

Peer Assessment

Team Name: Moving Houses

Team member:	Contribution (%):	Items Contributed:
Jared Song S3857657	20	<p>Functionality for users, books, browsing and requests, including the models, repositories, services and controllers.</p> <p>Added security measures for duplicate users or books, so that no identical book or user may exist.</p> <p>Added queries for updating users, books, and searching for books with a given title, author name, ISBN, category and seller. Automated the set status and role when creating user accounts, and implemented interactions between requests and the relevant models.</p> <p>Postman testing and unit testing.</p> <p>Assisted in setting up the RDS SQL database for PostgreSQL.</p> <p>Product and Sprint Backlog grooming and scrum documentation as scrum master. Jira documentation and updates.</p>
Aili Gong S3443647	20	<p>Tested the backend through Postman.</p> <p>Created the login/signup page and connected to backend.</p> <p>Created the home and book page which retrieve the data from back end.</p> <p>Created user dashboard and connected to the backend, which allows user to view personal info, transaction history etc.</p> <p>Created the dashboard for admin and connected to the</p>

		<p>backend, which allows admin to do work.</p> <p>Created tests for front end.</p> <p>Connected the project to CircleCI.</p> <p>Create search function in front end.</p> <p>Screenshot the front end for documentation.</p>
Alex Aloï S3842524	20	<p>RDS Postgre SQL Database Schema</p> <p>Model connections for everything in schema</p> <p>Hibernate join on objects needed, represented through schema</p> <p>Dockerise app</p> <p>Running the Docker image through CircleCI</p> <p>User stories 15, 16, 17, 24, 62</p>
Carl Karama S3713721	20	<p>Creating backend functionality for User and Book Reviews including model repository, service and controller</p> <p>Product Backlog and Sprint Planning grooming</p> <p>Postman testing</p>
Shannon Dann S3858053	20	<p>Transaction functionality including model, repository, service and controller.</p> <p>Business functionality including model, repository, service, controller and properly linking to users.</p> <p>Postman testing and unit testing.</p>