**Team 3 | Friday 10:30am | Tutor: Dipto**

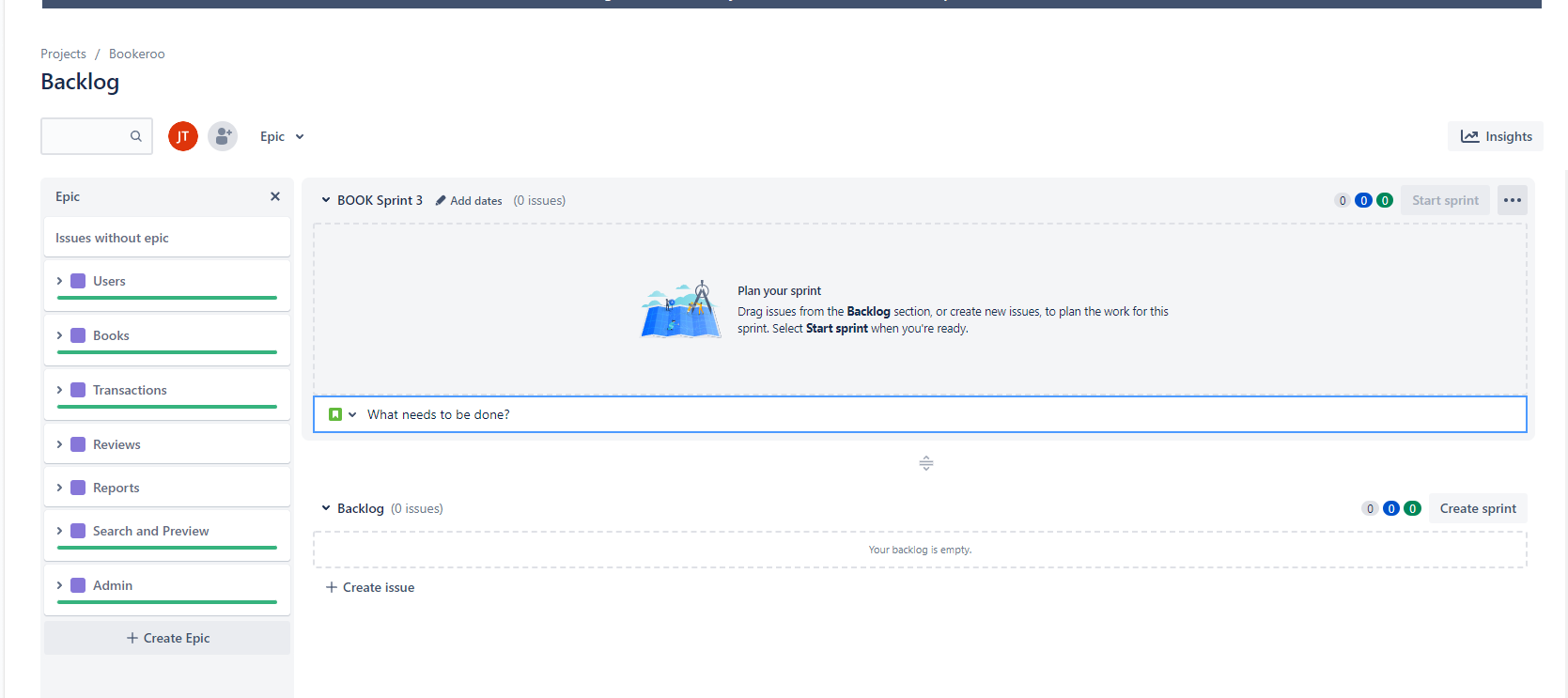
|  |  |  |  |
| --- | --- | --- | --- |
| **Member** | **Student Number** | **Role** | **% Contribution** |
| Hibbaan Mohammed Nawaz | S3847243 | Front-End Developer | 25 |
| Ross Alexopoulos | S3839735 | Back-End Developer | 25 |
| Jeffrey Tan | S3851781 | Scrum Master, Front-End Developer | 25 |
| Allister Toos | S3843971 | Back-End Developer | 25 |

**Team Members + Peer Assessment**

**Links**

**Jira**

<https://bookeroo.atlassian.net/jira/software/projects/BOOK/boards/1>



^^Product Backlog Done!

