On the Web Listing (OWL) Marketplace
Final Project Report

Group 7: Philip Canzian, Jose Casiano, and Forrest Moulin

College of Information Sciences and Technology, Pennsylvania State University

IST 440W: IST Integration and Problem Solving

Kaamran Raahemifar, PhD, PEng

August 6, 2022

Table of Contents

Problem statement

The online shopping trend has been on the rise throughout the years. Buyers and sellers alike are looking for common interfaces to interact within the real-estate market. Simplicity, precision and speed are attributes that customers tend to seek when buying a property. The less time to find exactly what a customer is looking for, the greater the satisfaction. Due to the pandemic, lifestyles around online shopping has changed drastically as there has been a societal movement towards becoming more virtual. Essentially, customers are looking to purchase new homes while minimizing the time spent outside of their own homes. Due to the pandemic, this long-term trend of online shopping has increased exponentially, thus creating an opening for the virtual property marketplace. 97% of homebuyers use the internet to research properties when looking for new homes (Investopedia 2021).

Objectives

The original team's focus of development is to prioritize the customer's utility of the web application. The purpose of this application is to develop and implement a simple search system for listed properties primarily by location and pricing. Primarily, search by location entails typing and entering zip codes, addresses, and states. The secondary search option is by pricing. Features of searching by price include a range setting, where a user can input a minimum and maximum price range. If the user has an account, they will be prompted with an option to save and name their searches.

With enough allotted time, the team's secondary objective is to focus on the listing of the properties, from the view of the homeowner (the customers). This allows the users to have the ability to upload photos, videos, and demos when listing the property onto the webpage. However, if it is not feasible to complete before the deadline, the primary focus will be to develop the customer's search feature.

Overall, the goal of this project is to take the source code provided and make it reusable. In this case, utilizing Java, HTML, CSS, and JavaScript source code to create a web application.

End-user Needs

The demographic of users targeted:

- 97% of homebuyers use the internet to research properties when looking for new homes (Investopedia 2021).
- 60% never saw the home until after the purchase of the home in 2020 (Investopedia 2021).

Constraints

The main constraint during project development and execution was time. Due to the structure of the summer semester, there is roughly a month less than a standard semester (fall and spring), which prevented our team from reaching the final goals and objectives. However, our team was able to nearly complete most of the project as intended.

Literature Review

Assessment of Available Solutions and Techniques

- *Primary: Search by (customer view)*
    - First Objective: Locations by Zip Codes, Addresses, or even States
    - Second Objective: Prices/pricing
    - (Key words – Future/Additional Work)
        - Range settings (min and max)
    - Option to save and name searches
        - Prompt with optional decisions to save or not

- *Secondary: Focus on Listing Properties if time allows (homeowner/real estate agent view)*
    - Photo uploads of houses
        - Possibly demo if enough time

- *Goals:*
    - Make it reusable, Java Spring
    - HTML, CSS, JavaScript, Thymeleaf for front end
    - End goal: Acquire domain + server with Azure App Service

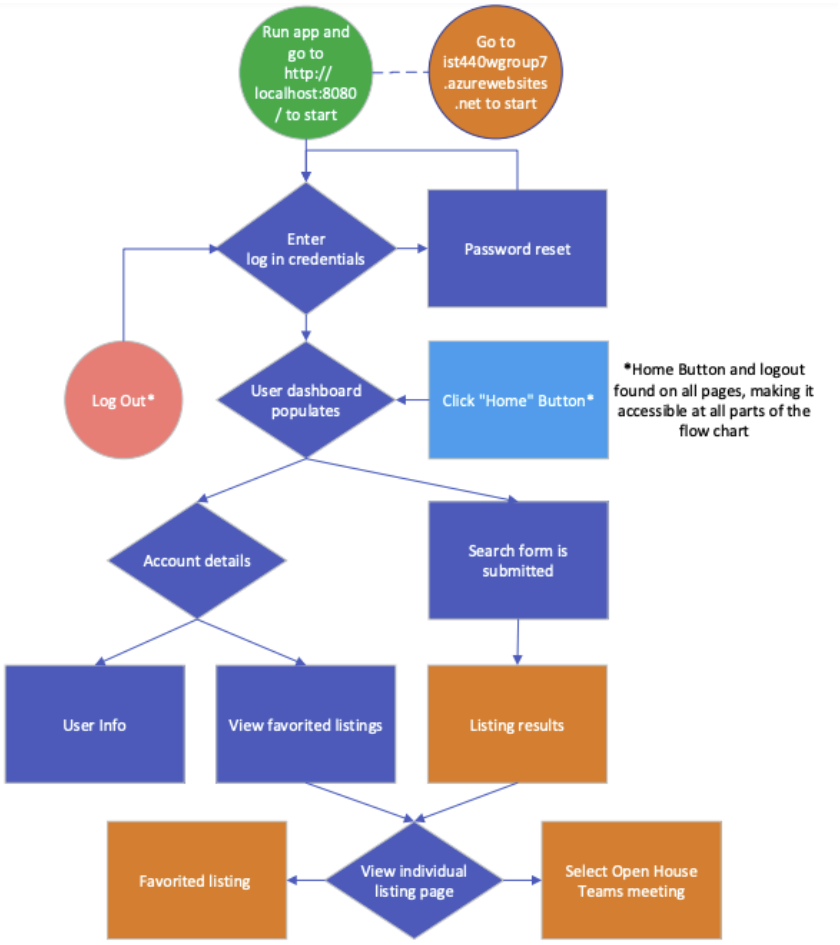
- *Additional Solutions/Future Work*
    - Webpage updates and design enhancements (front end)
    - Virtual Room Tour with Microsoft Teams integration
    - Real Estate chat bot feature
    - Save a listing to your account
    - Mobile application development for iOS/Android

