

## Acceptance Testing

SEPT - Group 02

Milestone 03

TestID	1.1
User Story	"As a user, I want to view the cover image of the book I'm looking for so that I can get a first impression of what the book might be about"
Goal	To ensure that the "view book" button is functional and users can select their desired book and view the books cover image
Acceptance Criteria	Given that the user is on the book products list and finds the book they are looking for When they click on the book product and the page has loaded up Then the cover image is viewable and the user can get a first impression of the book
Setup	<ol style="list-style-type: none"><li>1. cd into the backend</li><li>2. Run maven package and java jar files for all microservices</li><li>3. Cd into frontend in separate terminal</li><li>4. Run npm start</li></ol>
Steps	<ol style="list-style-type: none"><li>1. Go to localhost 3000 (<a href="http://localhost:3000/">http://localhost:3000/</a>)</li><li>2. Once on the homepage, click on the 'View Books' button</li><li>3. Select desired book from 'All Products' page</li><li>4. Verify the individual book page displays correct information without any errors</li></ol>
Expected Results	The 'Individual Book Page' page displayed without any errors.
Actual Results	The 'Individual Book Page' page displayed without any errors.
Bugs Found	None
Tester Name	Rylan Koneri

TestID	1.2
User Story	"As a user, I want to view the table of contents of the book I'm looking for so that I can acquire an idea of how long the book might be"
Goal	To make sure that the "view book" button works and users are able to select their desired book and read the page listings.
Acceptance Criteria	Given that the user is on the book products list

	<p>And finds the book they are looking for When they click on the book product and then click on 'Tables of Contents'</p> <p>Then the user can view the tables of contents and see how long the book is.</p>
Setup	<ol style="list-style-type: none"> <li>5. cd into the backend</li> <li>6. Run maven package and java jar files for all microservices</li> <li>7. Cd into frontend in separate terminal</li> <li>8. Run npm start</li> </ol>
Steps	<ol style="list-style-type: none"> <li>1. Go to localhost 3000 (<a href="http://localhost:3000/">http://localhost:3000/</a>)</li> <li>2. Once on the homepage, click on the 'VIEW BOOKS' button within the centre of the page</li> <li>3. Navigate with scroll wheel through book listings</li> <li>4. Select desired book and click button to view table of contents</li> <li>5. Verify the page displays correct product information without any errors</li> </ol>
Expected Results	The 'Table of Contents' page displayed without any errors.
Actual Results	The 'Table of Contents' page is displayed without any errors.
Bugs Found	None
Tester Name	Rylan Koneri

TestID	1.3
User Story	"As a user, I want to view the table of contents of the book I own so that I can skip to the chapters that sound interesting"
Goal	To ensure that the user is presented with a sign up page with appropriate fields for signing up.
Acceptance Criteria	<p>Given that the user is on the book products list And finds the book they own When they click on the book product and then click on 'Tables of Contents'</p> <p>Then the user can view the tables of contents and skip to the chapters they want.</p>
Setup	<ol style="list-style-type: none"> <li>9. cd into the backend</li> <li>10. Run maven package and java jar files for all microservices</li> <li>11. Cd into frontend in separate terminal</li> <li>12. Run npm start</li> </ol>
Steps	<ol style="list-style-type: none"> <li>5. Go to localhost 3000 (<a href="http://localhost:3000/">http://localhost:3000/</a>)</li> <li>6. Once on the homepage, click on the 'VIEW BOOKS' button within the centre of the page</li> </ol>

	<ul style="list-style-type: none"> <li>7. Navigate with scroll wheel through book listings</li> <li>8. Select desired book and click button to view table of contents</li> <li>9. Verify the page displays correct product information without any errors</li> </ul>
Expected Results	The 'Table of Contents' page displayed without any errors. User was able to access book chapters information
Actual Results	The 'Table of Contents' page displayed without any errors. User was able to access book chapters information.
Bugs Found	None
Tester Name	Rylan Koneri

