



Individual Report

Roger Tan

1 Job Role

1.1 Description

The main scope of my job was to ensure that all the employees adopt Test Driven Development (TDD) when appropriate and tests meet the Testing Standards Document (eCook Testing Standard Document). It also consists of providing help and information on testing and integration of the product to the employees.

2 Work Carried Out

2.1 Deliverables from Metrics

In order to ensure that the Integration process runs smoothly, an Integration Guidelines Document (eCook Integration Guidelines) was made to provide employees with information on how to integrate their completed tasks into an existing product. Once integration is done, the employee would have to run the Deterministic Test Pass (DTP) to ensure that the integration process did not break the existing product. I will then review each DTP record to see if any features or functions of the product has broken or is not working to its desired purpose. The broken features or functions are considered as bugs where the employee who is responsible for the feature would then have to fix it.

There is also review (eCook Testing Standards Review Document) on the whole product to ensure that each Class has its own Test Class. Each Test Class is also reviewed to ensure that they follow the guidelines provided in the Testing Standards Document.

I worked on eCook's Testing and Integration Plan document, the Testing Standards Document and the Testing Standard Review document as well as reviewed eCook's DTP Record with the software team.

I made sure to modify the document to include instructions to the DTP section.

2.2 Additional Work

- I made slides for the product's features in the tender presentation.

2.2.1 eCook's QA Manual

- Did the Introduction, Organisational Structure and Testing & Integration Manager's Section

2.2.2 MainMenu Class

- Did the MainMenu Class & Test Class together with Zayyad and Prakruti

2.2.3 SlideShow Class

- Modified the SlideShow Class for "Exit to Main Menu" Feature
- Modified the SlideShow Class to call the VideoHandler Class for each video object
- Modified the SlideShow Class to call the GraphicsClass for each graphic object
- Modified the SlideShow Class for Slide Controls. (Next, Previous, Exit, Timer Buttons)
- Modified the SlideShow Class to have the Timer added to the NotesGUI Class so that the timers will be displayed in the NotesPanel

2.2.4 ImageHandler for Moshi-Mushi

- Did the ImageHandler together with Jonathan and Zayyad

2.2.5 MediaControl

- Did the MediaControl Class together with Zayyad. Did the Test Class alone.
- Improve the GUI of the MediaControl by using css. This is done together with Zayyad.

2.2.6 VideoHandler

- Did the VideoHandler Class together with Zayyad. Did the Test Class alone

2.2.7 GraphicsHandler

- Did the GraphicsHandler Class together with Zayyad. Did the Test Class alone

2.2.8 GUI package (Further Improvement on the GUI was passed on to Ankita and James)

- Did the GenerateShoppingListScreen Class and Test Class together with Zayyad
- Did the IngredientsScreen Class and Test Class together with Zayyad
- Did the LoadExternalRecipe Class and Test Class together with Zayyad
- Did the MainMenuContent Class and Test Class together with Zayyad
- Did the RecipeScreen Class and Test Class together with Zayyad

2.2.9 AudioHandler

- Did the AudioHandler Test Class with Paul

2.2.10 Timer

- Use ComboBox instead of Listview for time selection
- Stop the NotePanel's timeline when selection of time is happening
- Plenty of bug fix on the Timer Class (e.g., Timer doesn't continue to count when slide progression
- Improve the GUI of the Timer using css1

2.2.11 notesGUI Class

- Modified the notesGUI class to add the Timer into the NotesPanel

2.2.12 Bug Fixes

- Did a few bug fixes on the Timer, VideoHandler, Slideshow and MainMenu Class

2.2.13 css.css

- Made the CSS together with Zayyad to configure the looks of the overall product

2.2.14 css1.css

- Made the CSS to configure the looks of the Timer

3 Conclusion

3.1 Self-Critique

Initially when the company decided to adopt JavaFX, to write a functional Junit test was nearly close to impossible as JavaFx's Class are incompatible with Junit. Therefore, at the start whenever any JavaFx Class was required in the project, the tests were run manually. Hence, testing was not up to standard.

However, as more research was done, the company stumbled upon the JavaFX Threading Rule which made JavaFx's Class compatible with Junit. Therefore, automated Junit Test was able to run. Combining both automated and manual alongside with constant reviews on the Test Class as mentioned above, hence the testing procedure of the product were up to standard.

Having followed closely the progress of the product, there are constant updates on the DTP records as well.

Individual timesheets and weekly reports were handed in promptly and having done substantial amount of coding.

3.2 Evaluation of Contribution

Therefore with the above deliverables and self-critique, I would say I have fulfilled my role and contributed substantially to the company.