

Boilerbazaar

Sprint 3 Retrospective

Team 30: Dung “Ryan” Doan, Xavier Huu Pham, Jeffrey Yian Wang, Shicheng Fang, Michio L
Sekiguchi

What went well?

In general, we finished the rest of the features that we planned for our application. This includes functions related to interacting with other users such as following, blocking, and rating. We also developed our home page by moving some features to it such as previously viewed listings, reviews to give, and user searching.

User Story #1

As a user, I would like to see my previously viewed books on the home page.

#	Description	Estimated Time	Owner
1	Create UI summarizing a listing and link it to its listing page	4 Hrs	Ryan
2	Create Ajax function to get previously view textbook	1 Hr	Ryan
3	Create JS function to dynamically populate the UI	3 Hrs	Ryan
4	Create unit tests to test functionality of seeing previously viewed textbooks on homepage.	2 Hrs	Ryan

Completed:

When a user navigates to the home page, they can see a box to the bottom right with basic information of their 5 most recently viewed listing. The user can use the next and back button to view each listing one by one, and use the open button to quickly go to the listing's page.

User Story #2

As a seller, I would like to see which textbook a buyer messaged about.

#	Description	Estimated Time	Owner
1	Create algorithm to send textbook information along with message to seller.	2 Hrs	Michio
2	Create UI for textbook information.	4 Hrs	Michio
3	Create unit tests to test functionality of textbook messaging.	2 Hrs	Michio

Completed:

When a user is on the buy page and viewing a specific listing, they can message the seller. This message will open a new conversation with the seller. Each conversation displays the interested listing for both the buyer and seller.

User Story #3

As a student, I would like to be able to see a list of websites that are selling the textbook that I searched for with the cheapest price highlighted.

#	Description	Estimated Time	Owner
1	Create algorithm to parse html and pull data and/or use APIs.	6 Hrs	Shicheng
2	Create algorithm to highlight cheapest price.	1 Hr	Shicheng
3	Create unit tests to test functionality of finding websites with the cheapest price.	2 Hrs	Shicheng

Completed:

This user story is finished completely. When a user goes to a buy page of a listing, the user can see prices of the book on Ebay and Google Play, with lowest price highlighted.

User Story #4

As a student, I would like to be able to easily access the websites selling the textbooks.

#	Description	Estimated Time	Owner
1	Create algorithm to generate links to the website listings for user to click on.	2 Hrs	Shicheng
2	Create UI for links.	4 Hrs	Shicheng
3	Create unit tests to test functionality of generating links.	1 Hrs	Shicheng

Completed:

This user story is complete. When a user goes to a buy page of a listing, the user can see prices of the book on Ebay and Google Play, with lowest price highlighted. Besides, link to ebay and google play will also be highlighted.

User Story #5

As a student, I would like to be able to receive desktop notifications for price drops.

#	Description	Estimated Time	Owner
1	Add UI to opt in and opt out for desktop notifications	3 Hrs	Xavier
2	Adjust function for checking for price drops	2 Hrs	Xavier
3	Implement function for displaying desktop notifications	4 Hrs	Xavier
4	Create unit tests to test functionality of desktop notifications	2 Hrs	Xavier

Completed:

When the user navigates to the Watchlist section of the Settings page, they can press a button to either opt into desktop notifications or learn about how to opt out of desktop notifications depending on whether the appropriate permission is allowed. If desktop notifications are enabled, when price changes are detected on a listing in a user's watchlist when the user logs in, a desktop notification is displayed informing them of the change.

User Story #6

As a student, I would like to be able to follow a seller to see if they list any other textbooks.

#	Description	Estimated Time	Owner
1	Add UI to follow and unfollow a seller	2 Hrs	Xavier
2	Add UI for seeing a list of followed sellers with links to their profiles	4 Hrs	Xavier
3	Implement function to add a followed seller to the database	1 Hr	Xavier
4	Implement function to remove a followed seller from the database	1 Hr	Xavier
5	Create unit tests to test functionality of following a seller	2 Hrs	Xavier

Completed:

When the user navigates to another user's profile, they can follow or unfollow the other user with a button depending on whether they are currently following the other user. Users can see the profiles they are following from the Following section of the Settings page. This tells them the

number of sales and purchases of a user as well as their average rating. Users can click on the number of the user in their following list to navigate to their profile.

User Story #7

As a buyer, I would like to be able to see other textbooks that are related to the current listed textbook.

#	Description	Estimated Time	Owner
1	Add UI for displaying textbooks related to a listing	4 Hrs	Xavier
2	Implement function to find related textbooks	3 Hrs	Xavier
3	Create unit tests to test functionality of displaying related textbooks	2 Hrs	Xavier

Completed:

Related textbooks are defined as textbooks that share the same author and/or course. When a user navigates to a listing, they are presented with a list of related textbooks on the right side of the page if there are any in the database. The related textbook database is generated from every textbook that is historically listed on our site.

User Story #8

As a student, I would like to be able to add a profile picture.

