

Final Project

Car Hub

Group 4

(Thanh Tran, Zohair Khan, Cory Keuning, Yonatan Tafesse, Yod Xiong)

I. Introduction

Welcome to CarHub. It can be daunting when trying to sell a vehicle, especially if one does not want to trade it in or sell it to a used car lot for less than its real value. Searching for a vehicle, driving from used car lot to used car lot, and dealing with all sorts of salespeople can be equally taxing. CarHub attempts to solve this problem with an easy-to-use application in which sellers can list their vehicles and buyers can browse those listings. The listings include relevant information about the vehicle and the seller's contact information, so they can then get in contact with each other directly, and the sale of the vehicle can be conducted on their own terms. Sellers can get their vehicles in front of potential buyers, and buyers can avoid the used car lot grind.

II. Feature overview

Functional Features:

- Sign up / log in.
 - Vehicle sellers can create a profile on CarHub with a username, password, and contact information through the sign-up page.
 - Sellers can then login with their username and password through the login page.
- Create Listings.
 - Sellers can create a listing with a picture and information about their vehicle through the Create Listing page.

- Their contact information is automatically included in the listing.
- Modify Listings.
 - Sellers can modify listings they have already made with updated information through the edit listing page.
- Search Listings.
 - Buyers can search through the listings on the listings page.
 - Buyers can also filter listings by category through the listings page.
- View Listings.
 - Buyers can click on a listing and view the vehicle information and the seller's contact information through the listings page.
- Edit Profile Information
 - Sellers can edit their profile and contact information through the edit user page.

Non-Functional Features:

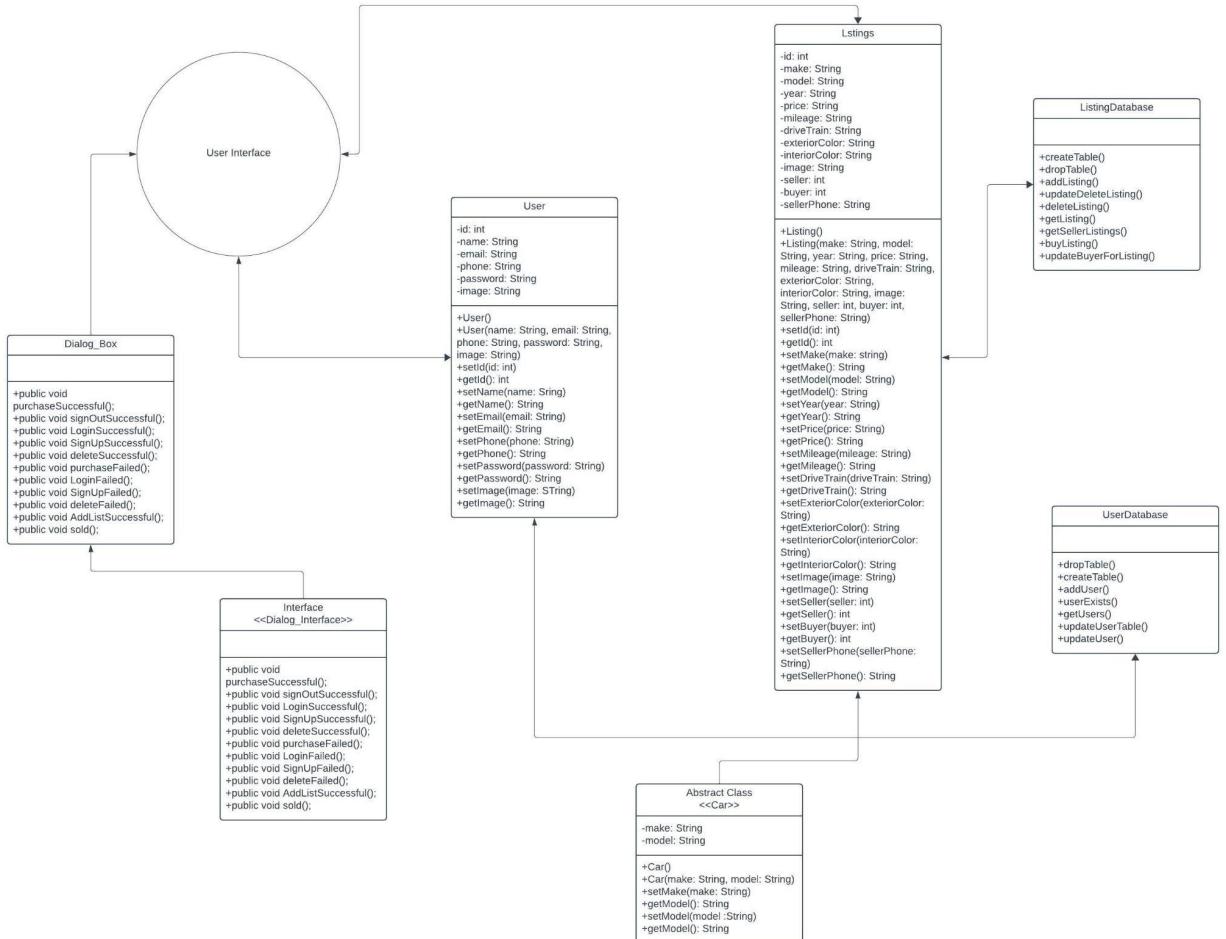
- Data Persistence.
 - This is achieved by storing all the data in CarHub in a database the application interacts with.
- Simplicity.
 - This is achieved by designing the user interface in a simple manner. There is not too much going on on each screen, and the purpose of each button is clear to the user.
- Maintainability

