**Term Project Report for**

***ITCS 6112***

**TOPIC: 49er MarketPlace**

**Presented To: Presented By:**

**Dr. Ali Sever Dipesh Chakrani (801080983)**

**Achal Shah (801074085)**

**Shrav Mehta (801085764)**

**Namra Desai (801084948 Nayan Milind Mahajan (801024259)**

Table of Contents

**Abstract**

**I. Introduction**

1. **Purpose**
2. **Project Scope:**
3. **Assumptions:**
4. **Constraints:**

**II. Software Requirement Specifications**

1. **Functional Requirements**
2. **Non-Functional Requirements**
3. **UML Diagrams**

**a. Use Case Diagram**

**b. Class Diagram**

**c. System Sequence Diagram**

**d. Activity Diagram**

**4. Programming Languages**

**5. Development Methodology**

**III. Screenshots**

**V. Conclusion**

**VI. References**

Abstract

Marketplace is online ecommerce website which is used to buy and sale items from students/alumni and faculty. Transactions are operated by marketplace owner/ service providers. Here, Users can access the items based on their interests and availability of items. Users can post their items that need to be sold out with detailed description including its price. Then, third parties contacts user for that item. The transactions are operated by the application. Moreover, We can rate the buyer or seller based on our experience. This also helps buyers about future transactions.

Third party is basically buyer who is interested in the item. Every item can browsed on the website and can be viewed based on the preference. The browsing of item is operated on both the ends. Transactions can be multiple based on item count. Both buyer and vendor gets the notification for any updates on the item’s post. Therefore, we can say that it sophisticated online item purchasing medium which eases the efforts of vendors and buyers.

# Introduction

1. Purpose:

Our aim of this project is to give develop marketplace for UNCC student, professors, alumni etc. This marketplace can be used for items sale. This marketplace won’t be accessible to outside UNCC. Both vendor and buyer are from UNCC.

1. Vendor:

Vendor needs to post the advertisement/post on marketplace about the item that needs to be sold out. Vendor should mention details of the item including its price.

b. Buyer:

Buyer is basically audience other than vendor. All UNCC people can access the items that are available for sale. They can browse the items based on their interest in buying the items.

2. Scope of the project:

The project is developed on the centralized server which has all the data about the items and it is accessed by the application. Hence, User can perform all the tasks via marketplace server/operators.

3. Assumptions

1. All product are genuine. No verification.
2. No verification on person’s identity.
3. There can be identical posts. Vendors are posting one Ad per item.
4. Server is always online.
5. Transactions do not have any outstanding failure.

4. Constraints

1. All accounts are created by UNCC people only.
2. No other person buy the item.
3. No cash transactions.

# 2. Software Requirement Specifications

1. Functional Requirements:

The application should allow seller to:

a. Login into his account with credentials.

b. Allow to post multiple advertisements.

c. Can contact customers.

d. Should get notification on updates of the advertisement.

e. Should allow to edit existing advertisement

f. Allow to add detailed description.

h. Should get update on sold items as well.

The application allows Buyer to:

a. View all the advertisement based on their search.

b. Should able to do transactions any advertisement of item.

c. should allow to read description and should allow to contact seller.

1. Non-functional Requirements:

a. **Security**:

Only UNCC people can access the items. This should handle at the admin level. The transaction of banking should be very secure.

b. **Maintainability**:

The application itself is isolated in two fragments each with specific functionality. This application should be controlled at operator level. Hence, it is reliable on operator level of marketplace.

c. **Scalability**:

This application highly accommodated by items and the inventory should be capable of handling large data. And it should also allow a greater number of users to access the items.

d. **Online** **Functionality**:

This application is can be completely handled online. Functionally is more useful on online cloud technology or similar ones.

3 . UML Diagrams

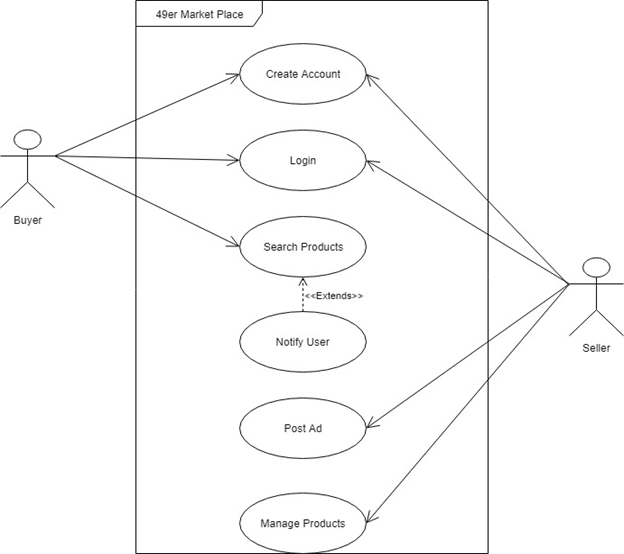
* 1. **Use Case Diagram:**

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

**Actors**:

Buyer: Buyer logins to his account, search for the product according to his likings. The system then notifies the seller.

