RDBMS -> MySql

**Table Name -> Members**

Coln -> ID (Primay Key)

name , roles, username, password (encrypted format)

**Table Name -> Books**

Coln -> UUID (PK)

Title, Author, Edition, Category, Status (AVAILABLE, BORROWED and OVERDUE), Rack

**Table Name -> User\_Book\_Issue\_Details**

Coln -> Book\_uid (FK), user\_id(FK), Return Date, Issue Date,

1. POST - /login

{

“username”:

}

Response:

{

"result":{

"messageCode":"OK",

"message":"Login successful",

"status":200

},

"user" : {

"customerId":80975412,

"customerName":"Shivam Thombre",

"roles" : “user”,

}

}

**User Authenticated Successfully**

1. GET -> /books/{1234}/{Sherlock}/{fiction} - Anshul

Response ->

{

books: [

{

“title” : “”, “author : “”, “edition” : “”, “status”: “”, “Rack”: “”

},

{

“title” : “”, “author : “”, “edition” : “”, “status”: “”, “Rack”: “”

}

]

}

1. POST : /borrowed ----- Shivam

**Request ->**  { “user\_id”: , “book\_id”: }

**Response -> {**"result":{  
"messageCode":"OK",  
"message":"Has borrowed",  
"status":200  
}

1. POST: /return/{user\_id} --Anshul

Request:

{

“booksIdToBeReturned”: [

“1234”, “5678”, “9012”

]

}

Response:

books: [

{

“bookId” : “1234”, “status : “AVAILABLE”, “charges” : “0.0”,

},

{

“bookId” : “5678”, “status : “AVAILABLE”, “charges” : “10.0”,

},

]

1. GET -> /books/{user\_id}?startDate=01/01/2021&endDate=31/01/2021 -- Shivam D

{

"result":{

"messageCode":"OK",

"message":"Book History",

"status":200

},

"BooksHistory":[

{

"userId":"987654sfds1311", "BookId": "1234567888", "Author": "sfdfdf", "title":"Sherlock HomeS" "Issued Date":4/12/12, "Return Date":"12/12/12"

},

{

"userId":"987654sfds1311", "BookId": "1234567888", "Author": "sfdfdf", "title":"Sherlock HomeS", "Issued Date":4/12/12, "Return Date":"12/12/12"

},

{ "userId":"987654sfds1311", "BookId": "1234567888", "Author": "sfdfdf",

"title":"Sherlock HomeS", "Issued Date":4/12/12, "Return Date":"12/12/12"

}

]

}

1. GET / -- Vishal

POST: /addNewBook

Request:

{

“title”: “”,

“author”: “”,

“uuid”: “”.

“edition”: “”

}

Response:

{

“title”: “”,

“author”: “”,

“uuid”: “”.

“edition”: “”,

“status”: “”,

“rack”: “”

}

com.hackathon.book;

Controller Class Name – BookController

Handler Class Name – BookHandler

Service- BookService

Repository -> BookRepository extends JPARepository (interface)

B