# Unlimitrade

Ethan Doyle - ethanadoyle

Max Lane - MaxLane1033

Yasir Qasemi - YasirQasemi

Asher Siegel - asher-siegel-boulder

Connor Short - ConnorS7

Ivan Gorodinski - gorodinskiia

## Project Description

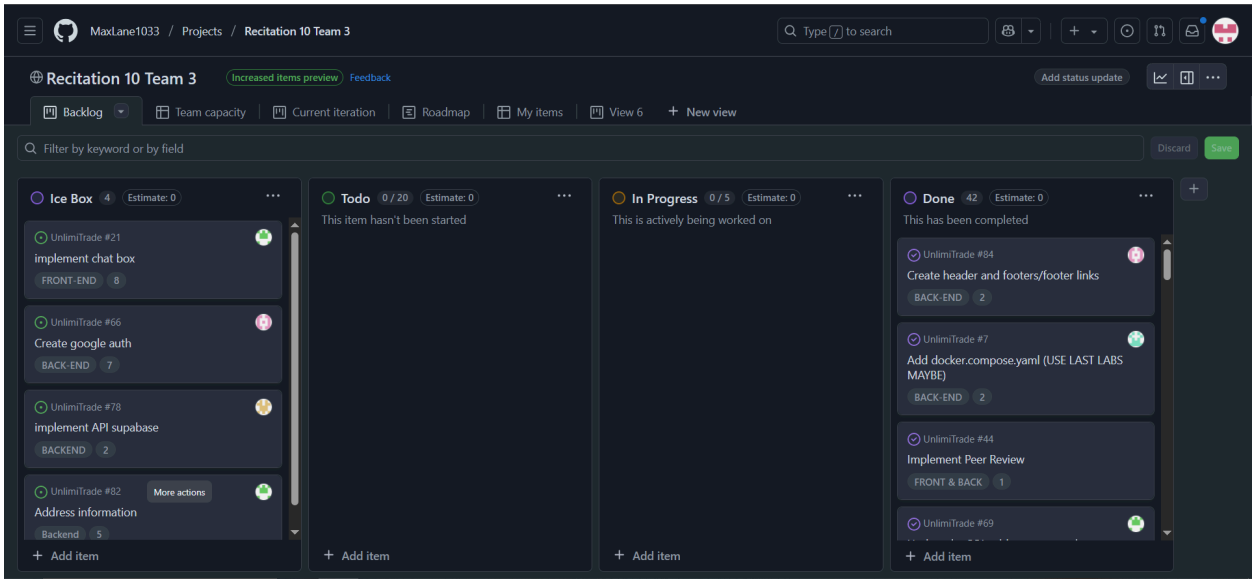
Our application is a trading platform where users can browse, post, and exchange items without involving money. Each user creates an account and has a public profile where they can manage their posted items and view trade history. There is no buying or selling on the platform, which means we avoid storing any sensitive payment information, reducing security risks.

Once a user registers and logs in, they are directed to the home screen where they can navigate to post new items, explore available trades, and manage their profile. On the browse page, users can filter items by category and choose an item they'd like to trade for. When initiating a trade, they select one of their own items and send a trade offer to the item's owner.

The receiving user is then able to view the trade offer, along with images and descriptions of both items, and either accept or deny the trade. Accepted trades update the item status and move the trade into each user's history.

The core functionality of the platform centers around the trading flow between users. Once a trade is proposed, both items involved are marked as pending, preventing them from being offered in additional trades until a decision is made. If the trade is accepted, the item statuses update to "traded" and the details are saved in each user's trade history. If the trade is denied, both items return to the available pool. This streamlined system allows users to engage in fair, organized item exchanges and keeps everyone's listings up to date in real time.

**Project Tracker - GitHub project board:** [Link To Project Board]




**Video:** [Link To Project Demo]

**VCS:** [Link To GitHub Repo]

**Contributions:**

- Ethan Doyle: I built the majority of the profile page, including the profile.hbs layout and the edit profile setup. I installed multer to allow for photo upload for posting and profile photo features. I implemented the logic to display posted items and trade history dynamically. I implemented the trade history system using SQL logic to ensure correct items and usernames appear based on user roles. I implemented the trade now button in browse and created the accept and deny functionality in the "My Trades" section, handling pending trade logic with session-based feedback.