**Teams**

<https://teams.microsoft.com/l/channel/19%3aQhGbNrgTOJSzYUYUzNyadTyARZ0x6OFaL9RpSzWaUm01%40thread.tacv2/General?groupId=1fa623d5-db2b-4891-b30c-4b3ecfb81383&tenantId=d1323671-cdbe-4417-b4d4-bdb24b51316b>

**GitHub**

<https://github.com/s3847243/Bookeroo>

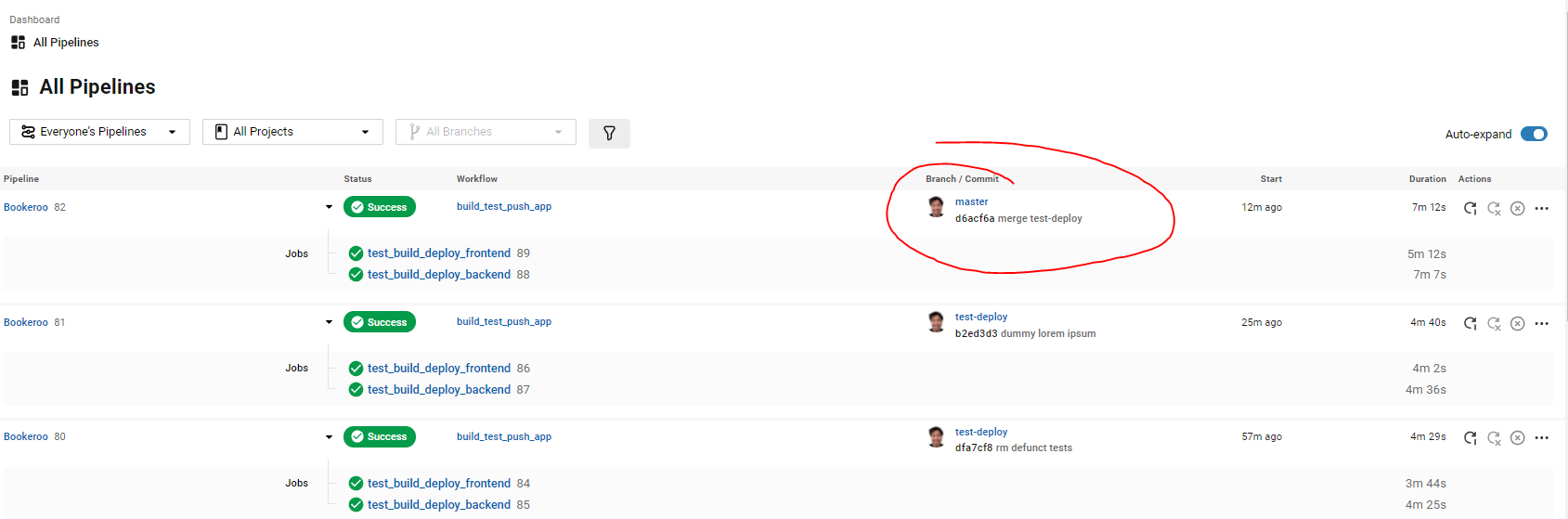
**CircleCI**

<https://app.circleci.com/pipelines/github/s3847243/Bookeroo?invite=true>

**Items contributed**

|  |  |
| --- | --- |
| **Member** | **Items** |
| Hibbaan Mohammed Nawaz | Front-End and Tests for the following…  PBI: 3, 7, 12, 13, 16  Displaying transaction history, cancelling orders. Creating book listings. |
| Ross Alexopoulos | Back-End and Tests for the following…  PBI: 11, 12  Listing and reviews microservice |
| Jeffrey Tan | Front-End and Tests for the following…  PBI: 11, 14, 15, 21  Post and see reviews, checkout transactions. |
| Allister Toos | Back-End and Tests for the following…  PBI: 7, 14, 15, 16 |

**CircleCI Screenshot**



**DockerHub Screenshot**

These are our public containers used.