- The modular design of the application makes maintenance easier. One can make changes in the classes instead of having to deal with the UI interacting directly with the database.
- Availability.
 - The application, if it were released in the real world, would be accessible to users 24/7.

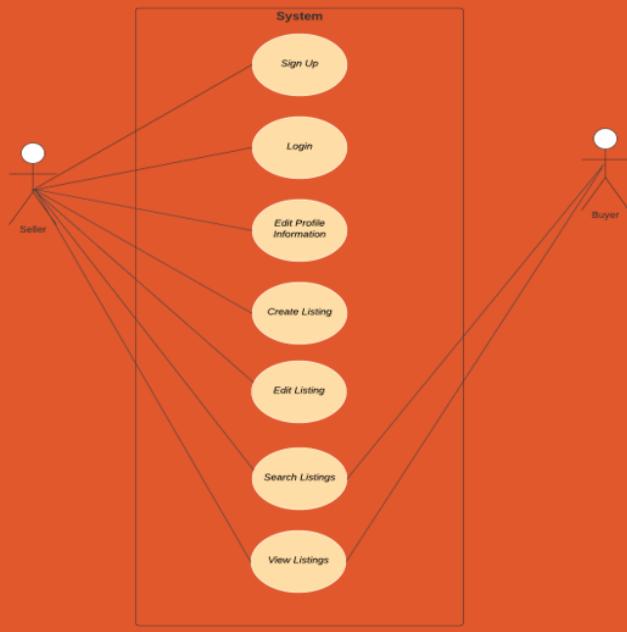
III. Software Architecture and Design

The application is implemented internally through the combination of JavaFX, Java classes, and a sqlite database using the model view controller architecture pattern. When a user creates a profile, creates a listing, or edits their information, the data is transferred through the implemented functionality of JavaFX and stored in the database. When a user views the main listings page, searches the listings, or filters the listings by keyword, the data is pulled from the database, listing objects are created in the Java classes with the data, and the listings are displayed on the user interface through the JavaFX.

