**WEB APP**: Inventory Management

**NAME**: Umair Rashid

**STUDENT #:** 100312938

**DATE**: 11/30/2018

**CLASS**: INFO 3135

**DOCUMENT**: PROJECT REPORT

**Description:**

Inventory management can be used by the participating stores to view inventory across all stores and update their own inventory. This application can search, view and update inventory from database with some very useful feature which can make life easy for business users such as data validation and easy to use interface. This application has a login system which can help provide the security to the business inventory to prevent any unauthorized use to the application

**Motivations of the app:**

Inventory management is one of the most critical tools in retail industry, and it is vital to keep track of the inventory as the store grows the inventory can grow rapidly and keeping track of the inventory can get very difficult. I wanted to create an application which help businesses with inventory management. I know most student application developers would love to make a game, but I love making application that are useful and can be used in a genuine business or daily life. I find games fun to play but I find that application that can make life easier is more fun to develop.

**Features:**

**Login Page**

One of the best features of the application is security that only an authorized user can access this application. The application has a database which stores the username and password for the different stores. The stores username is their 4-digit store number. The application itself has a data validation system that can detect usernames that are incorrect and will not allow the user to submit data without inserting the correct username. This system helps us to send unnecessary request to the database. Once the user enters the correct username and password. The user can use the application.

There are 10 stores in the database right now and each of them has been assigned a password. Currently I am using my first or last name as passwords for the stores to easily memorize them. The passwords are stored using hash method in PHP to ensure security of the passwords.

The login system ensures that once a store signs out from the application, it will not be able to get into application by just entering the right URL. It must sign in again to use the application.

**Database**

This application is connected to the database called inventory\_database. This database has 4 different tables in it:

* Inventory
  + Product ID
  + Store ID(Store Number)
  + Quantity
* Product
  + Product ID
  + Description
  + Price
  + Weight
* Stores
  + Store ID
  + Street Number
  + Street Name
  + City
  + Province
  + Phone Number
* Userinfo
  + Username
  + Password Login

Inventory has all the information about which store is carrying which product of how much quantity. Product table has information about the product based on the product ID. Stores has store information based on storeID(also called store numbers) and the userinfo has passwords based on store Numbers.

**Navigation Bar**

This website has a navigation bar made using the library bootstrap. This navigation bar displays the store number of the store that is logged in the application. This navigation bar is displayed at every page except the login page. The navigation bar has the following links:

* Update
* Search
* Change Password
* Sign out

**Search Page (Page right after login)**

As soon as the user logs in, the first page the user will be re-directed to is the search page where we can search inventory. Inventory can be searched by three different ways:

* By Product ID
* By Product Name
* By Store Number

Based on the search the user will be directed to a different page to show the results. The results will be shown in a table. We are using a library called Datatables to display our tables. This library has a bunch features that can enhance the experience of the user such as search option within the table, sorting for columns, pagination and choosing how many rows to view per page.

**Update**

The update page is used to update inventory for the stores. The update page will allow you to update the inventory of only the store that has logged in the system. This is to prevent any other stores changing inventory for a different store. The update can be done by entering the product ID. If the product ID doesn’t exist in the system, then the user will not be allowed to submit and will be notified.

Once the user enters the correct product ID then a search bar will fade out and a we will see a new form that display all the information about the product we were looking for. All the fields will be grayed out that means that only quantity can be updated. It will also display the current quantity to show the user how many exist in the database as of now.

**Change Password**

This page allows the user to change the password for the user logged into the system. Here, the user will have to enter their current password, and their new password twice. The application will make sure that the current password is right and then checks if the new passwords match. After the running the checks, it will change the password for user and notify the user that the password has been changed and they use the new password the next time they login to the system.

**New concepts or insight Learned**

Bootstrap

One of the fun libraries I learned to use is called Bootstrap. It’s a very popular CSS library that can quickly and easily create beautiful web pages. It can also help create a number of different features of a webpage such as navbar, forms, buttons etc

Animation(jQuery)

I also had the change to try jQuery and use the animation feature in my UPDATE page. This library is built on JavaScript, and it’s one of the most popular libraries out there. Its being used by many in the industry, and its very simple to learn. There is a lot of documentation available for this library

DataTables

Another fun plug-in I used in my application is DataTables. I was actually very impressed at how easy it is to implement and it and how much of different it makes to the tables. This was one of my favorite plug-ins to use, and I definitely will be using this in the future

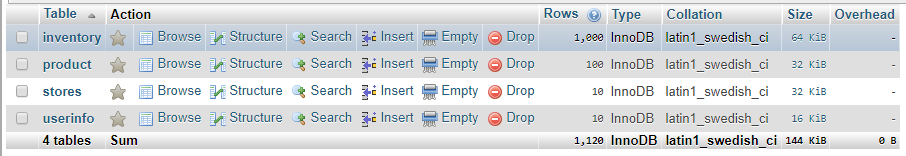
Data Validation

I heard a lot of people say how important it is to have data validation within the application to avoid inaccurate data, and I got to experience this first hand with this application. We, as developers, tend to feed the application the data its meant to deal with but when it comes to data validation we have to think of how can different types of data can cause errors. Basically, we must think from user’s point of view and take it into account how users can make errors and how can it impact our application. One of the example is that when we login we make sure that the store number is valid before even requesting any information from the database.