Seller: Seller logins to his account and can post and ad for new products as well as manage existed product he has posted to modify details or to delete them.



Use Case Diagram

Use Case Views:

The use cases are described below:

Buyer/Seller Use Case

Use Case: Create Account

Description: User need to have account to verify he is student or a faculty through Niner credentials for authentication purpose.

Buyer Use Case

Use Case: Search Product

Description: User Searches for the product according to his liking and can notify user if he she is interested in the product.

Seller Use Case:

Use Case: Post Ad

Description: Seller can post an ad for new product he/she wishes to sell.

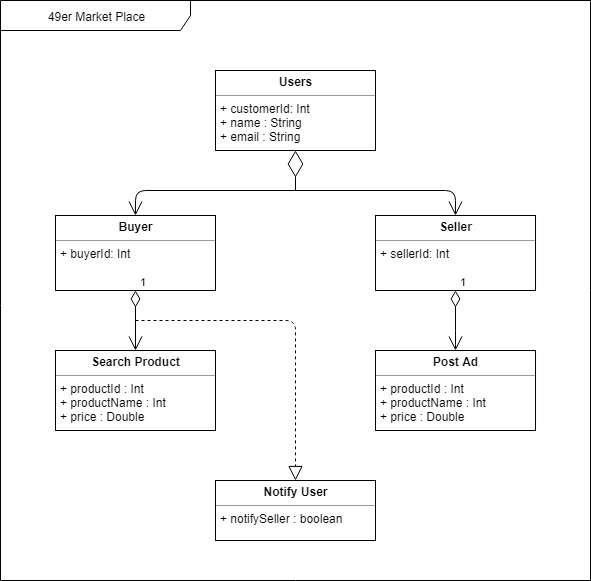
Seller Use Case:

Use Case: Manage Product

Description: Seller can update product details as and when he wishes.

b. **Class Diagram:**

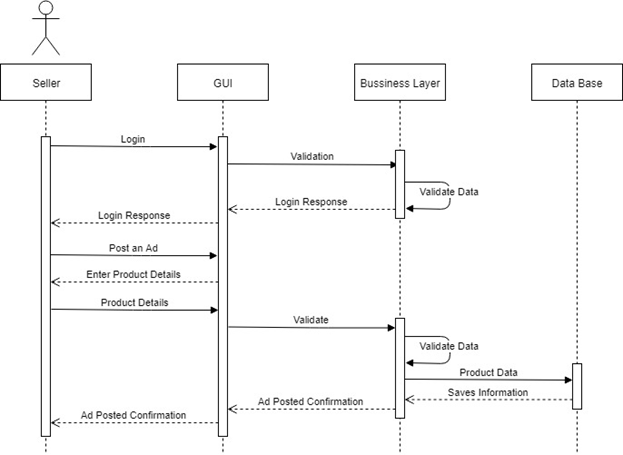
A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the systematic of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.



Class Diagram

1. **System Sequence Diagram:**

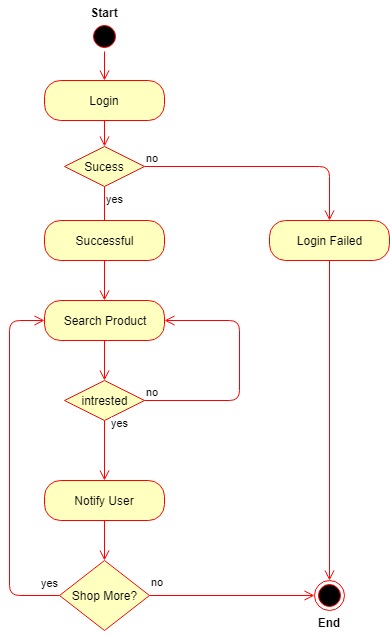
A graphical representation of a module’s function invoking functions of other modules in order to achieve a task (specific user requirement) is called a sequence diagram. A sequence diagram for the billing process is given below for reference.