Class Diagram



Use Case Diagram



IV. User Interface Design

Previous Version

Login



CARHUB

Search by keyword

Browse

Sell



Login

Search by keyword

Category: ALL

Location: Minneapolis

Price:

Between \$50 and \$75

Posted in the last 9 days

Sort by: Price Low-High

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

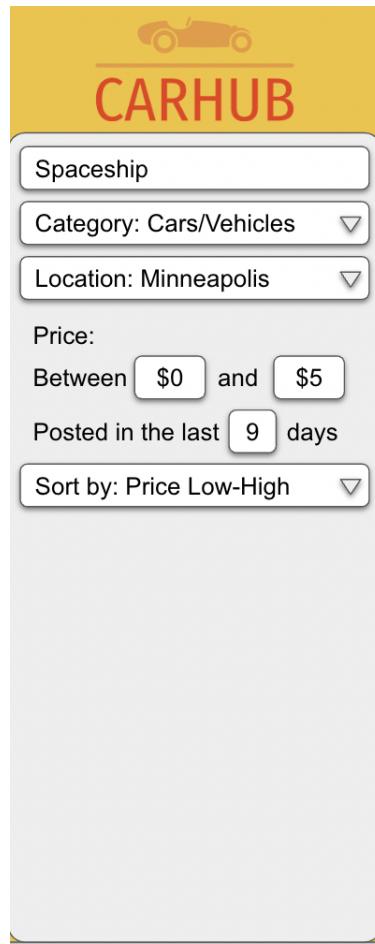
Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

Listing Image

Listing Title Goes Here \$55
Posted 8/28/2023 8:40 PM

◀ 1 - 10 of 100 ▶

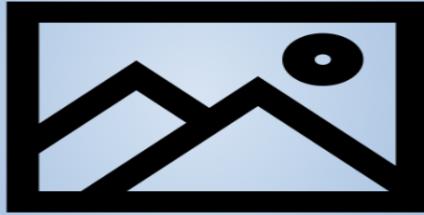


No results found!

 Email Password**Sign In**[Forgot password?](#) First name Last name Email New password**Sign up**

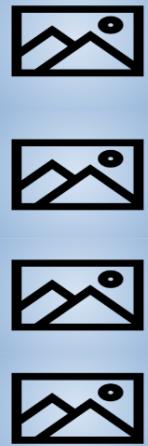
[Login](#)

Item Name: Preferably detailed



Item description: Considering being as detailed and descriptive as possible. The more the details the higher the chance of purchase.