Graphical user interface, text, application

Description automatically generated

**Summary of User Stories, Code, Tests, Functionality**

|  |  |
| --- | --- |
| **PBI** |  |
| **3** | As an admin, I want to download reports about books, transactions, and other business data in CSV format, so I can conduct stock-take and complete accounting forms. |
| Back-End |  |
| Front-End | **Code:**  Reports.js in Transactions.js   * React component to convert data into CSV |
| **7** | As an admin user, I want to view a summary of all past and current transactions (sorted by date), so I can keep track of business profits. |
| Back-End | **Code:**  In ***transactionmicroservices,***  TransactionController.java   * allTransactions() handles web requests   transactionService.java   * holds logic to interact with database   **Tests:**  In *test/java/com.rmit.sept.bk\_transactionservices* |
| Front-End | **Code:**  Transactions.js   * render transaction info   actions/transactionActions.js   * Holds the function to send API request |
| **11** | As a public user, I want to review books that I have purchased, so I can give my thoughts and opinions on that book. |
| Back-End | **Code:**  In ***bookmicroservices,***  ReviewController.java   * allReviews() handles web requests   ReviewService.java   * holds logic to interact with database   **Tests:**  In *test/java/com.rmit.sept.bk\_bookmicroservices* |
| Front-End | **Code:**  BookDetails.js, BookReview.js, CreateBookReview.js   * React components to display reviews, and create review form.   actions/reviewActions.js   * Api calls for getting and posting reviews   **Tests:**  BookReview.test.js, CreateBookReview.test.js |
| **12** | As a business owner, I want to create listings for my books, so that other users can purchase those items. |
| Back-End | **Code:**  In ***listingmicroservices,***  ListingController.java   * createListing() handles web requests   ListingsService.java   * holds logic to interact with database   **Tests:**  In *test/java/com.rmit.sept.bk\_listingmicroservices* |
| Front-End | **Code:**  SellBook.js   * React form to create a listing   BookDetails.js & BookListing.js   * React components to render listings from backend   actions/listingActions.js   * Api calls for getting and posting listings   **Tests:**  BookListing.test.js |
| **13** | As a public user, I want to create share listings for my books, so that I can provide others with books I don't need. |
| Back-End | **Same as PBI 12** |
| Front-End | **Same as PBI 12** |
| **15** | As a public user, I want to cancel (and refund) a book purchase within 2 hours of ordering an item, so I can change my mind on a product. |
| Back-End | **Code:**  In ***transactionmicroservices,***  TransactionController.java   * updateStatus() allows status to be updated to cancelled.   TransactionService.java   * updateStatus () holds logic to interact with database   **Tests:**  In *test/java/com.rmit.sept.bk\_transactionservices* |
| Front-End | **Code:**  Transactions.js   * React component to render transactions as well as the button for cancelling orders.   actions/transactionActions.js   * Holds the function to send API request to update status of order |
| **16** | As a registered user, I want to view my transaction history (buy and sell), so I can track my sales and expenses on BOOKEROO. |
| Back-End | **Same as PBI 12** |
| Front-End | **Same as PBI 12** |
| **21** | As a user, I want to see the availability of a book, so that I can know whether or not I can order it. |
| Back-End | **Code:**  In **listing*microservices,***  ListingController.java   * getByIsbn() returns listings for current book   ListingsService.java   * findByIsbn() holds logic to interact with database   **Tests:**  In *test/java/com.rmit.sept.bk\_listingmicroservices* |
| Front-End | **Code:**  BookDetails.js & BookListing.js   * React components to render all listings returned from API   actions/listingActions.js   * Holds the function to send API request to get all listings for given book   **Tests:**  BookListing.test.js |

**Communication Logs**

Aside from weekly Stand-Up meetings (as demonstrated by the meeting minutes), our team conducted text-based communication via Discord (an app like Slack). To demonstrate our improvement in communication, the following are the log frequencies of last vs this sprint. The number of results is the number of messages sent in the text channel.

We had 48 more messages than last Sprint. Ross and Allister both improved in communication.

|  |  |  |
| --- | --- | --- |
| Member | Last Sprint | This Sprint |
| All members |  |  |
| Jeff |  |  |
| Mohammed |  |  |
| Ross |  |  |
| Allister |  |  |