Sequence Diagram

1. **Activity Diagram:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.



Activity Diagram

1. Programming Languages:

For this project we have taken HTML5, CSS3 and java script as front end technologies and Node.js as business layer to connect UI to the data base which is MongoDB (NoSQL).

What motivated us to take up this language for the project is The Node run-time environment includes everything you need to execute a program written in JavaScript. you can do much more with JavaScript than just making websites interactive.JavaScript now has the capability to do things that other scripting languages like Python can do. Both your browser JavaScript and Node.js run on the V8 JavaScript runtime engine. This engine takes your JavaScript code and converts it into a faster machine code. Machine code is low-level code which the computer can run without needing to first interpret it.

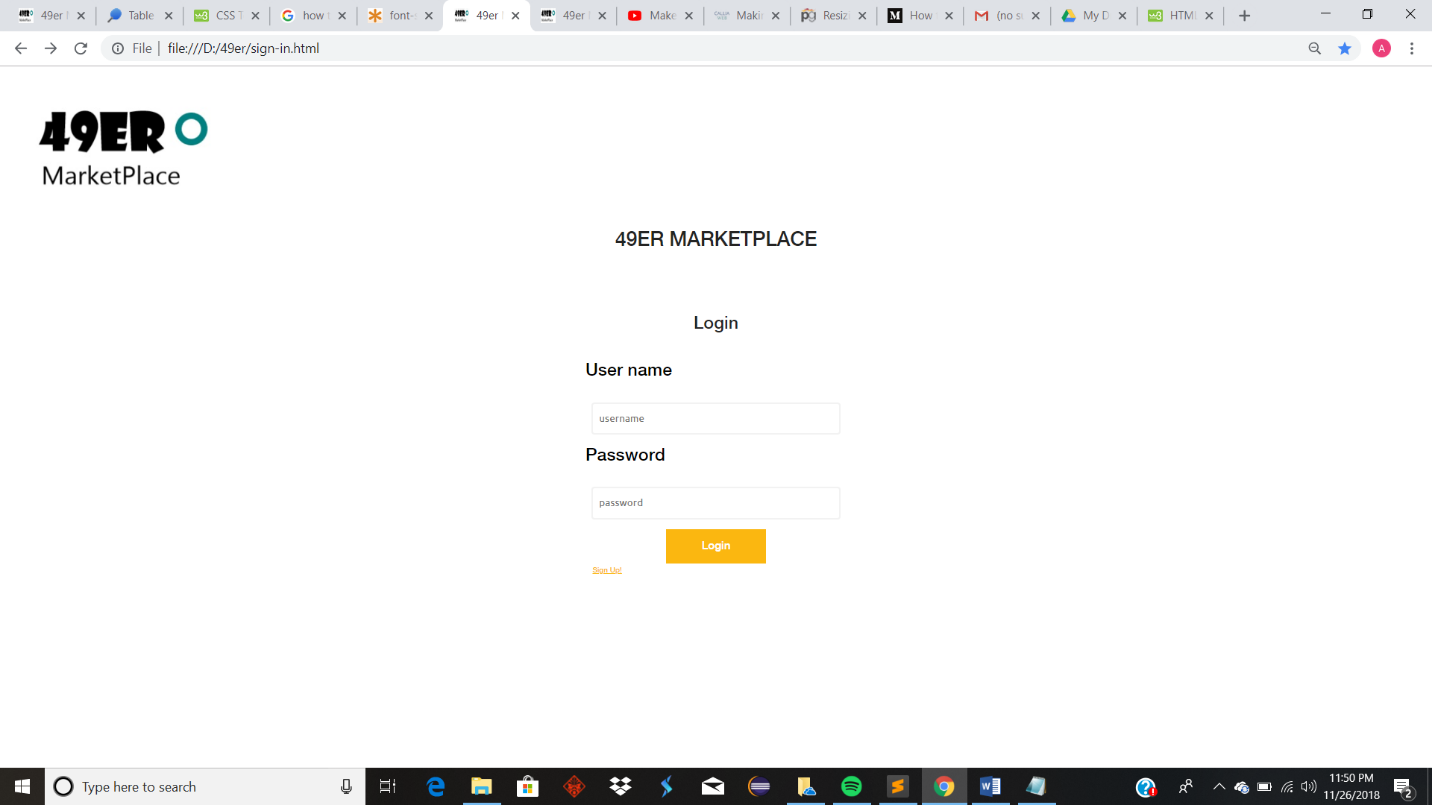
We have also used MongoDB for real time database. MongoDB stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time. The document model maps to the objects in your application code, making data easy to work with Ad hoc queries, indexing, and real time aggregation provide powerful ways to access and analyze your data MongoDB is a distributed database at its core, so high availability, horizontal scaling, and geographic distribution are built in and easy to use.

