StartUp submission 5 Report

Lingering Bugs / Issues / Dropped Features

We remove category feature in searchpage. Replace it with recommandation and history feature.

We temporarily remove mailbox feature, because Kai doesn't finish his work on time. Warning letter was already sent out. John and prof. Tim should receive an email about that.

Ye Jiang:

contribute function:

>getFeedData, getFeedItemSync
>getUserData
>getExchangebook
>getNeedbook
>getMail
>ResetDatabase

HTTP Request/Routes

getFeedItemSync : This is a local function that used by other funtion to change the number in the list into real book object

getFeedData: GET /feed/:userid , which replace the the getfeeddata() funtion in client side server.

This function could give a list of book that in both exchange and need book in current user's list, which could only be access by this user who has the current userid.

getUserData: GET /user/:userid , which replace the getUserdata(user,cb) funciton in client side server.

This funtion will send a user object to client which will contain some useful data inside,and only the user itself could get this object.

getExchangebook: GET /user/:userid/exchangebooks, which replace the getEchangebook(user,cb) function in client server.

This fucntion will send all books that in current user's exhange list, current user and other user who may want to konw what book current user want to exhange are all access to this book list.

getNeedbook: GET /user/:userid/needbooks, which replace the getNeedbook(user,cb) function in client side.

This function will send all books that in current user's need list. only current user may access this book list.

getMail: GET /user/:userid/mailbox, which replace the getMail(user, cb) function in previous client server.

This function wil send all the mails in current user's mail box and only current user could access those mails.

ResetDatabase : This function is used to reset database in order to debug and we also change it into a react component in client side.

Zhuodong Huang:

contribute:

>move database from client to server
>getbookitemscollection

>addhistory
>gethisroty
>myfilter

HTTP Request/Routes

getbookitemscollection: a helper function in server to get the “bookitems” collection. It will return an array that contains all the Json objects in the “bookitems”. This function uses in searchpage for search, and use to display all the book in our application.

addhistory: a helper function in server to add book into user’s history list after user watch that book.

PUT /user/:userid/historys/:bookid It replaces the addhistory mock server function. This http route will call the helper function addhistory. it will add the bookitem in to user’s history list. This http request is authorized to the request which has the same user id as the user in database. Then it will add the bookitem in to user’s data.

GET /user/:userid/historys It replaces the gethisroty mock server function. This http request will get the bookitems from user’s history list. It return data only when the request’s userid is the same as the one it try to get data from.

GET /bookscollcetion/:searchTerm it replaces the myfilter mock server function. This request will can the helper function getbookitemscollection to get all the book in our database. And then user searchTerm from request to search for books. It will return an array of bookitems that relate to searchTerm. This http request is available for all request id.

Zhongce Ji:
contribute:
>getbook
>postbook
>postcomment

HTTP Request/Routes

POST /book {bookbody} It replaces the postbook mock server function. This http request will post the book to the database. And it will return nothing. And just do the adding the book.

GET /book/:bookid It replaces the getbook mock server function. This http request will get the specific book from out databse. It will return the book data and for this function we do not need to check the authciation

PUT /book/:bookid/commentthread/comment {commentbody} it replaces the postcomment mock server function.This request will return the bookdata with the new comment in the comment array in the database. The function will return the data when the user is the right user.