#	Description	Estimated Time	Owner
1	Create button for user to add image.	1 Hr	Michio
2	Create algorithm to add image to database.	2 Hrs	Michio
3	Create algorithm to retrieve image from database.	2 Hrs	Michio
4	Create button for user to remove image.	1 Hr	Michio

5	Create algorithm to remove image from database.	2 Hrs	Michio
6	Create unit tests to test functionality of adding and removing a profile picture.	2 Hrs	Michio

Completed:

A user can go to their profile editing page and click on a button to upload an image that will update their profile picture. There is also a remove profile picture button that can be pressed only when the user has a profile picture. Their profile picture is displayed across the application.

User Story #9

As a student, I would like to be able to block another user from messaging me and viewing my listing.

#	Description	Estimated Time	Owner
1	Add block option on contact list, listing, and user profile	2 Hrs	Ryan
2	Generate algorithm to block a user from seeing listing/messaging.	2 Hrs	Ryan
3	Create UI for a blocked user trying to message a user that blocked them,	2 Hrs	Ryan
4	Create algorithm to update a user's block list.	2 Hrs	Ryan
5	Create button and algorithm to unblock user.	2 Hrs	Ryan
6	Create UI for a user to see their blocked users.	4 Hrs	Ryan
5	Create unit tests to test functionality of preventing block listings from appearing and blocking/unblocking a user.	4 Hrs	Ryan

Completed:

The option to block another user is present in their profile page, any of their listings, or in a message to them. Blocking makes it impossible for either user to search for each other and their listings, and to message each other. The option to unblock is present in the message page, or in a

separate page in settings called “Blocked Users”, which shows the user any user they have blocked.

User Story #10

As a student, I would like to be able to submit and view GroupMe links related to courses.

#	Description	Estimated Time	Owner
1	Create new navbar for GroupMe.	1 Hr	Michio
2	Create UI for GroupMe page.	4 Hrs	Michio
3	Create algorithm for submitting and verifying links.	2 Hrs	Michio
4	Create algorithm for retrieving links from database.	1 Hr	Michio
5	Create unit tests to test functionality of GroupMe functions.	2 Hrs	Michio

Completed:

A user can navigate to this page and view the current list of classes along with the links associated with it. The user can also submit a class and link and the page will update by adding the link to the page.

User Story #11

As a buyer, I would like to be able to give seller ratings based on interaction and ease of purchase.

#	Description	Estimated Time	Owner
1	Create UI prompt when seller sells item.	2 Hrs	Jeffrey
2	Create algorithm to add rating to sellers’ profile	1 Hr	Jeffrey
3	Create UI for creating seller reviews.	4 Hrs	Jeffrey
4	Create “ratings to give” box on home page.	2 Hrs	Jeffrey

5	Create unit tests to test functionality of seller ratings.	2 Hrs	Jeffrey
---	--	-------	---------

Completed:

When the buyer opens the home page, they can see a box with reviews to give to sellers that they just bought a textbook from. When clicking on the open button, a prompt appears for them to enter a review and rating for the seller. The seller rating is updated afterwards.

User Story #12

As a seller, I would like to be able to view reviews from my buyers.

#	Description	Estimated Time	Owner
1	Create a review page on settings page.	2 Hrs	Jeffrey
2	Create UI for review page.	4 Hrs	Jeffrey
3	Create algorithm to retrieve reviews about a seller from the database.	2 Hrs	Jeffrey
4	Create unit tests to test functionality of retrieving reviews.	2 Hrs	Jeffrey

Completed:

When going to the settings page and click on the ratings button. User can see the rating that they have received from the buyer.

User Story #13

As a seller, I would like to give reviews about my buyers.

#	Description	Estimated Time	Owner
1	Create algorithm to add review for buyer to database.	1 Hrs	Jeffrey
2	Create prompt for seller to choose who the buyer is.	3 Hrs	Jeffrey
3	Create UI for creating buyer reviews.	4 Hrs	Jeffrey
4	Create algorithm to update buyer ratings.	1 Hr	Jeffrey

5	Create unit tests to test functionality of buyer reviews.	2 Hrs	Jeffrey
---	---	-------	---------

Completed:

When the seller marks a listing as sold, they can select a seller who they sold the listing to. After they select a buyer, they are prompted with a rating and review to give. The buyer rating is updated afterwards.

User Story #14

As a student, I would like to be able to search for other users on the home page.

#	Description	Estimated Time	Owner
1	Create search field on home page.	1 Hr	Shicheng
2	Create UI for search page.	4 Hrs	Shicheng
3	Create algorithm to retrieve users from database.	1 Hrs	Shicheng
4	Create profile page UI when user clicks on another user.	4 Hrs	Shicheng
5	Create unit tests to test functionality of searching for users.	2 Hrs	Shicheng

Completed:

User story 14 is finished completely. When user searches by name and id on homepage, the website will display information of the user with a link to the user's profile.

What did not go well?

In general, mostly everything went smoothly this sprint. Everyone completed their user stories by the sprint demo.

How should we improve?

One thing that could be improved upon is documentation for the project. For the most part, we lacked documentation and comments that clarified how each part of the project, including functions, worked. Therefore, there were many times where we had to ask each other how a part of the project worked which impeded work.