Pros and Cons of Each Approach or Technique

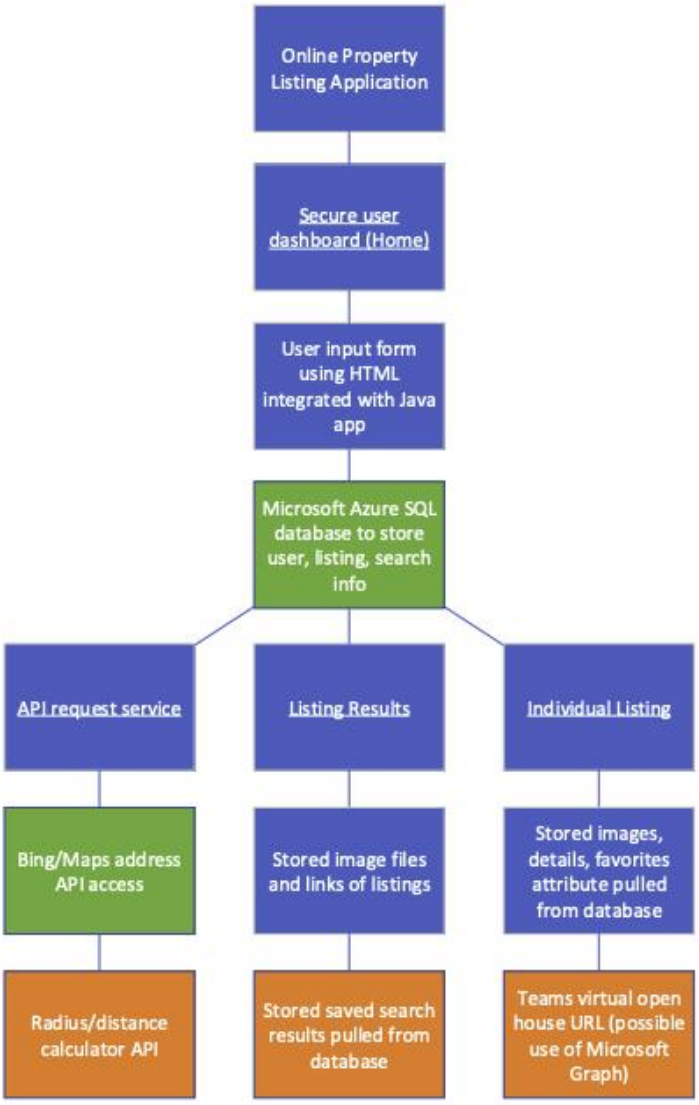
- Pros
    - Webpage easier to develop
    - Mobile apps are more convenient for customers
    - Easier to format/develop in PHP
- Cons
    - Mobile app would take longer to develop
    - Virtual Room Tour --> API Registrations (potential roadblock)
    - Overall user-interface
    - Group was not too familiar with PHP going into the project development

