

Meeting 1

Week 8

Things to Do!

Notes.

JavaFX

Adopting Java FX; API generally quite good so check out API.

<http://download.java.net/jdk8/jfxdocs/> <- JavaFX

Pauls Feedback

Functional Spec. – Good

Tender Presentation – Very good feedback

Breaking Down of User Stories

Are we confident breaking down the user stories?

- Our user stories are technical, more so than last years; they consider functionality of the product
- Consider the PWS, the user stories were taken too literally, the PWS explains in more detail
- Consider a JPanel to display something, it should not open until the person implementing it has implemented the story and told it to open

Example of how to break down a user story:

- Be able to take in a String from the XML file and alter the font based on this String

The user stories are relatively high level; they can be broken down multiple times. You can create sub-stories from the user stories and then create tasks from the sub-stories. The tasks are as low level as can be and refer back to the PWS.

AGILE Workflow

Remember to write as much description as possible in the description fields. Also difficulty level is a good box to fill out.

Log your work for each sub-task with descriptions.

Testing

Tests should be submitted with the code.

There is a DTP document that describes the testing process.

All testing should be logged; for example if you were testing a GUI element then still write a test class but write the test as a comment within the test class.

Ensure that the tests have sufficient comments to explain their function. Methods should all have:

```
/*  
  
/  
  
*/
```

style comments above them to explain the method.

Referring to Testing Standards Document: ensure that programmers name, date created etc. are included at the top of the test in a comment.

Also take a look at the test structure:

```
public class AccountManagerTest {  
  
    private AccountManager manager;  
  
    private StudentAccount account;  
  
    @Before  
  
    public void setUp() throws Exception {  
  
        manager = new AccountManager();  
  
        manager.addAccount("Joe Bloggs");  
  
        manager.addAccount("Jonas Johnson");  
  
    }  
  
    @Test  
  
    public void createMultipleAccounts() throws Exception {  
  
        account = manager.getAccount("Joe Bloggs");  
  
        assertEquals("Joe Bloggs", account.getAccountID());  
  
        account = manager.getAccount("Jonas Johnson");  
  
        assertEquals("Jonas Johnson", account.getAccountID());  
  
    }  
  
}
```

Integration

After integration use a TDP document, attach this to ticket (evidence of testing!), to ensure that it meets the required tests.

Final product folder contains all integrated code. When implemented **DELETE** your previous stand-alone folder of code.