TestID	1.4
User Story	"As an administrator, I want to login to an administrator account, so that I can add new users"
Goal	To ensure that the user is presented with a login page with appropriate fields for logging in.
Acceptance Criteria	<p>Given that an admin is on the website</p> <p>When I access the homepage and select SIGN IN icon</p> <p>Then I will be directed a separate page allowing me to sign in on the website</p>
Setup	<ul style="list-style-type: none"> <li>13. cd into the backend</li> <li>14. Run maven package and java jar files for all microservices</li> <li>15. Cd into frontend in separate terminal</li> <li>16. Run npm start</li> </ul>
Steps	<ul style="list-style-type: none"> <li>10. Go to localhost 3000 (<a href="http://localhost:3000/">http://localhost:3000/</a>)</li> <li>11. Once on the homepage, click on the 'login' button on the header</li> <li>12. Verify the page displays logs in the user without any errors</li> </ul>
Expected Results	The login page exists with appropriate input fields for users to enter their login details and are able to authenticate their credentials
Actual Results	The login page exists with appropriate input fields for users to enter their login details, but were unable to authenticate credentials
Bugs Found	Lack of validation and authentication for user logging in process
Tester Name	Rylan Koneri

TestID	1.5
User Story	"As a user, I want to create an account, so that I can purchase books suited to my taste."
Goal	To ensure that the user is presented with a login page with appropriate fields for logging in.
Acceptance Criteria	Given that a customer is on the website When I access the homepage and select SIGN IN icon Then I will be directed a separate page allowing me to sign in on the website
Setup	17. cd into the backend 18. Run maven package and java jar files for all microservices 19. Cd into frontend in separate terminal 20. Run npm start
Steps	13. Go to localhost 3000 ( <a href="http://localhost:3000/">http://localhost:3000/</a> ) 14. Once on the homepage, click on the 'login' button on the header 15. Verify the page displays logs in the user without any errors
Expected Results	The login page exists with appropriate input fields for users to enter their login details and are able to authenticate their credentials
Actual Results	The login page exists with appropriate input fields for users to enter their login details, but were unable to authenticate credentials
Bugs Found	None
Tester Name	Rylan Koneri

TestID	1.6
User Story	"As a customer, I want to search for a book by its ISBN, so that I can see if a book is available for purchase."
Goal	To ensure that the "view book" button is functional and users can search for their desired book through input of the books ISBN
Acceptance Criteria	Given that the user is on the book products list and finds the book they are looking for When they click on the search bar and enter the books correct ISBN Then search will display the book and it will be displayed for the user

Setup	21. cd into the backend 22. Run maven package and java jar files for all microservices 23. Cd into frontend in separate terminal 24. Run npm start
Steps	16. Go to localhost 3000 ( <a href="http://localhost:3000/">http://localhost:3000/</a> ) 17. Once on the homepage, click on the 'View Books' button 18. Click on search bar and enter ISBN details 19. Verify the search bar displays correct information without any errors
Expected Results	The search bar is accurately found and displays the book without any errors.
Actual Results	The search bar is accurately found and displays the book without any errors.
Bugs Found	None
Tester Name	Rylan Koneri

TestID	<b>1.7</b>
User Story	"As a customer, I want to search for books written by an author, so that I can see what books by that author are available for purchase."
Goal	To ensure that the "view book" button is functional and users can search for their desired book through input of the books author
<b>Acceptance Criteria</b>	Given that the user is on the book products list and finds the book they are looking for When they click on the search bar and enter the books author Then search will display the book and it will be displayed for the user
Setup	25. cd into the backend 26. Run maven package and java jar files for all microservices 27. Cd into frontend in separate terminal 28. Run npm start
Steps	20. Go to localhost 3000 ( <a href="http://localhost:3000/">http://localhost:3000/</a> ) 21. Once on the homepage, click on the 'View Books' button 22. Click on search bar and enter author name 23. Verify the search bar displays correct information without any errors
Expected Results	The search bar is accurately found and displays the book without any errors.

Actual Results	The search bar is accurately found and displays the book without any errors.
Bugs Found	None
Tester Name	Rylan Koneri

TestID	1.8
User Story	"As a customer, I want to search for a book by its name, so that I can see if a book is available for purchase."
Goal	To ensure that the "view book" button is functional and users can search for their desired book through input of the book title
Acceptance Criteria	Given that the user is on the book products list and finds the book they are looking for When they click on the search bar and enter the book title Then search will display the book and it will be displayed for the user
Setup	29. cd into the backend 30. Run maven package and java jar files for all microservices 31. Cd into frontend in separate terminal 32. Run npm start
Steps	24. Go to localhost 3000 ( <a href="http://localhost:3000/">http://localhost:3000/</a> ) 25. Once on the homepage, click on the 'View Books' button 26. Click on search bar and enter book title 27. Verify the search bar displays correct information without any errors
Expected Results	The search bar is accurately found and displays the book without any errors.
Actual Results	The search bar is accurately found and displays the book without any errors.
Bugs Found	None
Tester Name	Rylan Koneri