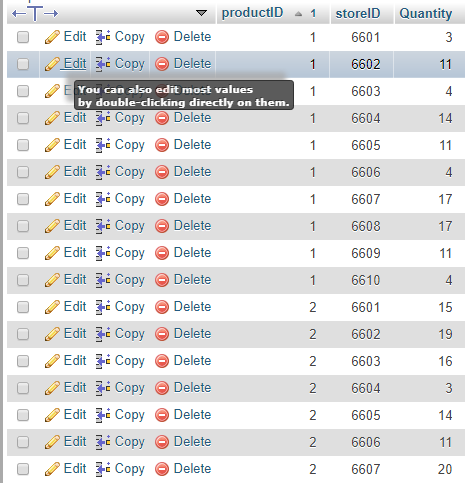
**Screenshots**

***Database***

All the Tables:



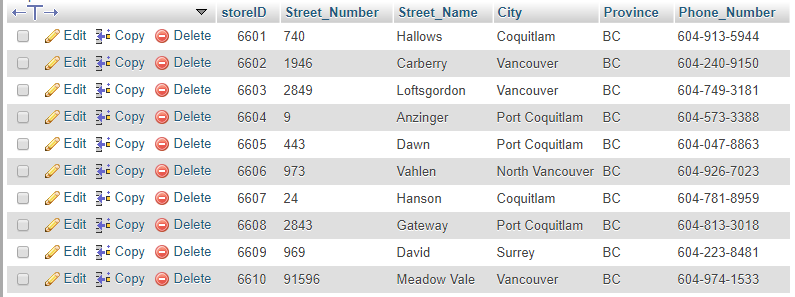
Inventory:



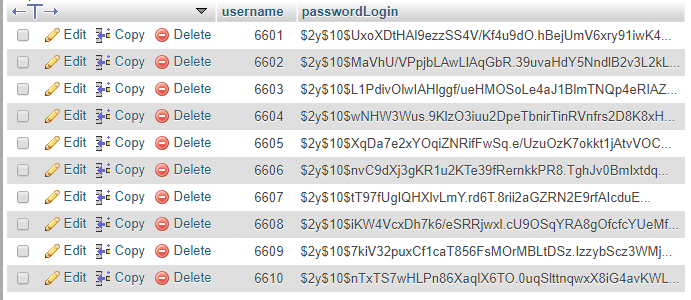
Product:



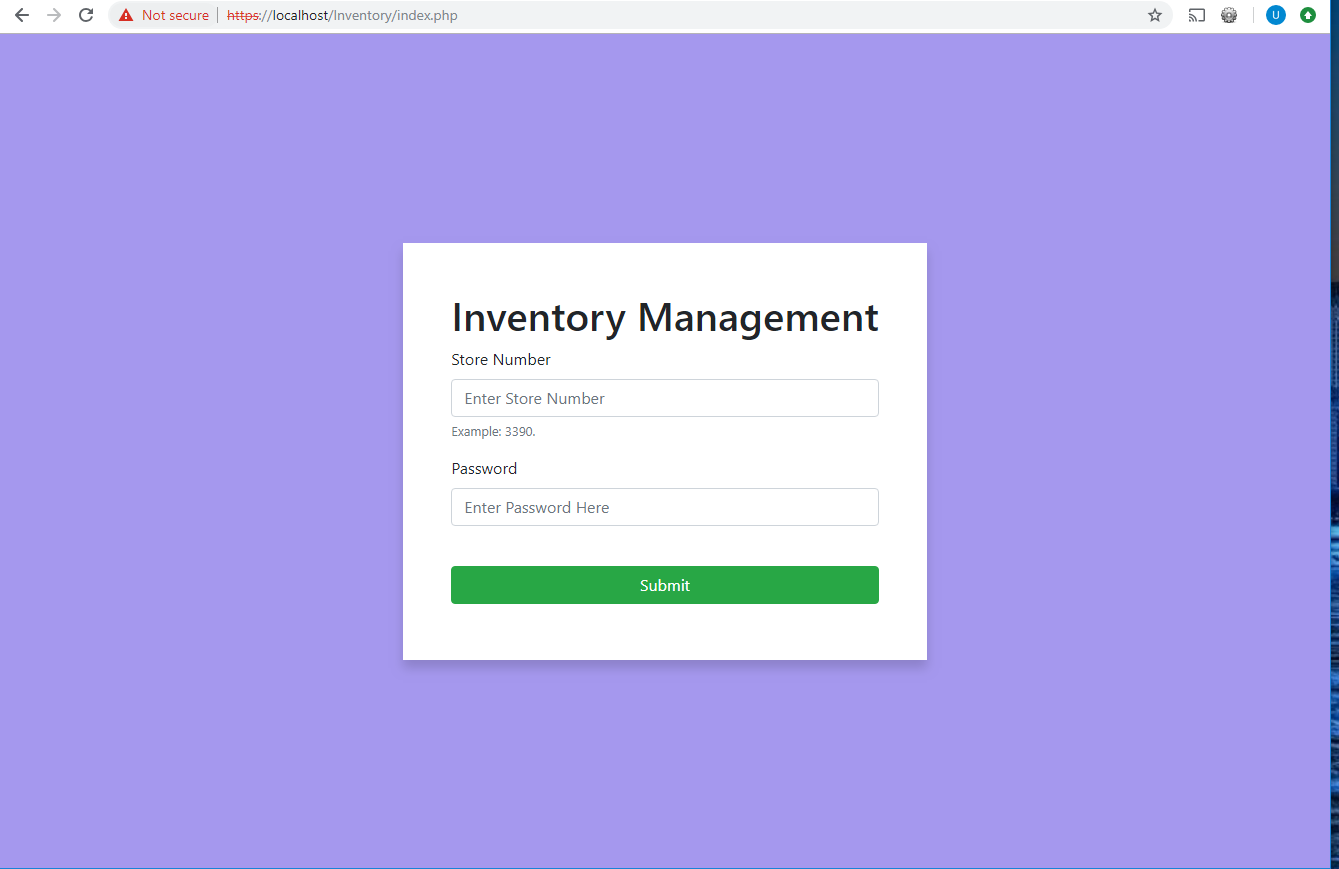
Stores:



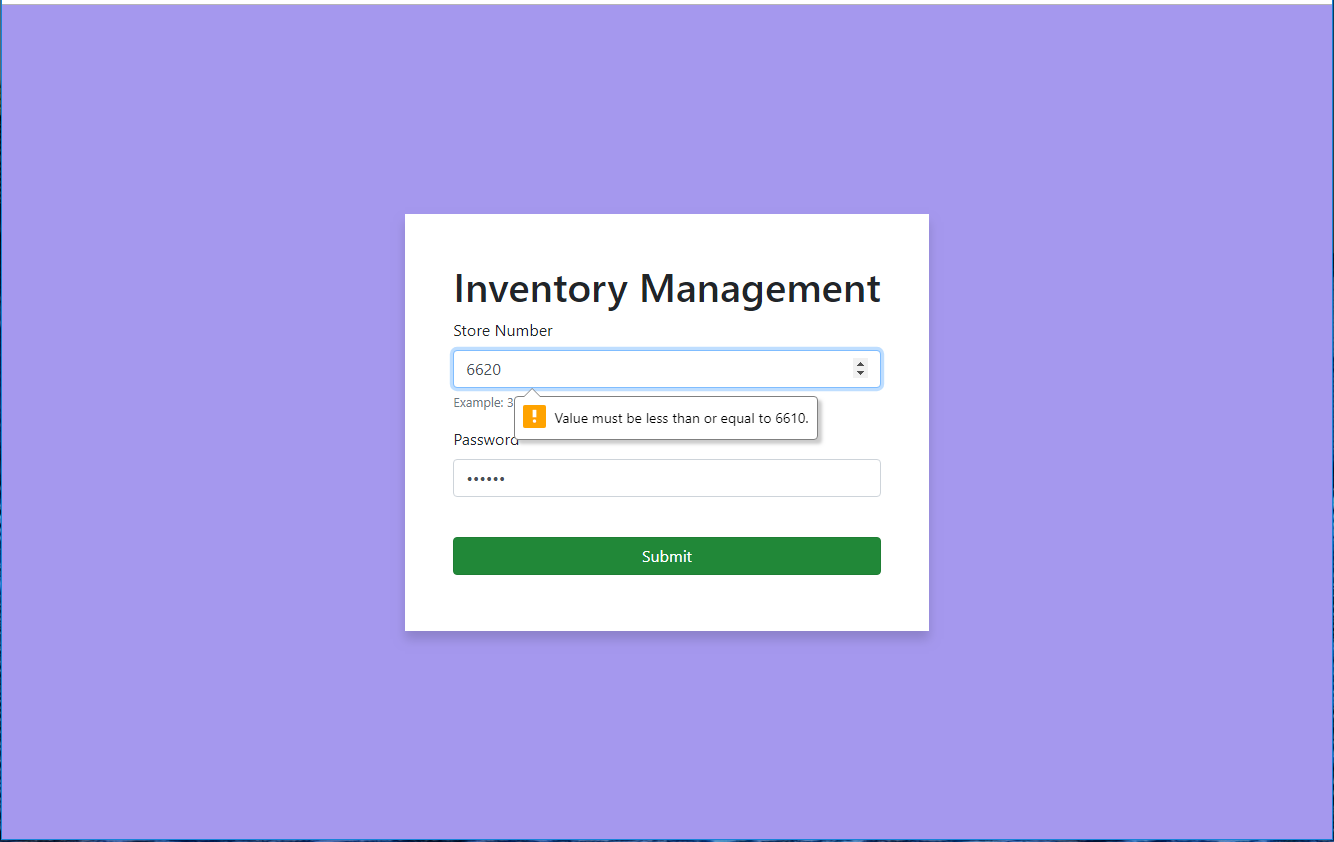
Userinfo:



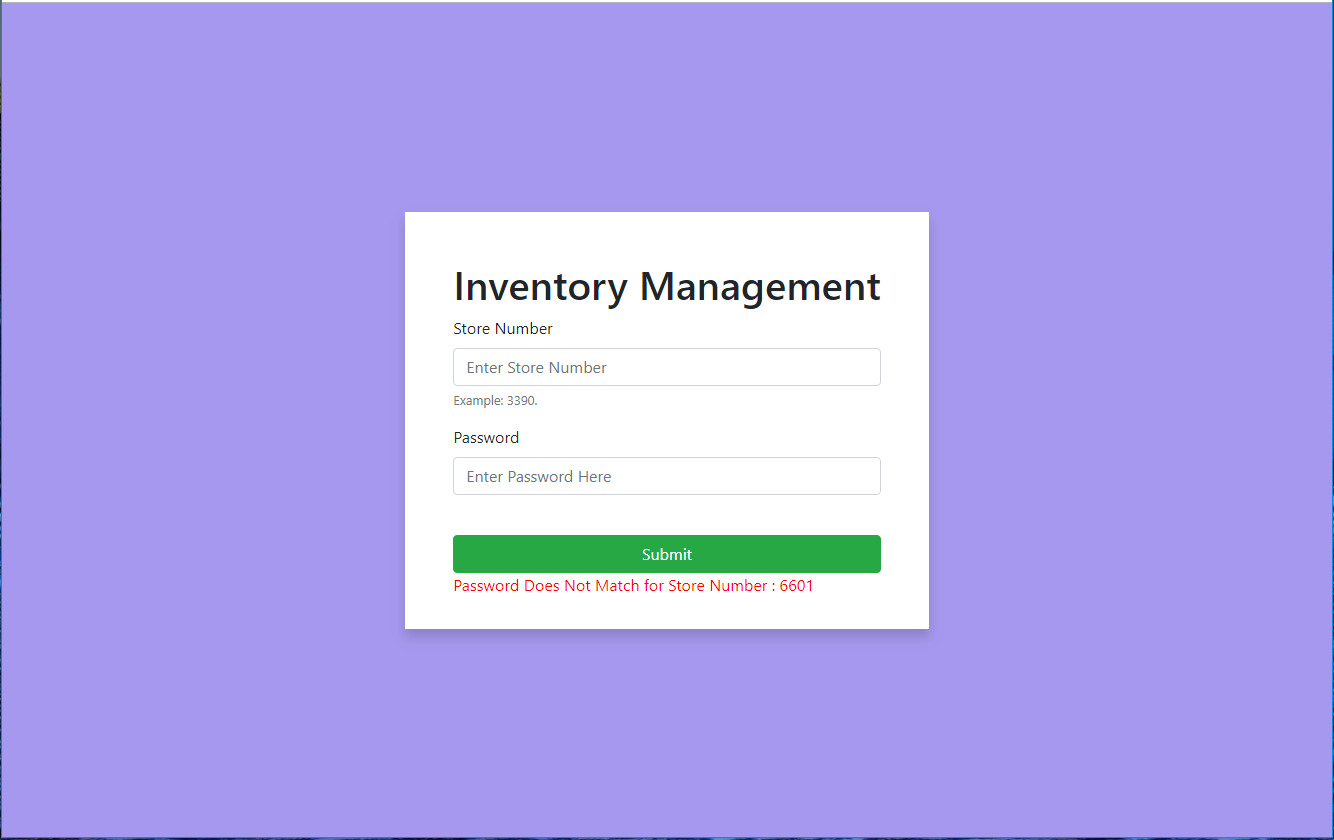
***Login Page***



Invalid UserName:



Invalid Password:

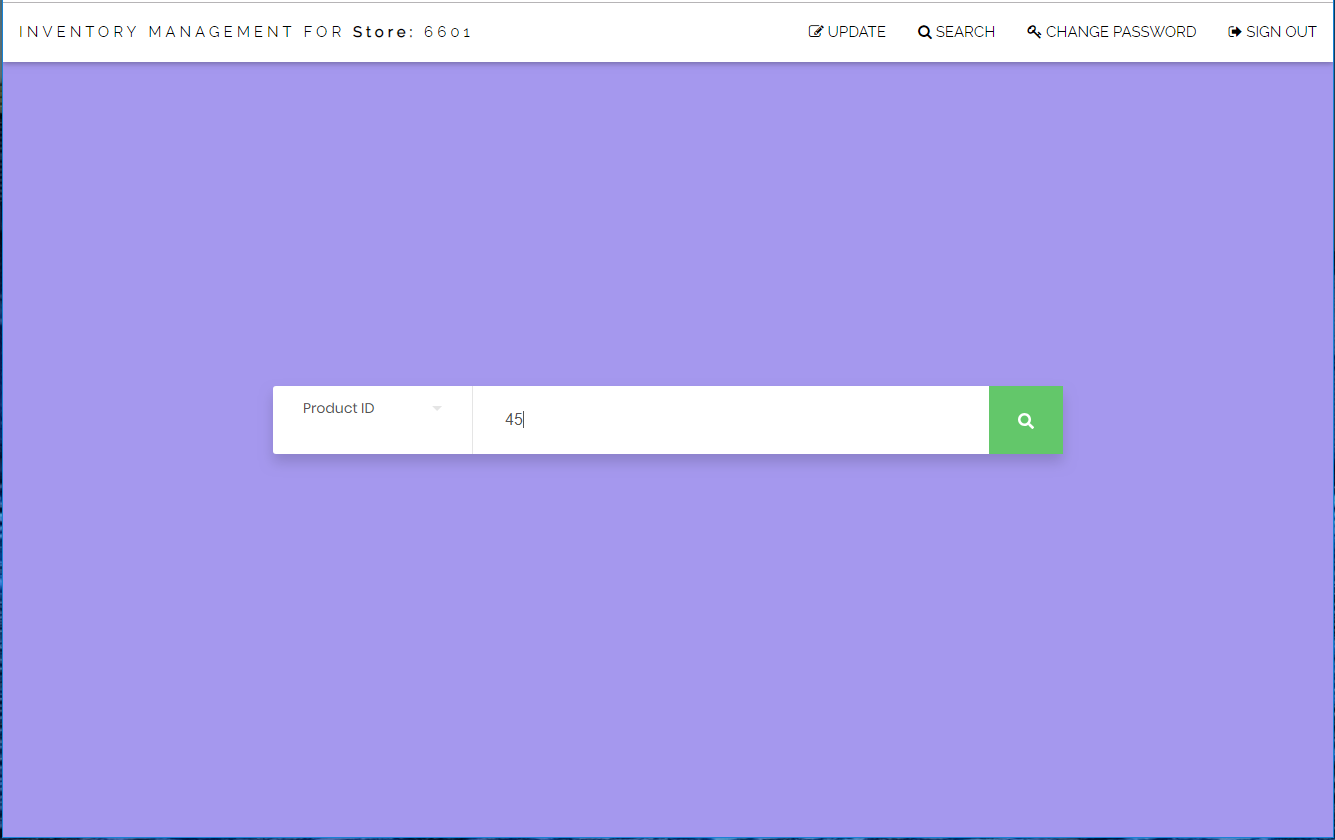


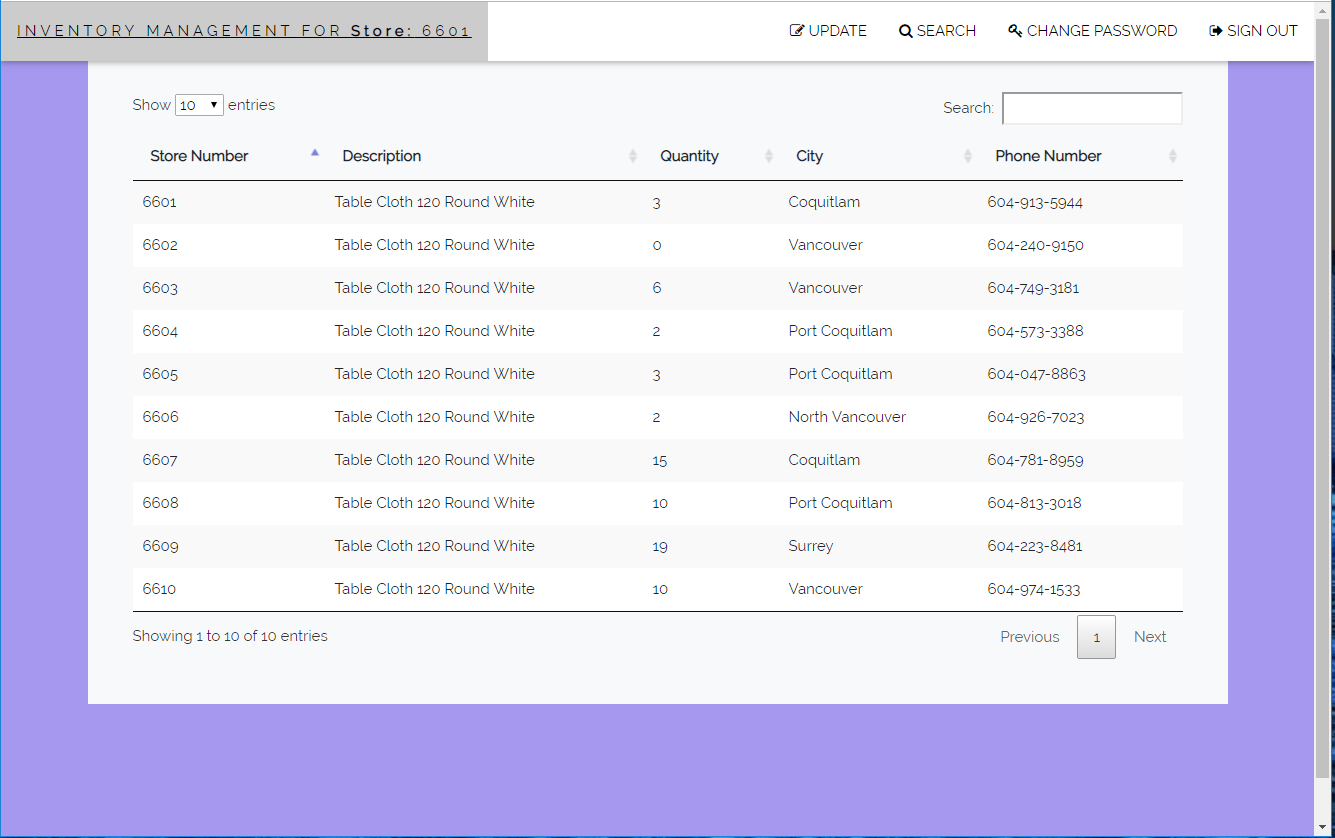