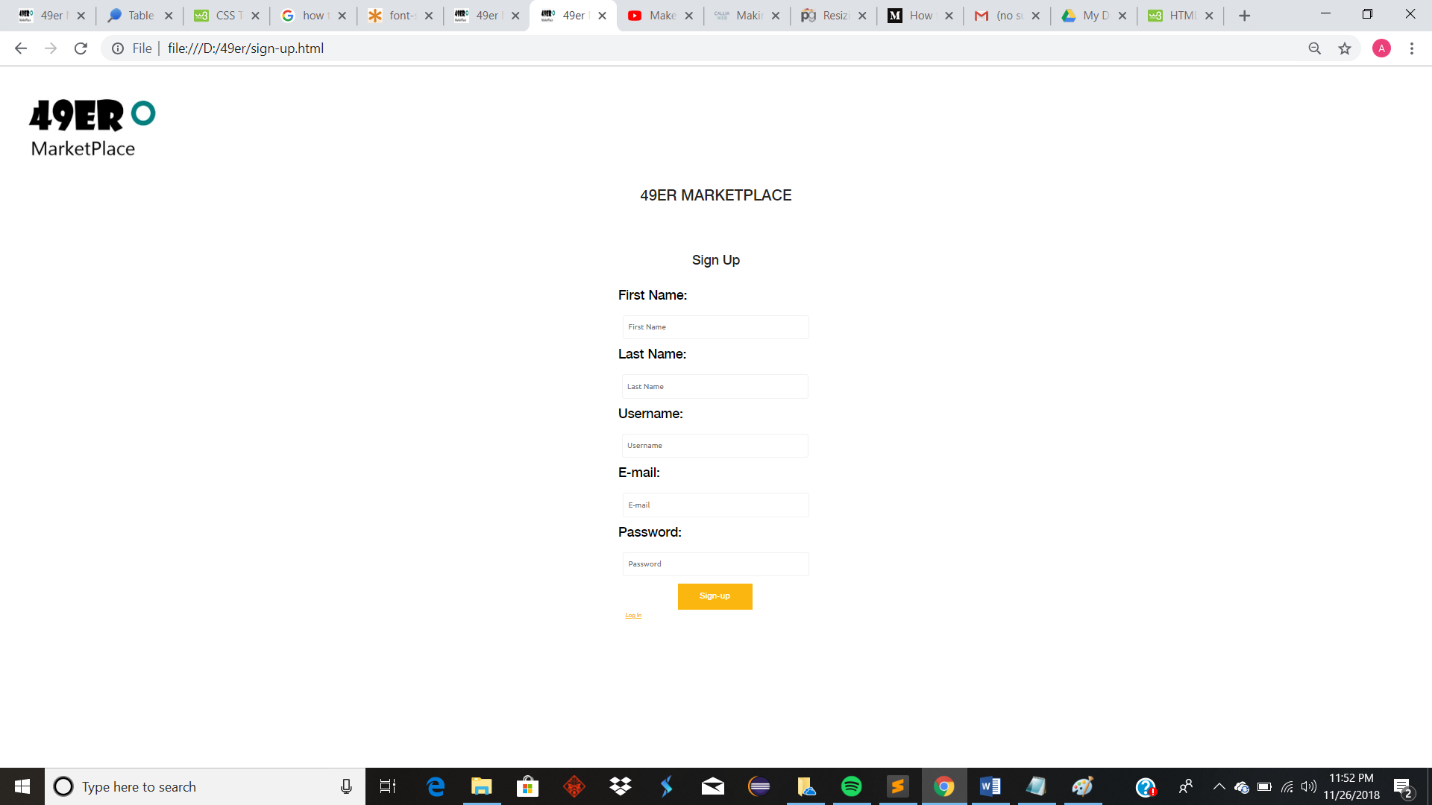
1. Development Methodology

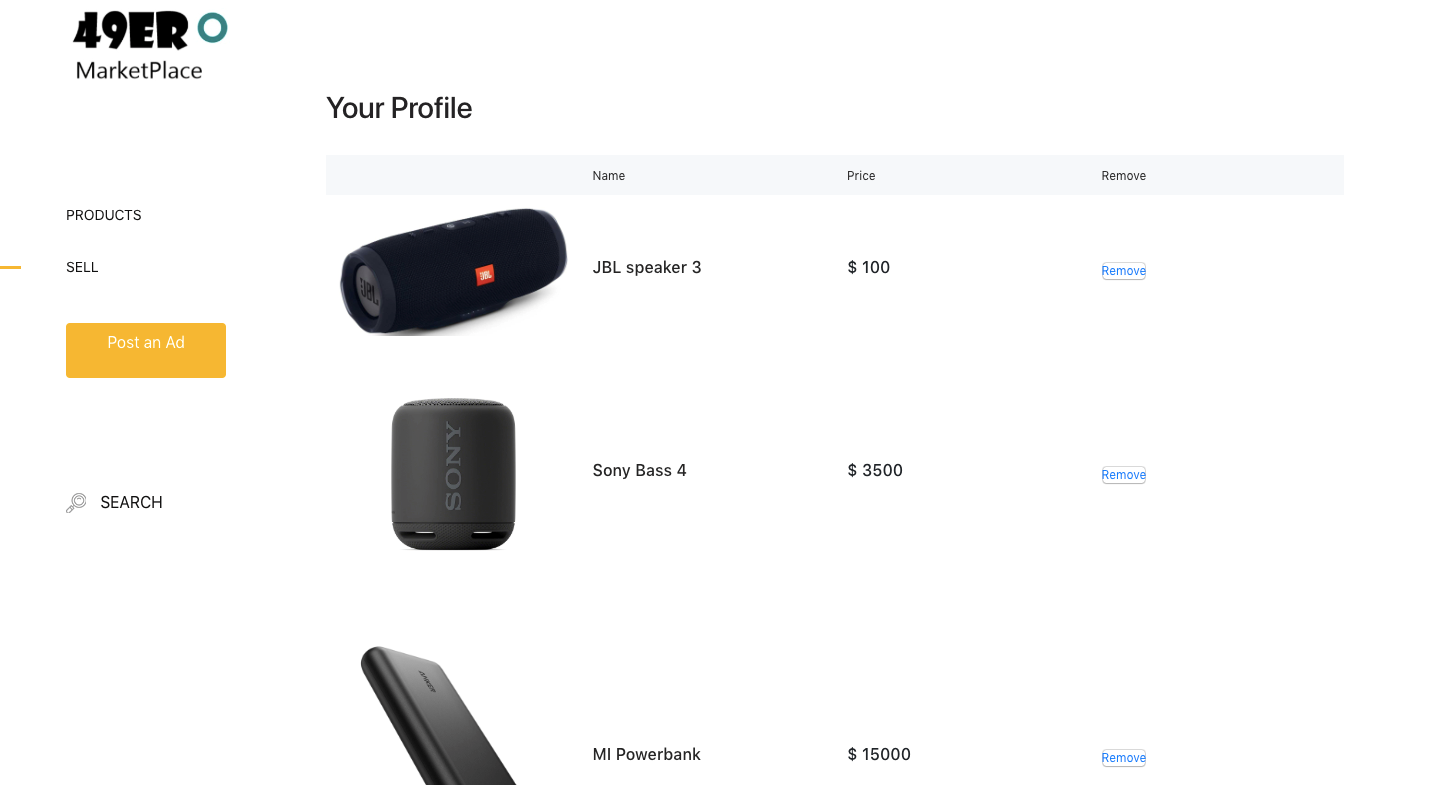
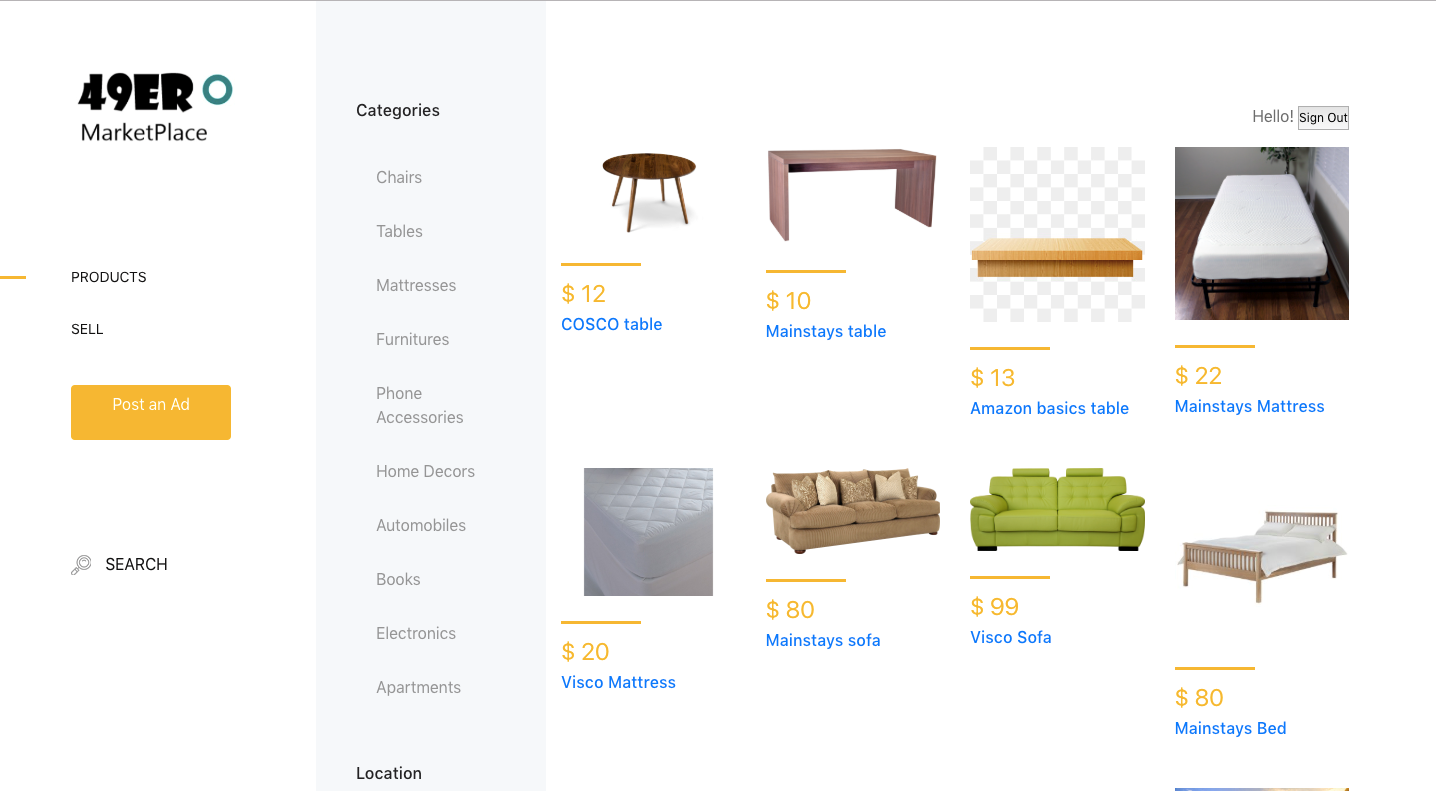
The methodology used in development for the term project is ‘Agile Methodology’. Herein while developing the project we subdivided our project into multiple stories and distributed the project work according to the stories. The stories that were built were majorly sub-divided into 2 main categories i.e. Buyer and Seller.