System Design

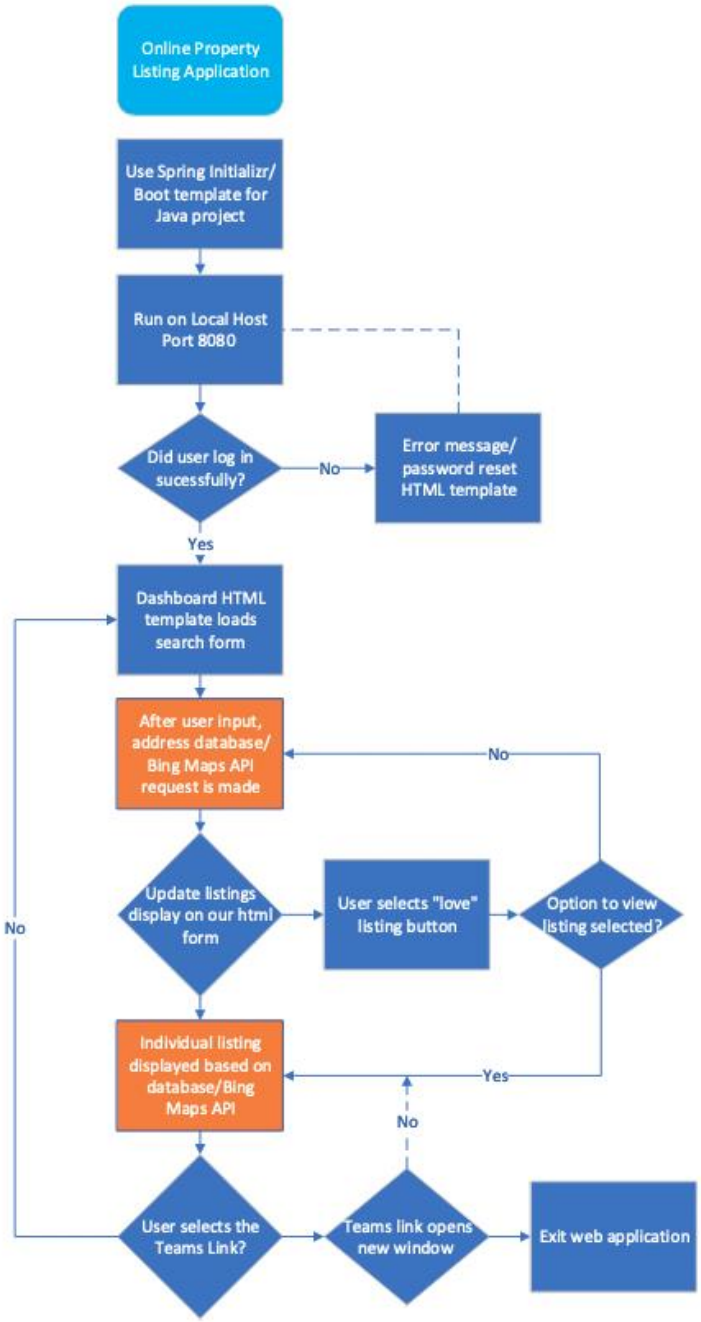
Operational Flow Chart



Functional Decomposition



Technical Design Flow

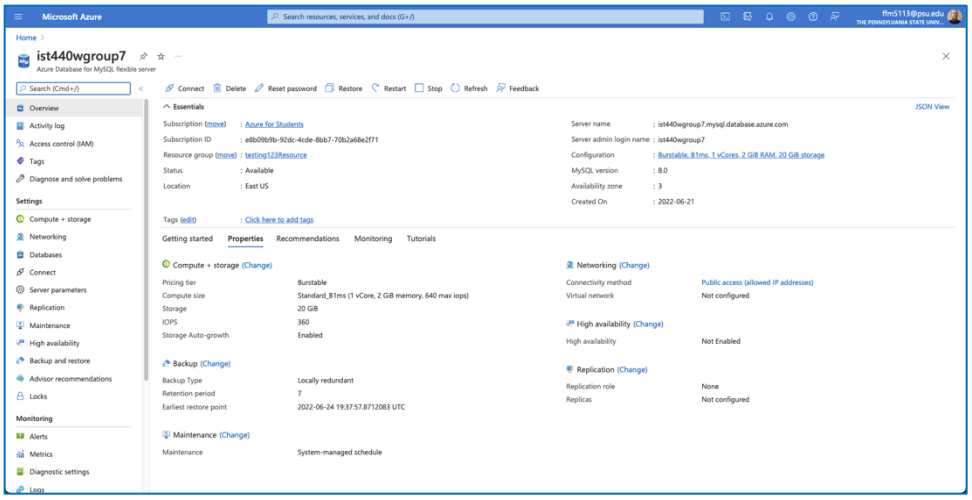


Azure MySQL Database Setup

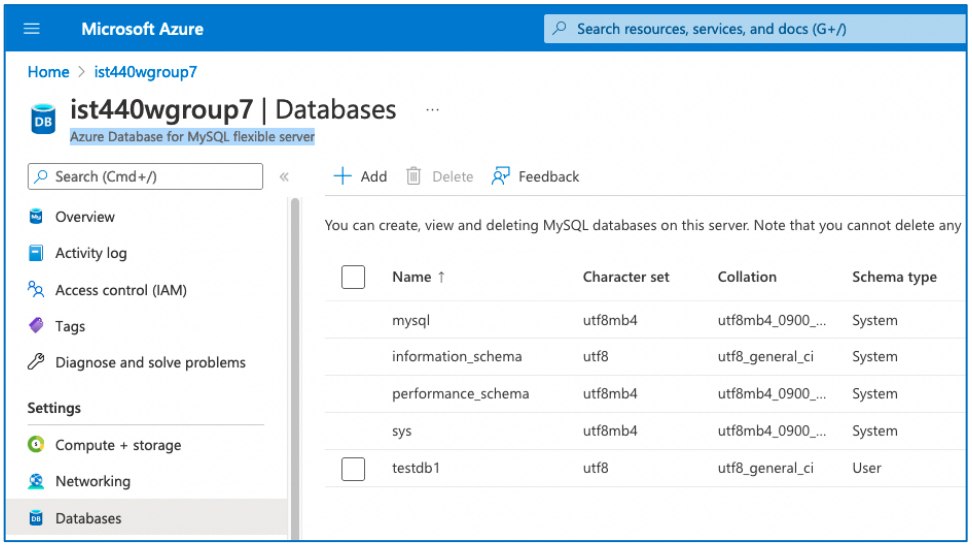
Creating a cloud-based database using Azure allowed the development team to collaborate on a centralized GitHub project that was integrated with a single database. This approach allowed the team to access the project database from anywhere with a secure internet connection. By implementing this type of technology, the group experienced a few benefits:

- Low costs and little server maintenance
- Access to real-time application data
- Potential to integrate cloud database with Azure App Service
- Database can be altered via Azure Cloud Shell

1. Create an Azure Database for MySQL flexible server



2. Create a database instance



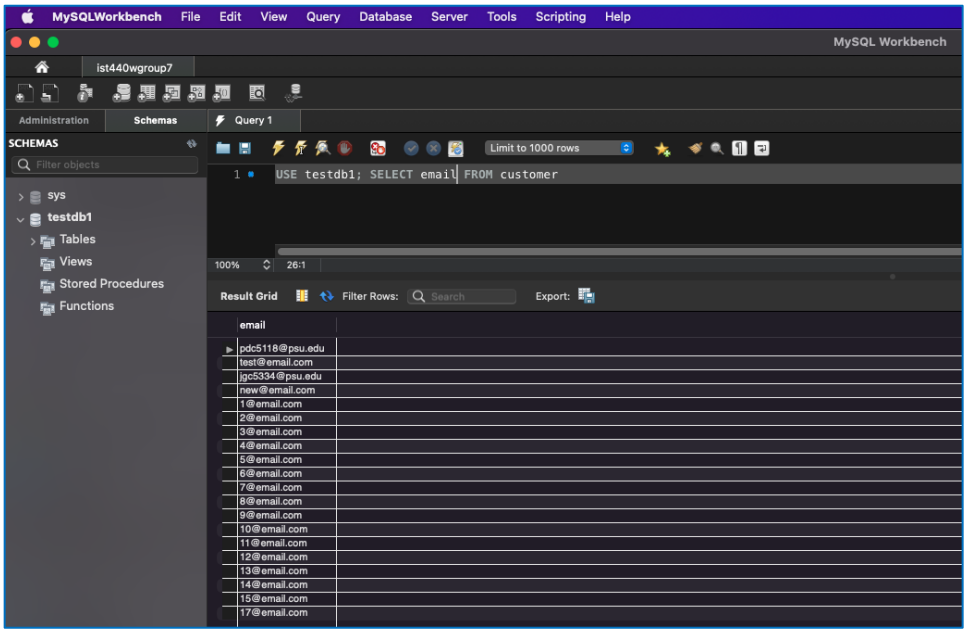
3. Connecting to the database with MySQL Workbench (Azure Portal)

MySQL Workbench

To connect with MySQL workbench client, follow the steps below.