\$27.00



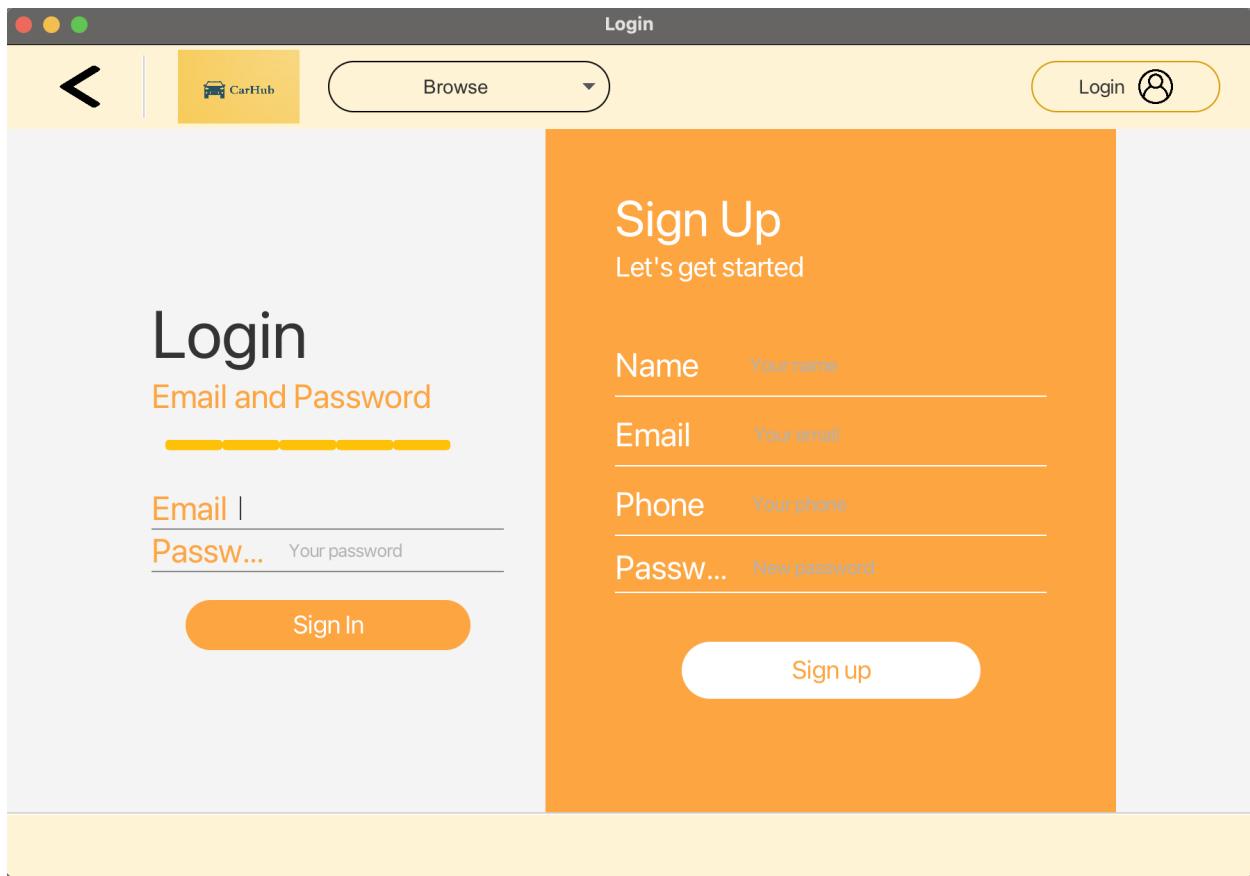
Sold By:
Anonymous

Contact Seller

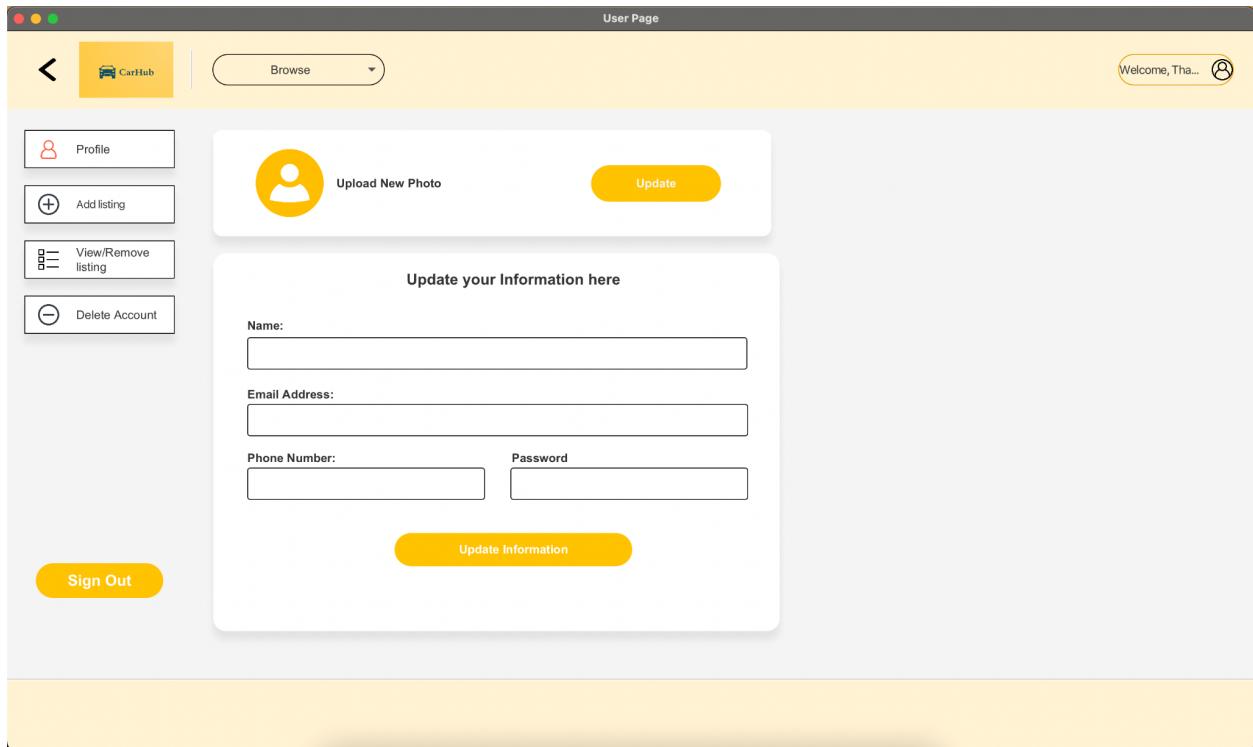
Add to cart

Final Version

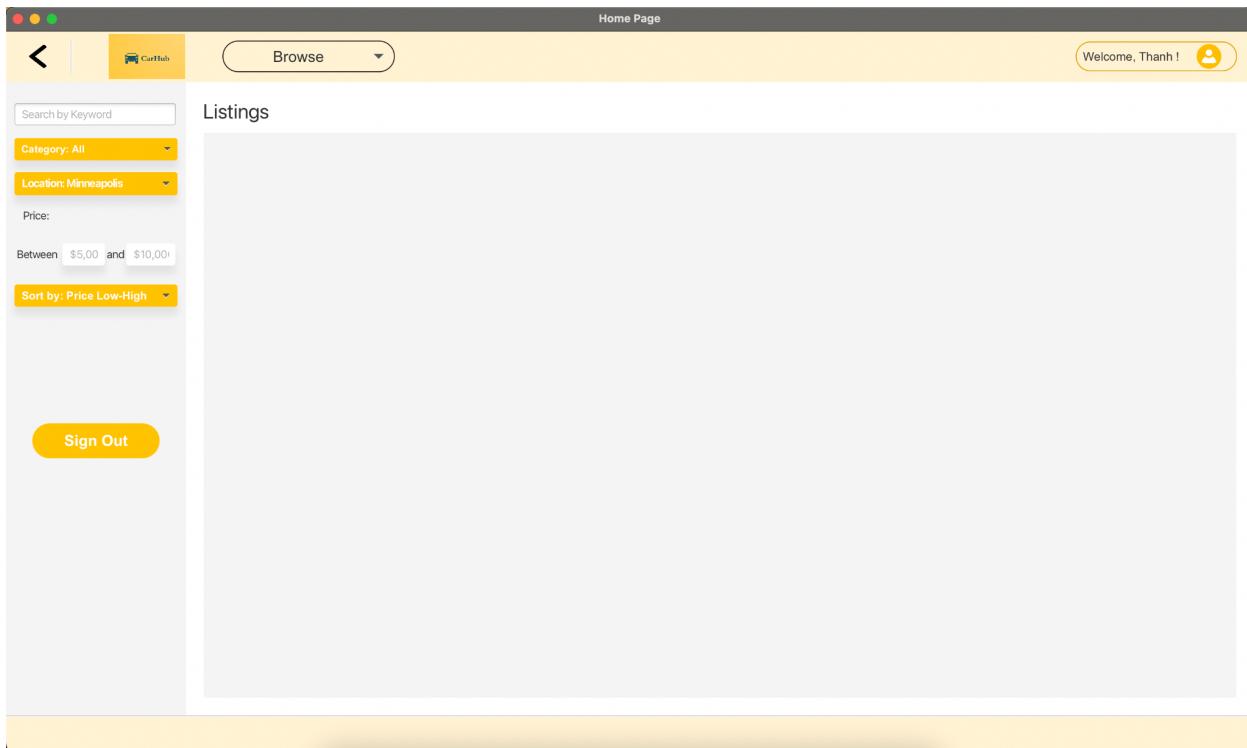
Login Page



User Page



Listing Page



View Listing Page

The screenshot shows the CarHub application interface. At the top, there is a dark header bar with three small icons (red, yellow, green) on the left and the text "View/Remove Listing" on the right. Below the header is a yellow navigation bar containing the "CarHub" logo (a blue car icon), a "Browse" dropdown menu, and a "Welcome, Th..." message with a user profile icon.

