**Minutes of Meeting Held on Monday 12th September 2016**

Attendance: Nathaniel Smith, Tessa Marelic, Xinwei Jiang, Changze Huang  
Absent: NIL

1. Confirm minutes from previous meeting:

Previous minutes were confirmed as correct.

1. **Follow up on action items from previous meeting:**

As we are starting the next assignment and entering the construction phase, all previous actions are complete and assignment one has been submitted.

1. **Individual reports from previous week:**

No individual reports, the next assignment is starting.

1. Items we discussed this week:

* Discussed assignment 2 and how the group should begin.
* The group created a list of artefacts that need to be completed in the construction phase and assignment 2 report. See appendix A
* Spoke to the tutors who suggested we start by choosing some important use cases. Build the gui and functionality that incorporate those use cases and create a state change diagram for the use case.
* The group decided to use QT creator as the Ide to build the application in so we can utilise the user interface capabilities of it with c++ code.
* To gain experience in QT everyone will first create their own log-in gui and state diagram for the use case of logging in.

1. **Action Items:**

|  |  |  |
| --- | --- | --- |
| **Action – what is to be done** | **Who will do it** | **When will it be done by** |
| Build the functionality and gui for use case: User wants  to log in | Everyone | Monday 19th September |
| Begin the Project Plan for the project | Tessa | Monday 19th September |

1. Date and time of next meeting  
   Monday 26/08/2016 2:30pm Building 3 Room 127
2. **Agenda for Next Meeting:**

* Evaluate progress on the build of each component.
* Create an architectural design
* Create a database design.

# **Appendix A:**

# List of Artefacts

* Design and implementation
  + Architectural Design
  + Uses Cases
  + Sequence Diagrams
  + Class Diagrams
  + State Diagrams
    - Bug - Status
    - User - reputation
  + Data persistence design
  + User – interface design
* Presentation
* Summary detailing the aspects
* Details of the construction phase
  + Elements added/updated
  + Unit Testing procedures
  + Summarise data from defects and integration reports
* Code
  + List all elements
  + Samples of support code such as unit testing
* Testing
  + Test plans
  + Test cases
  + Test reports
* Program
  + Forms
    - Login/Register()
    - Main/Search Bug(User Obj)
    - View Bug(User, Bug Obj)
    - Add bug(User, bug)
    - Edit Bug(user, bug)
    - Search User(User)
    - View Profile(User)
    - Edit Profile(User1, User2)
    - Add user()
    - Generate Report()
  + Classes
    - Bug.cpp
    - User.cpp
* Database Creation

# Coding Practices

do use **PascalCasing** for class names and method names.  
do use **camelCasing** for method arguments and local variables.

This->variable1 = variable1  
  
Example  
  
AddToBug(std::string bugTitle)

{

this->title = bugTitle

}

login\_LoginTextArea

addBug\_OkButton  
addUser\_OkButton