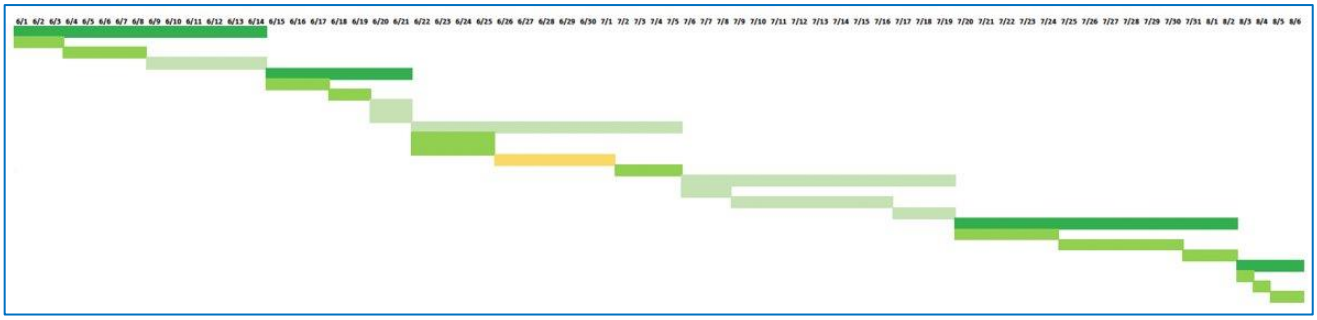
1. Click the + symbol in the **MySQL Connections** tab to add a new connection.
2. Enter a name for the connection in the **Connection name** field.
3. Select **Standard (TCP/IP)** as the Connection Type.
4. Enter **ist440wgroup7.mysql.database.azure.com** in hostname field.
5. Enter **ist440wgroup7** as username and then enter your **Password**.
6. Go to the **SSL tab** and update the Use SSL field to Require.
7. In the **SSL CA File** field, enter the file location of the **DigiCertGlobalRootCA.crt.pem** file.
8. Click **Test Connection** to test the connection.
9. If the connection is successful, click **OK** to save the connection.

4. Test the MySQL Workbench database connection



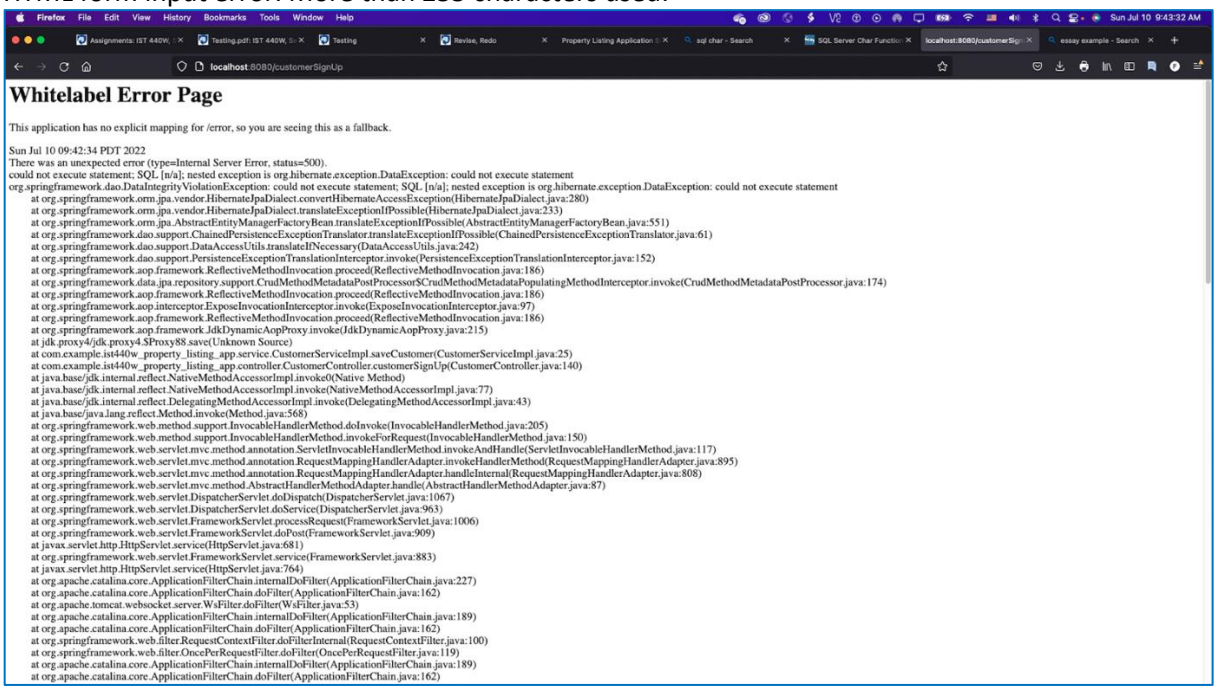
Gantt Chart

Task ID	Task Name	Start Date	End Date	Days	Assigned to	Comments
1	Back end	1-Jun	14-Jun	14		
1.1	IDE/Dev Kit/Repo setup	1-Jun	5-Jun	3	All	Completed on time
1.2	Object creation/design	6-Jun	10-Jun	5	FM, PC	Completed on time
1.3	Writing model classes	11-Jun	14-Jun	6	All	Majority completed
2	Database	15-Jun	21-Jun	7		
2.1	Create db using SQL	15-Jun	17-Jun	3	All	Completed on time
2.2	Integrate db with back end	18-Jun	19-Jun	2	FM, JC	Completed on time
2.3	Db integration testing/cleanup	20-Jun	21-Jun	2	PC, JC	Completed on time, minor future maintenance
3	APIs	22-Jun	5-Jul	14		
3.1	Research/review documentation	22-Jun	25-Jun	4	PC, JC	Partially completed Bing Maps API research
3.2	Integrate with back end	26-Jun	1-Jul	6	FM, JC	Partially completed
3.3	API integration testing	2-Jul	5-Jul	4	FM, JC	Tested integration with HTML form
4	Front end	6-Jul	19-Jul	14		
4.1	Wireframe design/enhancements	6-Jul	8-Jul	3	All	Partially completed
4.2	HTML for GUIs (with Thymeleaf)	9-Jul	16-Jul	8	FM, JC	Partially completed
4.3	Final updates, testing, cleanup	17-Jul	19-Jul	3	All	Partially completed
5	Working app/future enhancements	20-Jul	2-Aug	14		
5.1	Final webpage testing	20-Jul	24-Jul	5	All	Completed
5.2	Future enhancement documentation	25-Jul	30-Jul	6	All	Completed
5.3	Project wrap up	31-Jul	2-Aug	3	All	Completed
6	Class deliverables	3-Aug	6-Aug	4		
6.1	Demo/PPT Report	3-Aug	3-Aug	1	All	Completed on time
6.2	Record Presentation	4-Aug	4-Aug	1	All	Completed on time
6.3	Written Report	5-Aug	6-Aug	2	All	Completed on time
Total	-	1-Jun	6-Aug	67		

