game and choose different UI in the option section in the game.

ii) Below is a table with the examples how to apply testing in both agile and waterfall:

Agile	Waterfall
<ul style="list-style-type: none">- Use integration testing to see if one component for example the text display matches with the user's choosing option (another component).- We can use unit testing if in the future, the game wanted to add new feature such as a different storyline into the game and test would that unit work.	<ul style="list-style-type: none">- We can test the system as a whole as the game is straight forward and see if the game performs as expected as there is not much requirements to implement.- We can use acceptance test and let the consumer test out the program once it hit the demoing stage and see how the game perform and detect any weird thing during their experience.