**LOGBOOK**

|  |  |
| --- | --- |
| **1** | **Date:** 09-04-2018 |
| **Outcomes** | |
| - Created GitHub repo  - Discussed task assignments  **Austin**:   1. *Epic Stories* 2. *Acceptance Criteria*   **Henry**:   1. *User Stories*   **Chen**:   1. *UML Class Diagrams*   - Work on epic stories first (early in week) so that there is time for user stories, acceptance criteria and class diagram  - Since User Stories component is large, Austin will help out Henry with parts such as acceptance criteria if Henry needs help  - Based on Epic and user stories, Chen will make class diagram | |
| **Reflection** | |
| Epic Stories component was not too hard since it was high level abstraction. Used RGB model to create each epic story from the problem statement given.  User stories were then just broken down from the epic stories and many cases were considered from that high abstractions story. | |
| **Obstacles** | |
| Since we have only been given a week before the first iteration, there has been some time constraints due to other assignments from other subjects and also from work outside of university. | |

|  |  |
| --- | --- |
| **2** | **Date:** 16-04-2018 |
| **Outcomes** | |
| - Discussed task assignments  **Austin**:   1. *Create classes based on UML diagram*   **Henry**:   1. *Create/manage database (SQLite)* 2. *refactor whole structure*   **Chen**:   1. *Create classes based on UML diagram*   - Discussed how we will store all necessary information i.e. accounts, events etc  - Since Henry has experience with databases, primarily SQLite, he will write code that will help us with our website. Austin and Chen will then make the skeleton of all the classes that need to be implemented. We gave ourselves a week to finish the first draft of our back end product. | |
| **Reflection** | |
| Iteration 1 was successful and we were all happy with our outcome. | |
| **Obstacles** | |
| We will all need to be on the same level in terms of knowing how to utilise the database if we are going to write methods that query the information needed. | |

|  |  |
| --- | --- |
| **3** | **Date:** 20-04-2018 |
| **Outcomes** | |
| - Discussed about how the website will look like and which front end programming languages to use ie. JavaScript, Split work into manageable portions.  **Austin**:   1. *Create the dashboard* 2. *Make dashboard similar to the index page*   **Henry**:   1. *Manage database, create index page* 2. *user management(login/logout), navbar* 3. *reusable component of list events*   **Chen**:   1. *Create add event page.*     - Will give ourselves a week to finish our portion and will meet back on Monday to discuss our next plan of action.  - Each team member is responsible of styling their portion using front end techniques but will need to make website look consistent in its looks and feel. | |
| **Reflection** | |
| Creating classes and also the methods for those classes was a success. We did not need to change our diagrams while creating our model. | |
| **Obstacles** | |
| We’ve encountered “python import loop” issue, solved by splitting flask\_sqlalchemy into a separated file. | |

|  |  |
| --- | --- |
| **4** | **Date:** 23-04-2018 |
| **Outcomes** | |
| - Continued working on our portions. Split up more work between the three of us.  **Austin**:   1. *Continue styling dashboard. Start on reset password page.*   **Henry**:   1. *Manage database* 2. *set up email* 3. *created event details page and implemented reserve/cancel reservation for session*   **Chen**:   1. *Create add event page.*   - Will give ourselves a week to finish our portion and will meet back on Monday to discuss our next plan of action.  - Will also need to start making tests for each component of website so that nothing breaks. Each team member will need to test their own portion. | |
| **Reflection** | |
| Website is starting to work together with all its components to show the correct data. Was able to also style the website so that it looks good. | |
| **Obstacles** | |
| None at this point. | |

|  |  |
| --- | --- |
| **5** | **Date:** 06-05-2018 |
| **Outcomes** | |
| - Continued working on our portions. Split up more work between the three of us.  **Austin**:   1. *Start on reset password notification page.* 2. *Continue styling dashboard and reset pages.*   **Henry**:   1. *Managed maximum login attempt* 2. *added filter to list\_events component and it can be used in dashboard page* 3. *improve security (avoid url bypass)* 4. *add notification for logout/reserve event*   **Chen**:   1. *Continue adding fields for add event page and also album* | |
| **Reflection** | |
| Identified some bugs in front end but was able to fix it through communicating with team mates about these bugs. Most user stories have now been implemented and we now need to create tests to see if anything breaks. Also refined code so that easy to read and maintain. | |
| **Obstacles** | |
| Hard to wrap reusable component in flask | |

|  |  |
| --- | --- |
| **6** | **Date:** 13-05-2018 |
| **Outcomes** | |
| - Continued working on our portions. Split up more work between the three of us.  **Austin**:   1. *Style Dashboard page* 2. *refactored methods of pagination*   **Henry**:   1. *Created deregister time frame for back-end support* 2. *UI improved*   **Chen**:   1. *Added deregister timeframe filed on add event page* 2. *add event page validation* | |
| **Reflection** | |
| Ensured all functionalities have been implemented according the user story that we wrote, improved readability of the code by code refactoring and cleaning up | |
| **Obstacles** | |
| Manuel test for each functionality seems to be vulnerable and time costuming as we haven’t set up unit test and integration test yet. | |

|  |  |
| --- | --- |
| **7** | **Date:** 21-05-2018 |
| **Outcomes** | |
| - Continued working on our portions. Split up more work between the three of us.  **Austin**:   1. *profile page for guest speaker* 2. *refactor code for event management*   **Henry**:   1. *implemented guest login* 2. *improved UI profile page* 3. *added attendance bar on list event component* 4. *added guest registration modal* 5. *implemented registration fee*   **Chen**:   1. *fixed bugs for add event* | |
| **Reflection** | |
| Iteration 2 kicked off and task splitting. We essentially implemented all required features(guest login/sign in, profile page, attendance indicator on index page) by specification as the most complicated part(user is able to attend multiple session for a event) was implemented in last iteration, also we fixed up some remaining bugs from last iteration. | |
| **Obstacles** | |
| Our reusable component of list event doesn’t work on profile page (URL of image used relative path of current page rather than abs path from root), however, it can work smoothly on dashboard page. Solved by adding a url\_for with a function to retrieve image from backend. | |

|  |  |
| --- | --- |
| **8** | **Date:** 28-05-2018 |
| **Outcomes** | |
| - Continued working on our portions. Split up more work between the three of us.  **Austin**:   1. *added exception handler* 2. *added test cases* 3. *drafted final report* 4. *updated velocity chart* 5. *update user stories* 6. *adjusted reset password/forget password page and allows it work for guest user*   **Henry**:   1. *fixed bugs for add event* 2. *code clean up* 3. *completed ER diagram basing on our DB design* 4. *adjusted UML diagram* 5. *added more test cases* 6. *added more exception handler* 7. *implemented early bird* 8. *adjusted models with constructor*   **Chen**:   1. *fixed bugs for add event (typeahead duplication, empty input when switch)* 2. *fixed wording* 3. *updated message dynamically when event listing has nothing* | |
| **Reflection** | |
| We focused on testing and enhancement after competition of all features before completed the final iteration of development. Some bugs are identified from unit testing and manual test for different functionalities as we don’t have integration test. According these issues identifications, we fixed the bugs as well as implemented some exceptions handlers and error handling, especially for edge case to improve scalability and reduce issue identifying process of our application. | |
| **Obstacles** | |
| None at this point. | |