- Max Lane: I implemented numerous individual features/aspects of the website, such as: the ability to leave reviews on a user's profile, the ability to see a list of a user's trades, I improved the design/layout of the home page, I built the footer, and I assisted in lots of the debugging/problem solving that my teammates struggled with. Another large role I played was, I completed the entirety of labs 11 and 13 by myself, along with completing a large chunk of lab 10 in office hours with my TA. Our group had an excellent strategy that one individual should do the labs while everyone else works on the project, which allowed us to be as time efficient as possible!

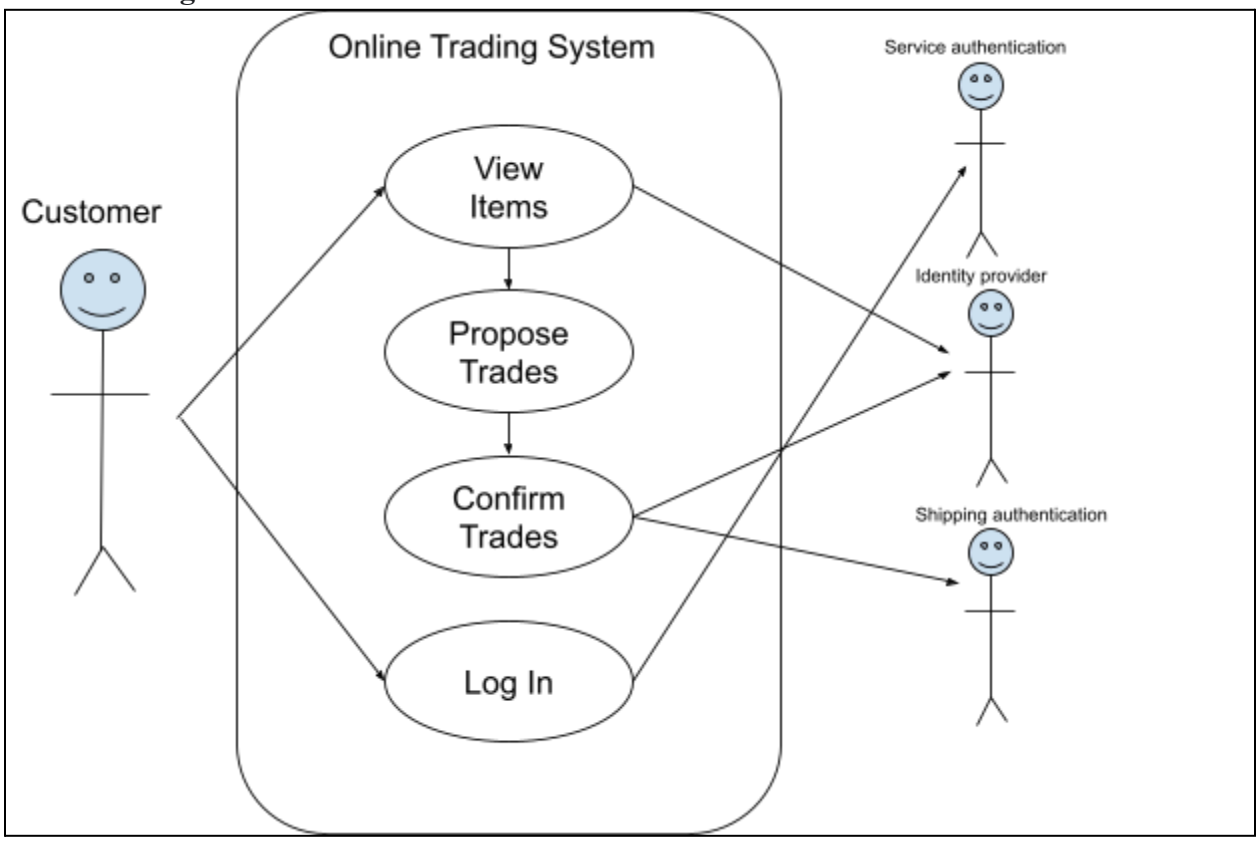
- Asher Siegel: I implemented a lot of the routes and the login/logout/register pages to ensure that the users are given the correct addresses when making an attempt to register/login and when logging out they have the choice to log back in. I also did most of the presentation for our group in which I went over it with them and made sure everyone is doing a slide that they are comfortable with. I also did most of the css for the pages to ensure that there was a consistent look throughout the pages which makes the application presentable and easy to look at. I also helped out with the home page which is the first page the user sees when they log in, hoping they have a good first impression of our website. Overall I believe our team worked great together and we were all compliant in completing work