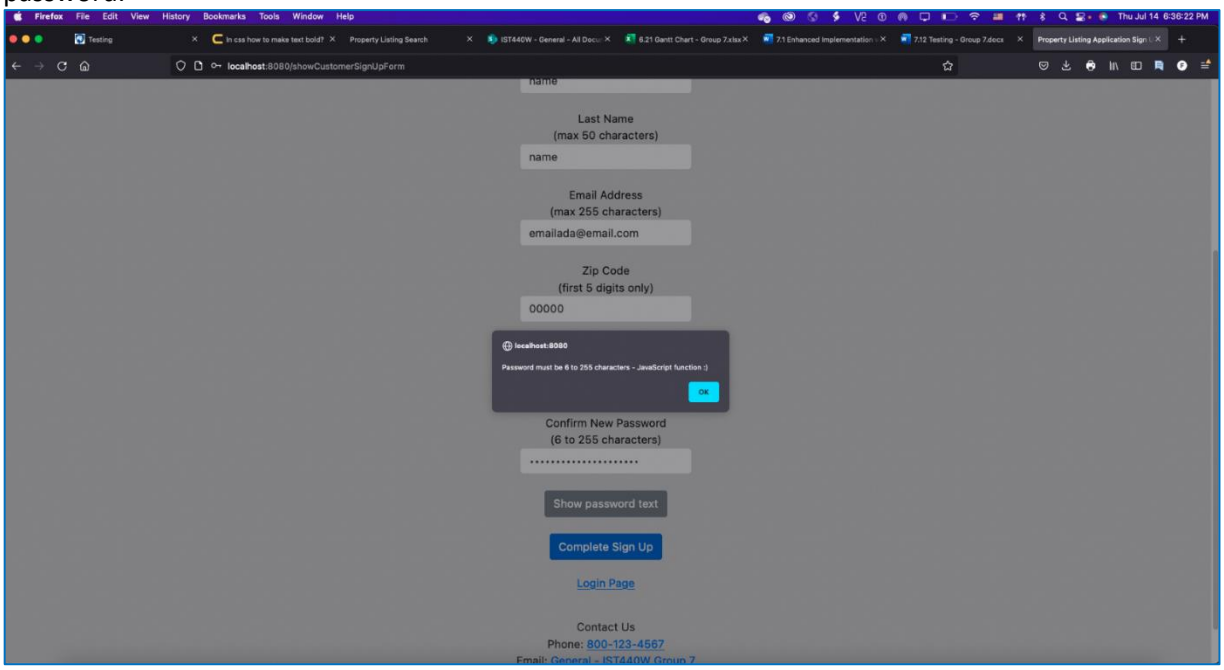


Testing and Verification

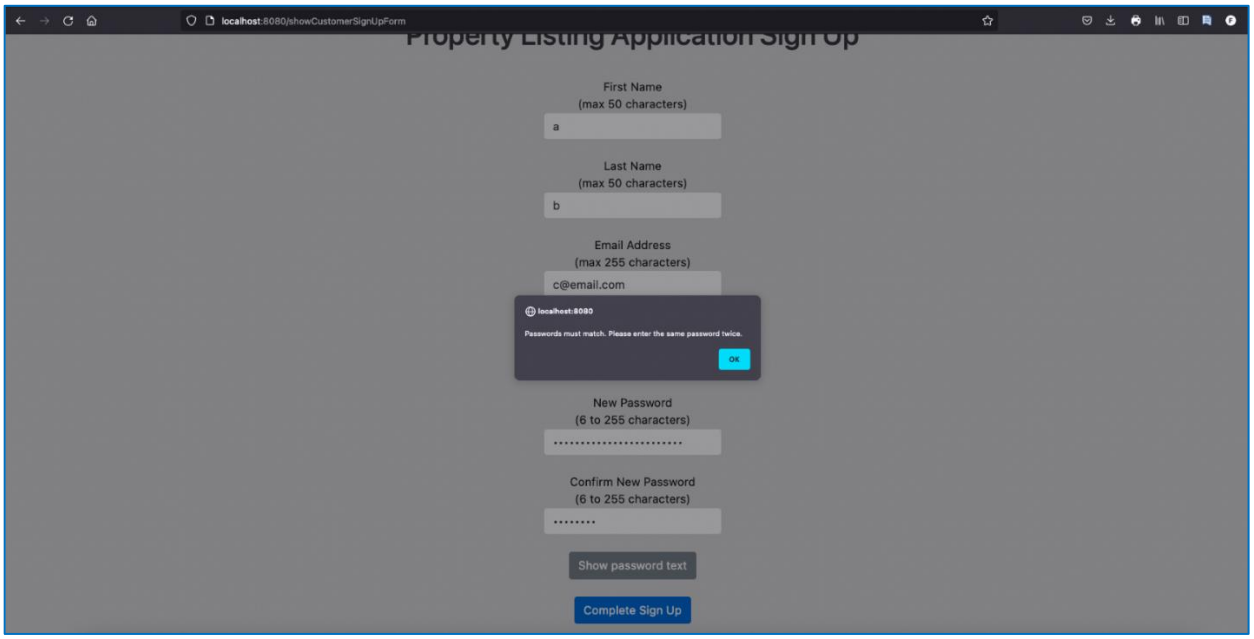
HTML form input error. More than 255 characters used:



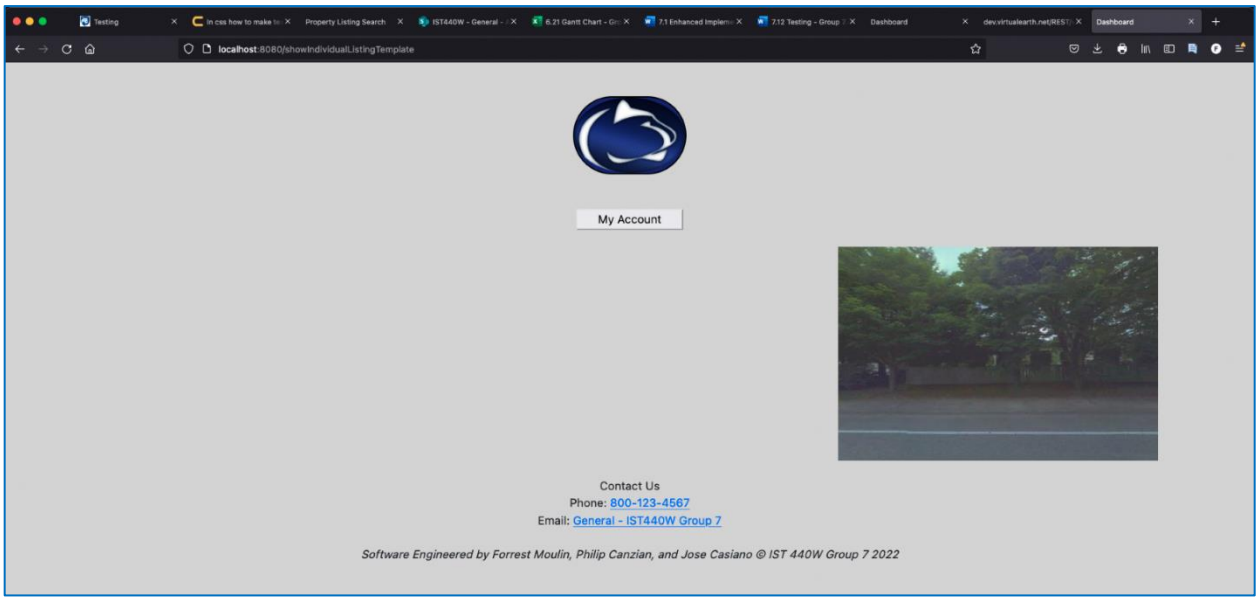
A JavaScript function was created to check the HTML text for 6 to 255 characters in the signup form. Below is the result when more than 255 characters are entered for a password:



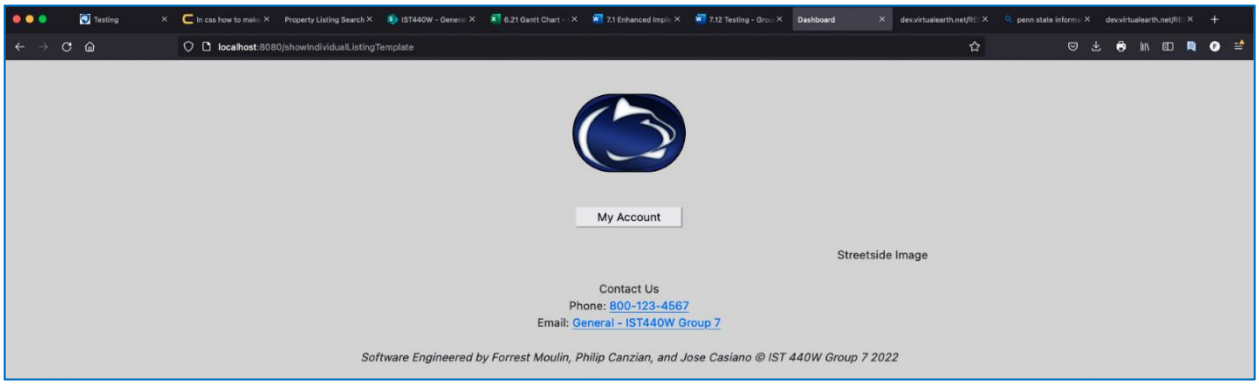
Second Attempt. Using wrong passwords during sign up:



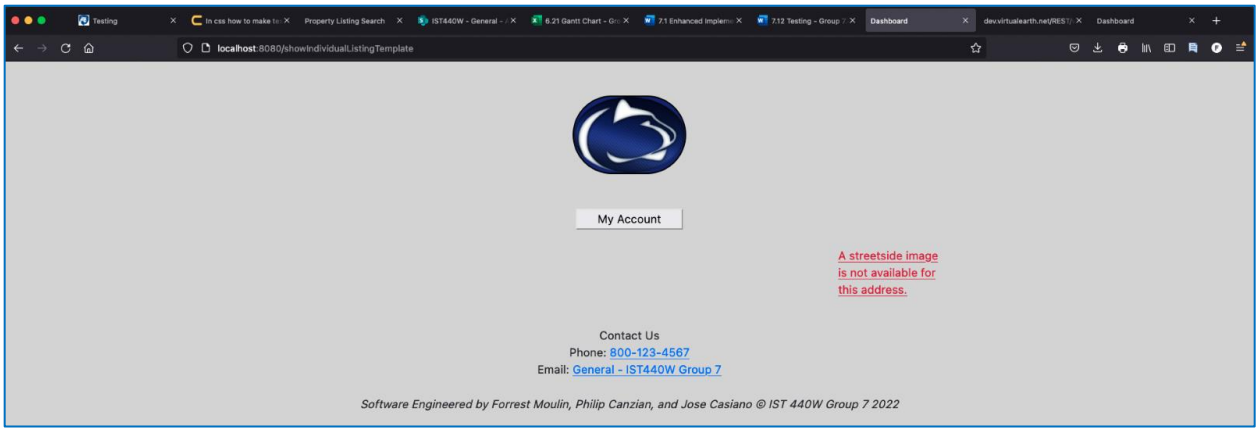
Example of Individual Listing successful Bing Maps image prototype:



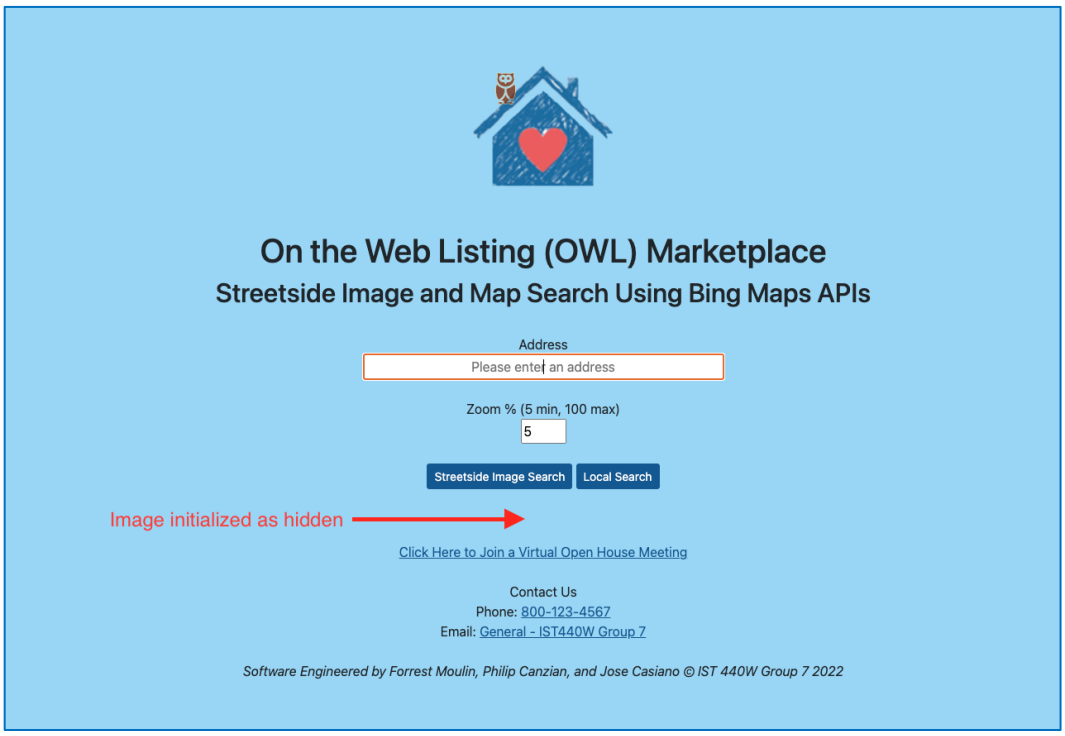
Example of Individual Listing unsuccessful Bing Maps image. In this case, there is no available streetside image, but only the alt text is displayed:



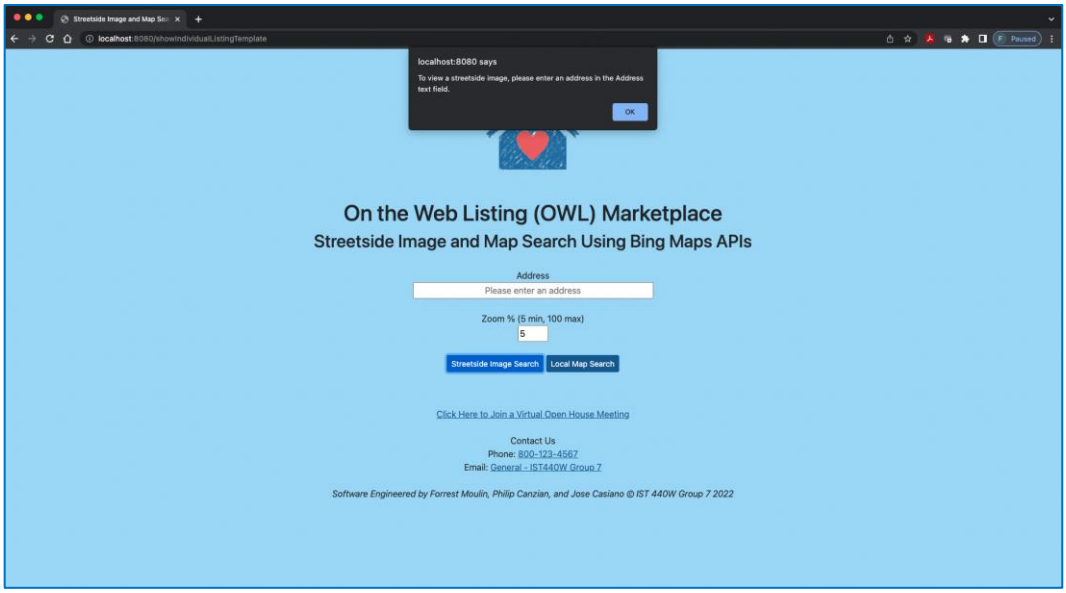
Example of Individual Listing unsuccessful Bing Maps image with alert message to inform user that an image is not available for the address:



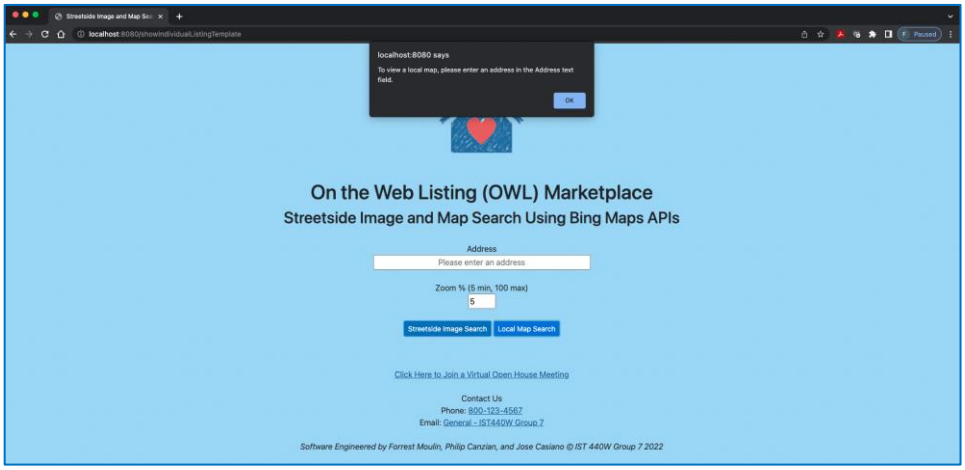
Search image is initially hidden to avoid confusion when seeing alt text:



