# 1. Status summary :

**Name**: Library Management System
**Team members:** Shashankit Thakur , Harsha Kalmath

    a. **Work done:**

Created backend REST Apis using SpringBoot for most of the functionalities that we are trying to achieve like , login/signup , search book catalog using various filters, update book status and count.
All the apis are manually tested using postman
Added connection to MongoDB database. Created the necessary tables in the database.
Implemented factory pattern for account creation. Account can be of type Member or type librarian.

    b. **Changes or issues encountered:**

Initially for our database design , we had 3 collections that we were going to build. But we realized that we had a redundant database that stored bookids and member ids separately to maintain the record of books issued by members of the library. Instead , we chose to make a single account database which will store 2 types of records , librarian and member login details. The member record will also have bookids and number of books issued fields associated with it.
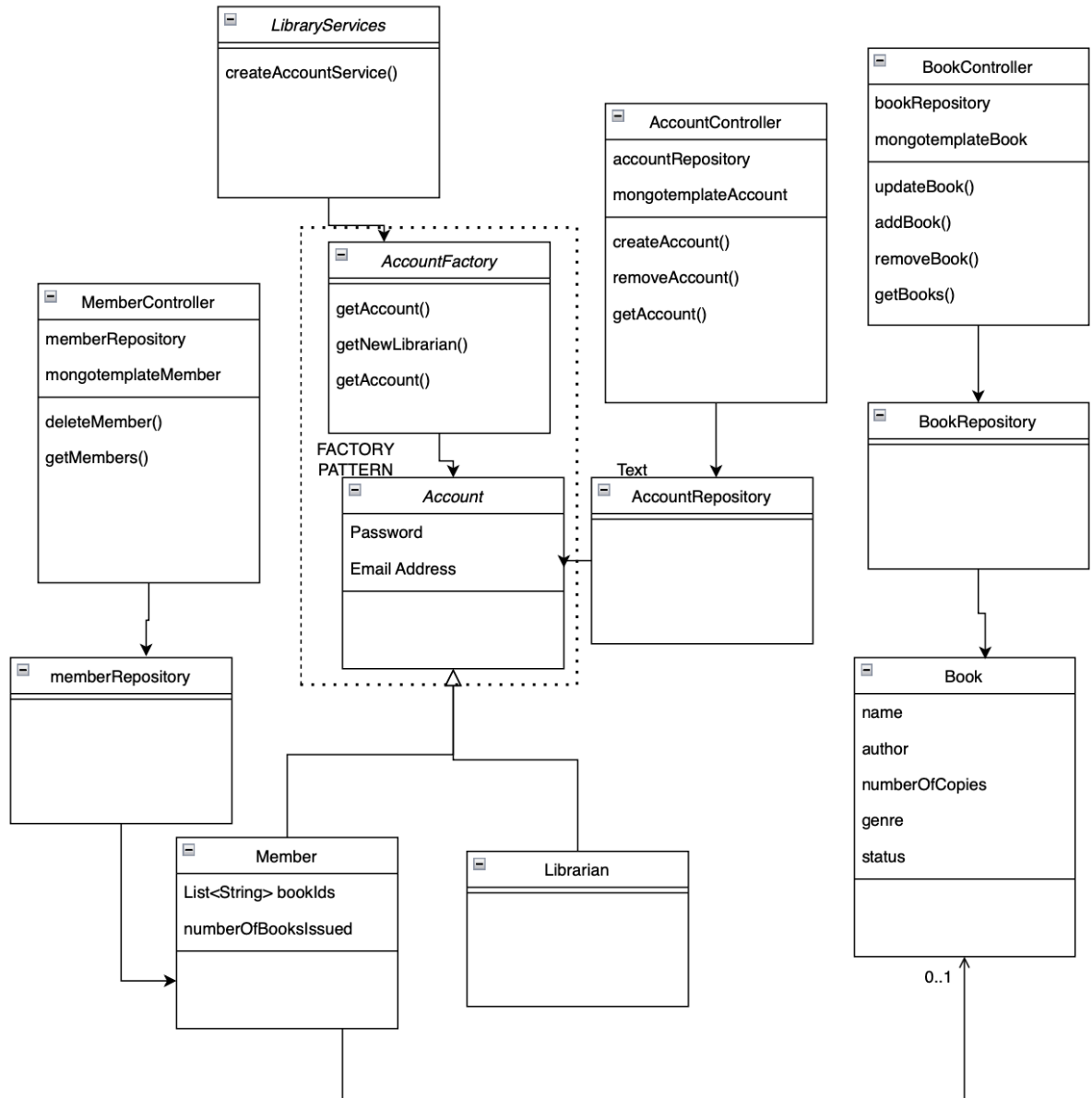We have another collection which stores the books in the library.
We have controller classes ( which has the code for the REST APIs) , the LibraryService class which will enclose all the business logic of our application , and the RepositoryClasses for dependency injection of the database logic with the java code.

    c. **Patterns:**

Factory pattern has helped us to delegate the responsibility of the appropriate subclass creation to the individual subclasses , depending on the scenario. It helps us in writing more generic functionality and code for the parent class (account class in our case) , like account deletion , checking membership access etc. We do not have to worry whether the account class object is of the type librarian or member.

## 2. Class diagram:



## 3. Plan for next iteration:
We plan to implement the front end of our application and integrate it with our backend.
We will be adding a terminal interaction between the user and the library system.
We will also add the functionality for handling book return policy and implement strategy pattern for the same.
We will be adding an observer pattern for our logs and debug statements.
We will also be adding more service methods in the library service class to handle books issued by members.