- Yasir Qasemi: I Implemented the browse page and the post item page. I built the post item and browse item pages from scratch including handlebars, routes in index.js, and the css/html. When someone submits a listing, the routes in index.js take the information, store it in the database, and populate it in browse items. The logic is when you post an item, it will send you to the browse items page and you can see everyone else's items. It also populates your own items in profile. I also did everything with the database and SQL. Since I took databases, I knew how to build it from scratch by using 1NF-3NF. That then turned into a plan with all the tables, indexes, and foreign keys which was pretty hard. I also discussed with the group all the ideas we had for the project and then created the tables. Most of the queries were written by me as well.

- Ivan Gorodinski: I first started this project by implementing the docker-compose.yaml file, so we could use Docker containers to run the website. Similarly, I also helped create the package-json file which held our dependencies and packages that would be used throughout the development process. I work heavily on the index.js file to ensure specific pages such as privacy.hbs, terms.hbs, browse.hbs, and about.hbs would run and render correctly. I also was the first to design pages such as the browse.hbs, privacy, and terms.hbs. The final addition to the index.js file I contributed was making sure that the register and login page used a hash password.

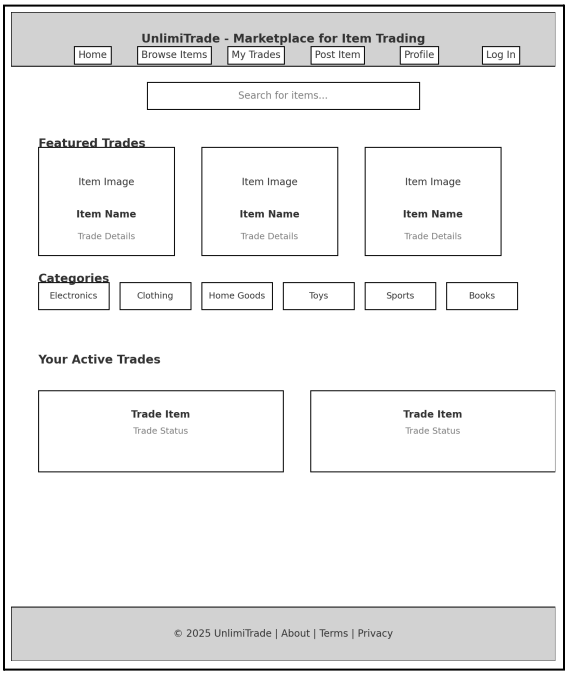
- Connor Short: My tasks for this project included setting up the index.js environment, building the handlebars hierarchy, and implementing the categories feature in browse. I started off by using the index.js layout from previous labs to aid in the creation of the backend for our project. In doing so I was able to get the baseline back end working which allowed us to start adding get and post routines to help traverse and add features to the website. Another one of my large tasks was creating handlebar files for each page and structuring them so that we could use them to their full potential. Handlebars made a big difference in how we structured our code and allowed us to reuse code throughout the website and I was instructed to ensure they all fit together smoothly. Lastly I implemented the categories feature into the browse items page which allows the user to sort trades by the category they were listed with which makes it much easier to find items you might be more interested in.