The main content area has a light gray background. On the left side, there is a sidebar with the following options:

- Profile** (Icon: person)
- Add Listing** (Icon: plus)
- View/Remove Listing** (Icon: document with lines)
- Delete Account** (Icon: minus)

Below the sidebar, the text "Revenue" is displayed in bold black font, followed by the value "0.0" in green. At the bottom of the sidebar is an orange "Sign Out" button.

The main content area features a large, empty white rectangular box labeled "My Listings".

Add Listing Page

The screenshot shows the 'Add Listing Page' of the CarHub application. The top navigation bar is dark grey with the title 'Add Listing Page'. On the left, there's a yellow sidebar containing links: 'CarHub' (with a car icon), 'Profile' (with a person icon), 'Add Listing' (with a plus icon), 'View/Remove Listing' (with a list icon), and 'Delete Account' (with a minus icon). In the center, there's a large white form area. At the top of this form is a 'Browse' dropdown menu and a 'Welcome, Tha...' message with a user icon. Below these are two buttons: 'Upload a photo of your vehicle' with a camera icon and a yellow 'Upload' button. The main form area contains six dropdown menus for vehicle specifications: 'Make', 'Model', 'Year', 'Price', 'Mileage', 'Drive Terrain', 'Interior Color', and 'Exterior Color'. At the bottom of the form is a large yellow 'Add Listing' button.

CarHub

Welcome, Tha...

Profile

Add Listing

View/Remove Listing

Delete Account

Browse

Upload a photo of your vehicle

Upload

Make

Model

Year

Price

Mileage

Drive Terrain

Interior Color

Exterior Color

Add Listing

Sign Out

Vehicle Profile Page

\$52,995
2021
Audi
A4
All Wheel

Mileage
17,000

Year
2021

Exterior Color
Black

Interior Color
White

Drive Terrain
4-wheel drive

Contact Seller (651)555-5555

Sign Out **Delete Listing**

V. Development process

The team primarily used a Discord channel for communication and to upload and share files with each other. We originally had a github repo set up for us to work on the same codebase, but that was abandoned rather quickly when the functionality of the app started. The group started working on their own separate projects at that point. When

members started struggling, they would have other members explain what they had done with their version.

We originally planned on building a marketplace application in which a user could list any items, but after milestone 1, we decided to narrow our idea down to a used vehicle sales app to make the project more manageable for us. This is when we came up with almost all the features we currently have, and those feature ideas we came up with together in class. Our strategy was to implement the minimum number of features while maintaining a usable application to keep it simple. We did not have an extensive testing process for our features. We were really just trying to get all of our features to work. When we got the feature to work was also when we decided that our feature was complete.

VI. Summary

The beginning of the project went well for us. Each of us naturally tackled an aspect of each milestone that we felt comfortable doing, and we scored very well on the first three milestones. Unfortunately, what didn't go well is a longer list than what did go well.

As we said above, the github repo was abandoned after milestone three. Everyone started working on their own projects, which led to a way bigger workload. Everyone was also getting stuck on different things, and we needed to continuously upload files to Discord. This also made it nearly impossible for us to delegate different tasks and actually work together on what we needed to implement in our project. Procrastination was also a problem with our group. With such large chunks of time between some of the milestones, it was easy for us to put off the work we needed to do until the last minute, which really hurt us. It led to us not getting our work done in time when we ran into problems, especially with all the different codebases.

What we would have done differently, if we could start again, is try hard to address the issues stated above. All of us working on the same codebase would have made delegating tasks much easier and would have kept all of us on the same page throughout the project. To address the time management problem, instead of leaving everything to the last minute, we would work on the project using all the time we had available. This way, we could spread out the work and do bits at a time. This would have also given us more time to work on problems when we encountered them.

Another thing that we might have done differently is to take a little more time to come up with a simpler idea for our project. It seems we may have bit off a little more than we could chew with our CarHub idea. Especially since only some of the group members had experience with SQL and databases and none of us had any experience with JavaFX. This made the project very difficult for all of us to stay on the same page and get the project done.

From this experience, we learned that planning is crucial when it comes to software development; planning time management, how to share code, and delegating tasks. With better planning comes easier execution.