Information for Team,

Sorry I can’t be there at the start I should hopefully be in later but not sure.

Here is some information for you,

Variables and Classes:

See attached Screenshots

Testing Used within the Code

We used scenario testing within the system, this allows the system to tell us if different bits work.

The scenarios in the system are as followed:

Add Book

Edit Book

Remove Book

Add User

Edit User

Delete User

Add Fine

Edit Fine

Delete Fine

Take Out Book

Return Book

User logs in

User logs out

My Paragraph on the Programming

The programming was done in NetBeans and we programmed it in Java and used JavaFX in the UI, we used an increment approach to the programming, this allowed us to build the system up part by part and making sure that the basics worked before we tried to add extra features to the system. The first increment was to make the backend of the system and to make the FXML Files for the UI, when making the backend we used a testing first approach which meant that we came up with many different scenarios that fitted in with the core of the system for example User Taking out a book, we would then build the system to make that scenario passed. What was good about this approach was that we could make sure that we haven’t messed anything up when we were making new parts of the system as if we changed any code that a scenario needed it would fail and we can quick see that we have broken the system and can fix it as the code changes are fresh in our heads. The second increment we did was about adding extras to the system and making the system better for the user. With this increment we thought about the extras that are needed in a library like different users, Users being able to change details, History of the Books being taken in and out, Copies of Books, Fines if a book is overdue, Admin Controls of Users. After adding one of the extras to the system we checked that we had not affected the main system by using the scenario testing. The final increment we did was about the future of the system. With this increment we looked about how maintainable the code was, maintainable code is good for anyone new looking at the code and for fixes to happened quickly as you can understand the code easier, we made sure that comments were placed all around the system, all variables were given correct and understanding names. We then looked at making sure the program ran the most effective it could this was done by making sure that we didn’t have any unused code or variables and making sure we not calling any useless functions or got functions that we used in development but not used in the final system.