**Use Case Diagram:**

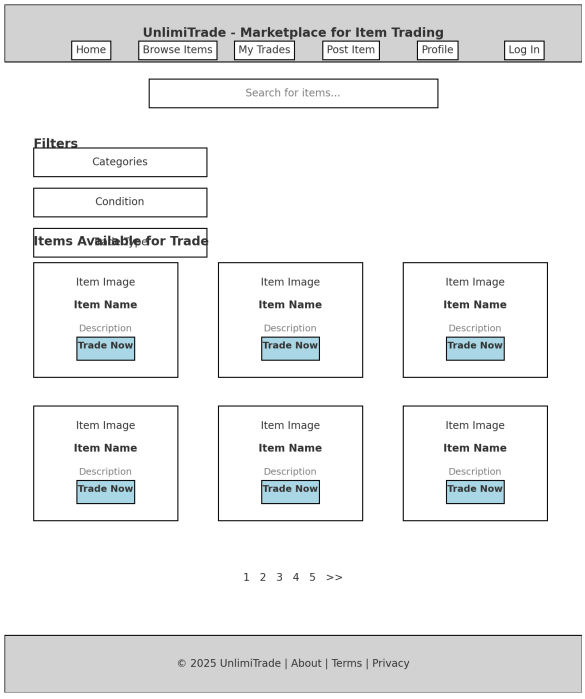


# **Wireframes**:

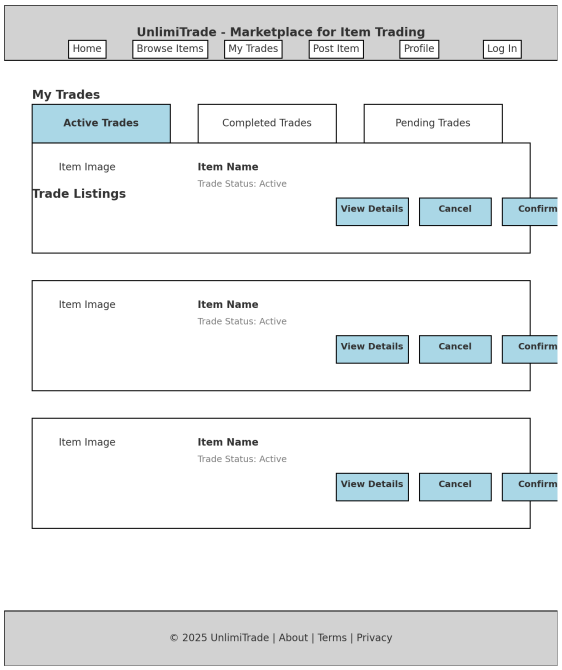
## Home:



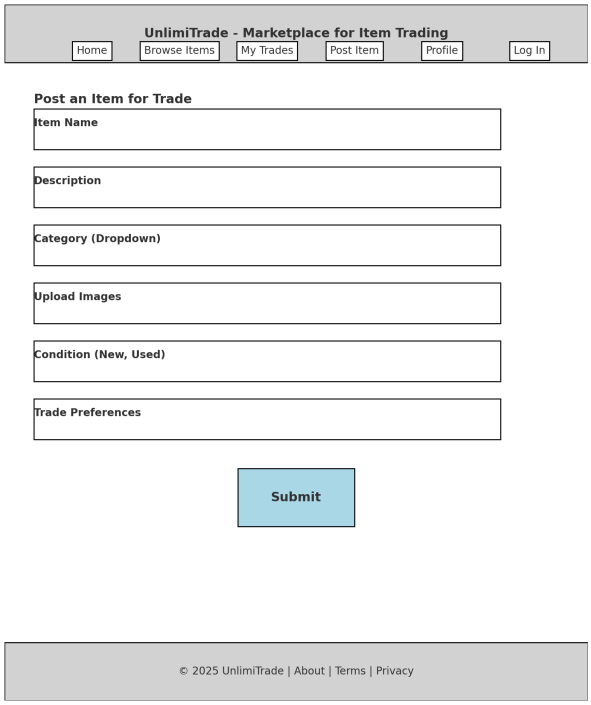

## Browse:



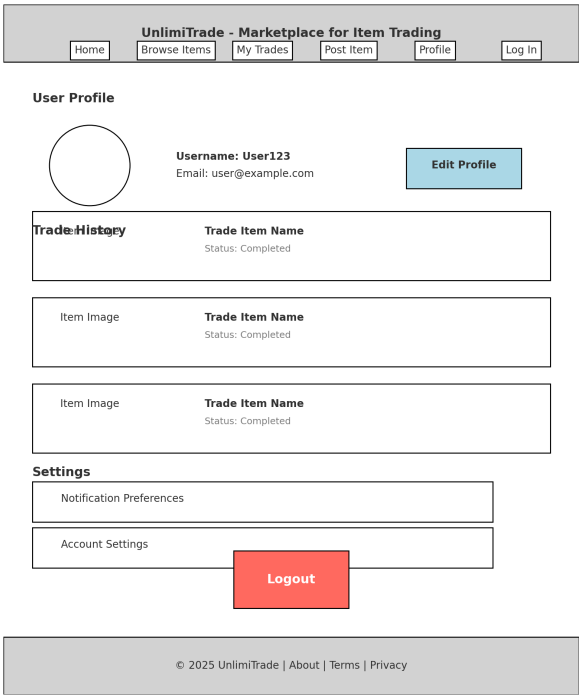
# My Trades:



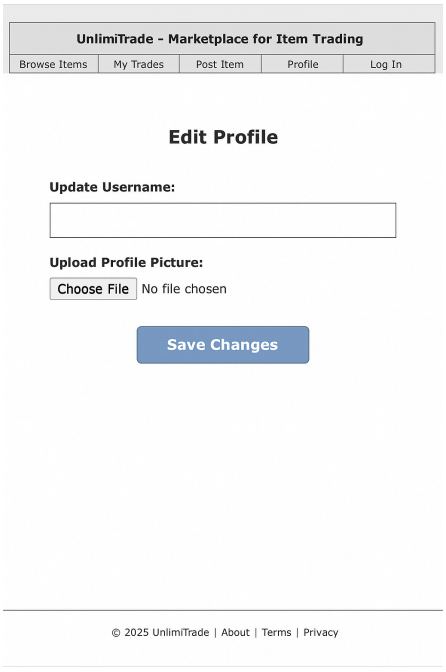

# Post Item:



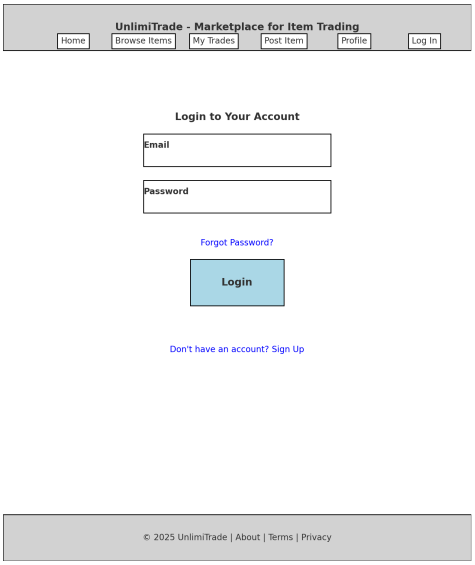
# Profile:




# Edit Profile:



# Login:




# Register:

Logout:


## Test results

Test #1→ **User should be able to post an item to the "explore trades" page.**

The results from this test were highly successful, as this was an integral part of our overall project. For clarity for the reader, we have since renamed the "Explore Trades" page to "Browse Trades". This test is really two tests within one. The first part is, users were successfully able to click on the Post Item Button on the nav bar, and be redirected to the Post Item page. From there, users were able to upload their own images, add their own unique descriptions and names for the items, and actually "post" the item. What "posting" the item means leads us to part 2 of this test. The item would be successfully added to the database and then displayed on the Browse Items page for other users to see, which is the definition of "posting" the items. When other users can visit the Browse Items page and successfully interact with the posted item, (which is what happened) we know this test has been passed!

Test #2 → USER CAN SUCCESSFULLY LOGIN TO THEIR ACCOUNT WHEN THEY ENTER THE RIGHT LOGIN CREDENTIALS:

The results from this test were once again highly successful. This test was slightly more simple than test #1 above, (seeing as test #1 was incredibly complex and time consuming to implement) because we were able to test very quickly and efficiently. In summation, when incorrect credentials were entered, ("incorrect" meaning, let's say the user registers with the username of "max", and tries to login with the username of "macs") the user was unable to enter the website. On the other hand, users were successfully

able to pass the test when they registered their username and password and entered the correct username and password in the login section. Now they can enter the website!

Test #3 → USER CAN CLICK ON AN ITEM FROM WITHIN THE "EXPLORE TRADES" PAGE TO VIEW / EXPAND MORE INFORMATION ABOUT THE ITEM.

Lastly, this test was passed successfully too. The results of this test were as follows: users were successfully able to view posted items on the Browse Items page. From there, users were then able to click on / select any given item and propose a trade for the item, with more information surrounding the item becoming available. This is exactly what we were hoping would happen, so it was fun to see this test be passed with flying colors. Another function we implemented was the ability to add a review to a user's profile, and this function was made available through the user's item tile on the Browse Items page. This was also great, (although not included in the original Lab 11 test writeup) because it allows for more user to user interaction and better overall security within the website. We were proud to pass all these tests, because we felt confident that our user interface looked really good!

**Deployment:**Link To Website