1. **Login**: As a user, I can login into the system either as a buyer or a seller.
2. **Buyer**: This section includes various actions that a buyer can perform. Below are the description of each sub-stories that we created during our development process.
   1. **Story 1**: As a Buyer, once I log in into the portal, I should be presented with all the product available for sell along with their rates.
   2. **Story 2**: As a Buyer, I should be able to search products that I intend to buy. I can search product by entering keywords and I should get the matching products.
      1. ***Story 2.a:*** As a Buyer sould be able to see search button and when I click on it I should be able to see a search box where I can enter keywords .
      2. ***Story 2.b:*** As a Buyer, I should be able to enter keywords and on clicking on button or pressing enter key, I should be able to see search results.
      3. ***Story 2.c:*** As a Buyer, after clicking on product, I should be able to see product details and on clicking on contact seller an email should be triggered to seller notifying him/her that seller is intrested in product.
3. **Seller**: This section includes various actions that a seller can perform. Below are the description of each sub-stories that we created during our development process.
   1. ***Story 1***: As a Seller, I should be able to post an ad for the product I intend to sell.
   2. ***Story 2***: As a Seller, I should be able to enter product details and images.
   3. ***Story 3:*** As a Seller, I should be able to see all the products on sell by me on a separate page and also should be able to manage the same (edit and delete existing products).

III) ScreenShot







Conclusion - A Marketplace exclusive for UNCC students/alumni and faculty that can communicate to sell/buy products.