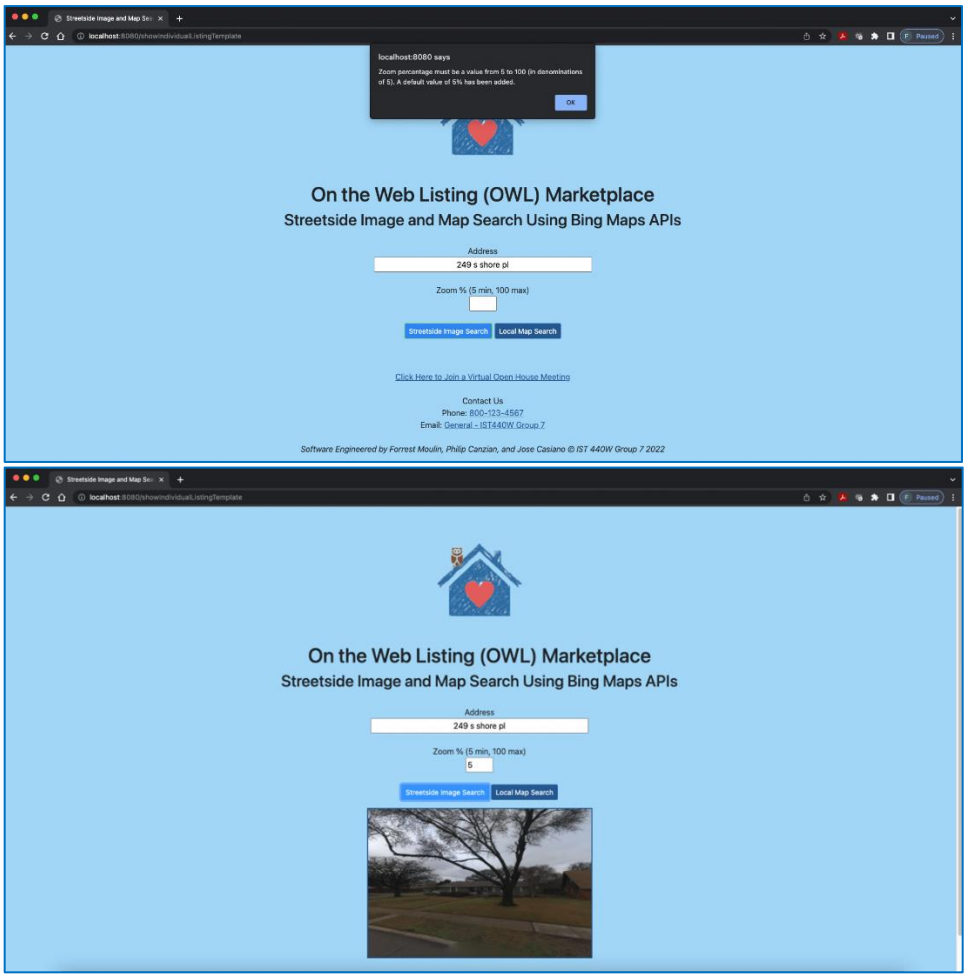
Streetside image search button checks for text in the Address text field:



Local map search button checks for text in the Address text field:



If a zoom level is not entered, an alert is sent to the user and a default value is added:



Git Hub Repository

Please view our GitHub repository for more information or to download and test the application. The repository will be made public for at least 30 days from the project submission date (8/6/22):
https://github.com/ffm5113/ist440w_property_listing_app

Presentation Video Link

https://pennstateoffice365.sharepoint.com/:v:/s/IST440W699/ERPhcHn469pItNRnWrtsQcUBGc560YzO0Ns28D8Q_vnDhA?e=itFmvS

Conclusion

- Research Example Projects
- Establishing Groundwork
- Leverage Project Management Methods
- Implementation and Integration Process
- Final Project Status and Results

Lesson Learned

- Setting up GitHub repository
- Implementing a cloud-based database
- Virtual project collaboration
- Managing time zone differences
- Flexibility of platform and API utilization

Challenges

- Our group had developed a solidified plan with an end goal and final directive as seen in our Gantt and PERT Charts
- Due to time constraints, our group was not able to reach this end goal
- Specifically, Use Case #2 was the critical path
    - Our time constraints prevented us from reaching this stage of the project, resulting in becoming future work

Future Work

- Develop a page that shows all properties after searching an address
    - **Goal/Next Step:** Search à Pulls up list of properties à Select an individual listings page à Individual listing page opens
- Set up a Teams/Meeting *API* (Currently can put our personal meeting link on the listing page – prototype)
- Azure App Service implementation
- Additional use cases for listing entities (real estate agencies, personal sellers, etc.)
- UI enhancements and mobile compatibility

References

MySQL Documentation. MySQL. (n.d.). Retrieved June 24, 2022, from https://dev.mysql.com/doc/

Nkramer. (2022, March 28). Microsoft Teams API Overview - Microsoft Graph. Microsoft Graph | Microsoft Docs. Retrieved June 20, 2022, from https://docs.microsoft.com/en-us/graph/teams-concept-overview

Rbrundritt. (2018, December 14). Locations API - Bing Maps. Locations API - Bing Maps | Microsoft Docs. Retrieved June 15, 2022, from https://docs.microsoft.com/en-us/bingmaps/rest-services/locations/

Segal, T. (2022, March 1). How COVID-19 changed consumer shopping behavior. Investopedia. Retrieved May 28, 2022, from https://www.investopedia.com/how-we-shop-now-5184434

Spring Data JPA. Spring. (n.d.). Retrieved June 24, 2022, from https://spring.io/projects/spring-data-jpa

Thymeleaf. (n.d.). Retrieved May 31, 2022, from https://www.thymeleaf.org/doc/articles/standarddialect5minutes.html

Thymeleaf. (n.d.). Tutorial: Using Thymeleaf. Retrieved June 24, 2022, from https://www.thymeleaf.org/doc/tutorials/3.0/usingthymeleaf.html