***Navbar Page (Logged in as 6601)***



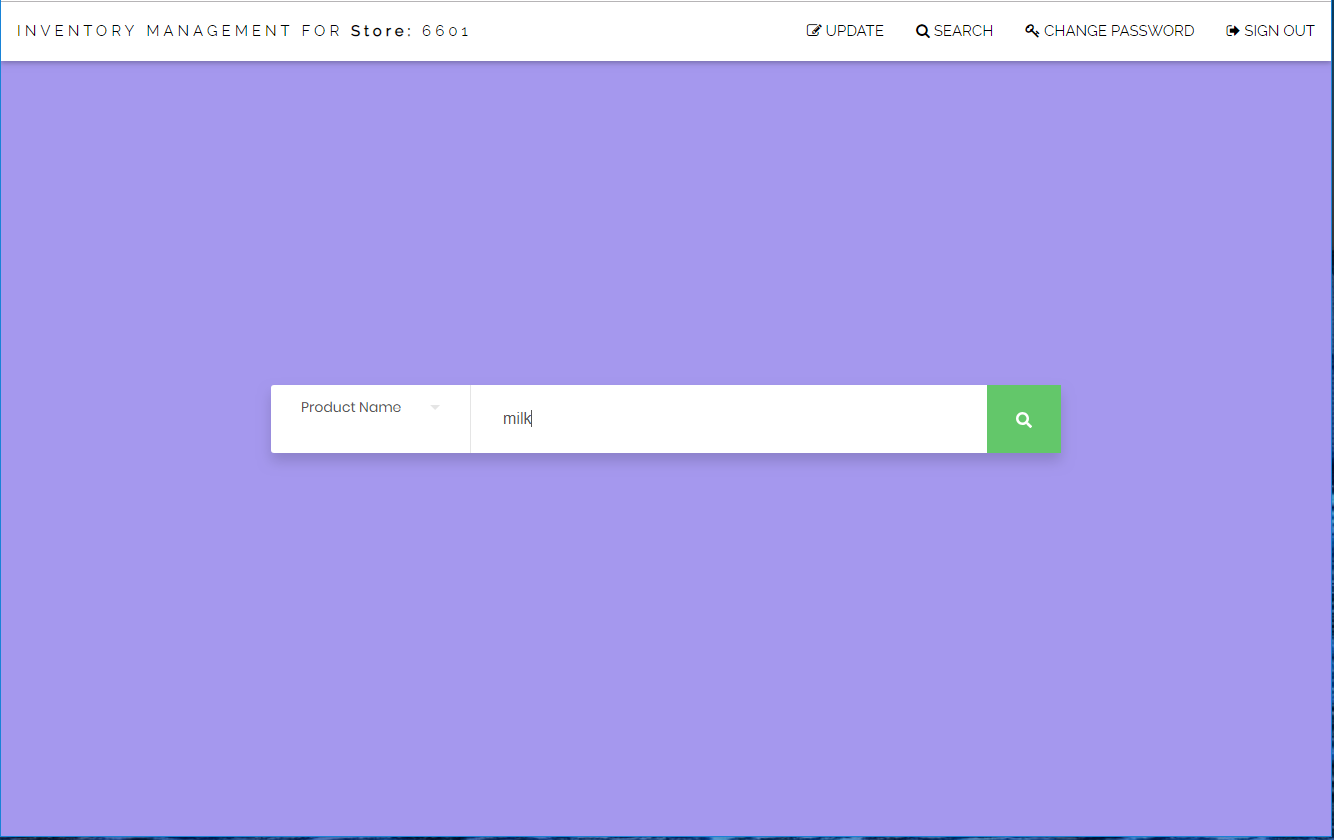
***Search Page (Logged in as 6601)***

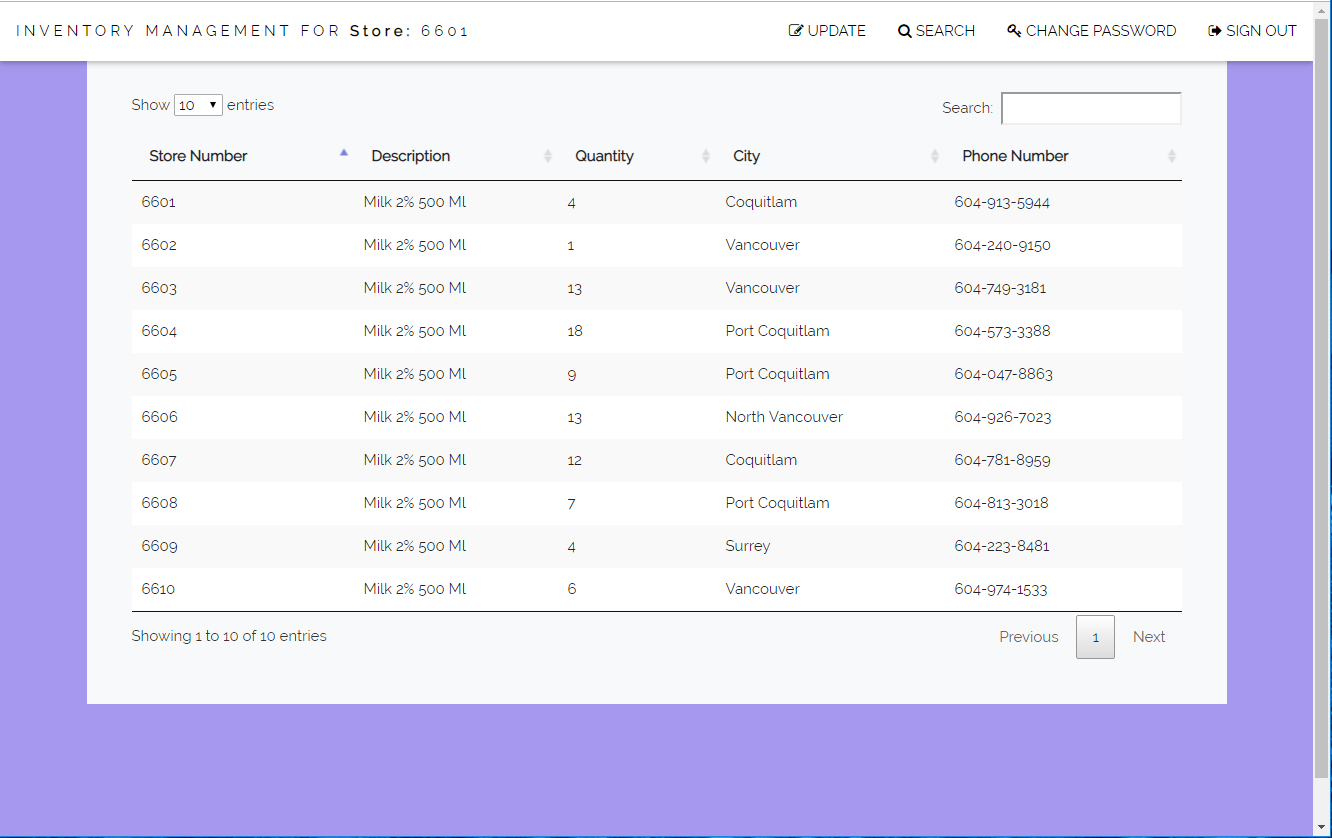
Search By Product ID:



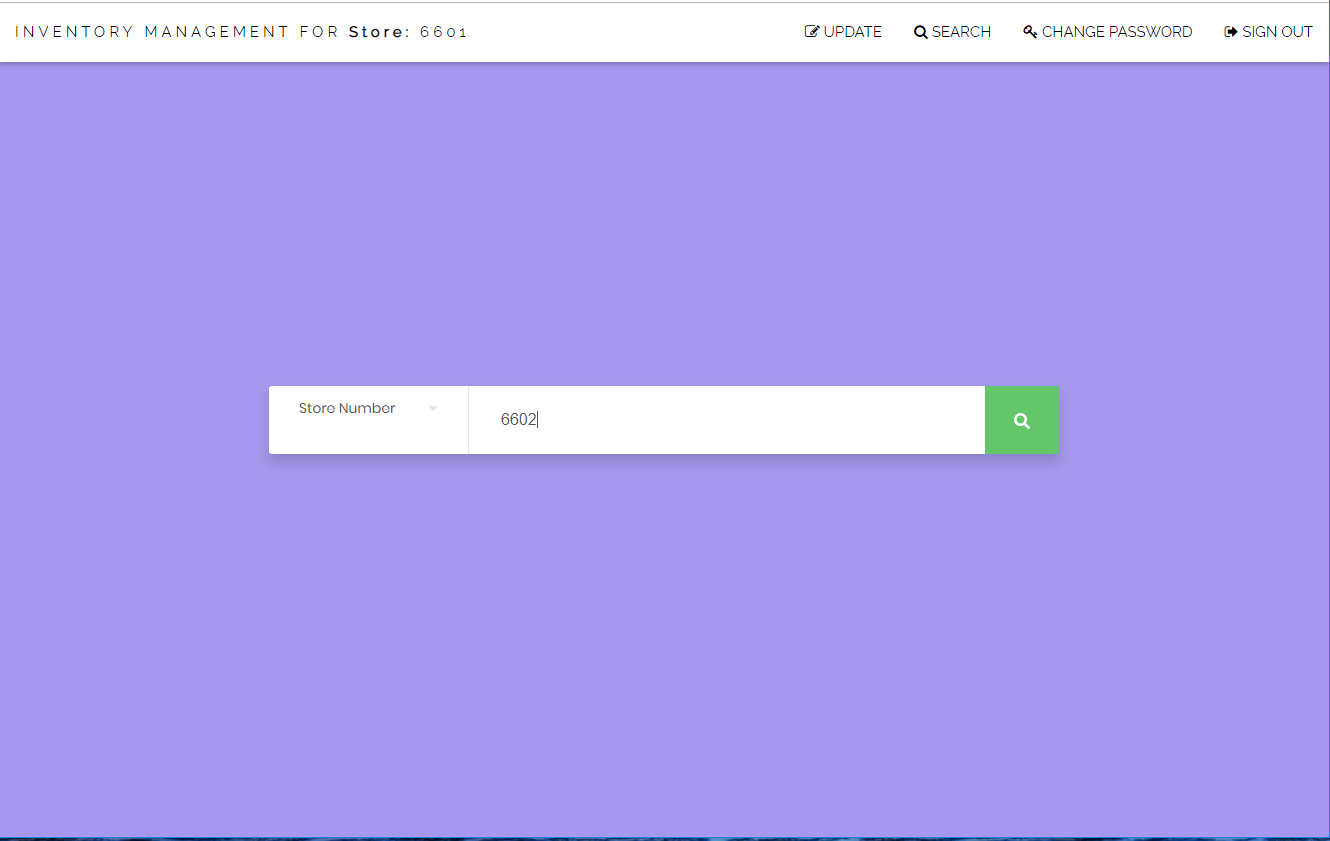


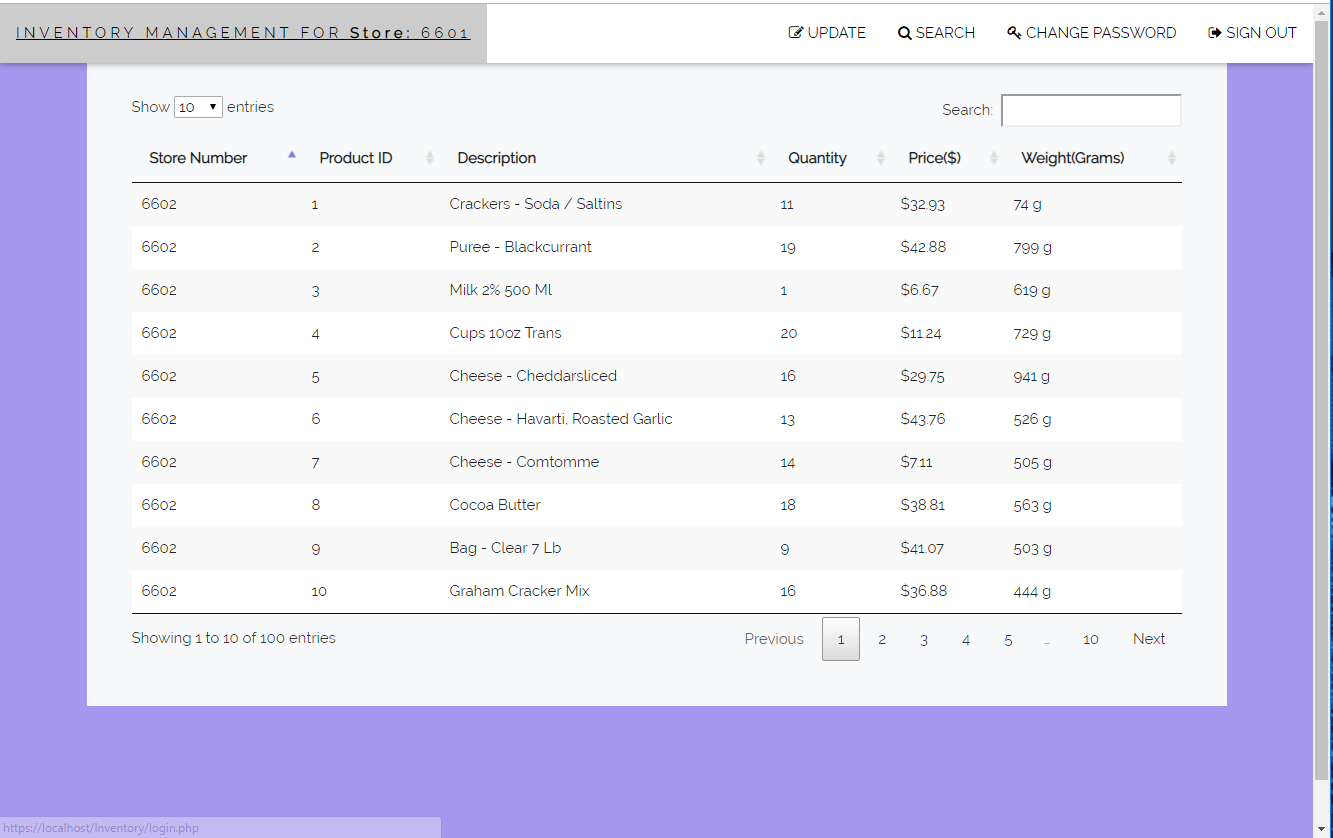
Search By Product Name:



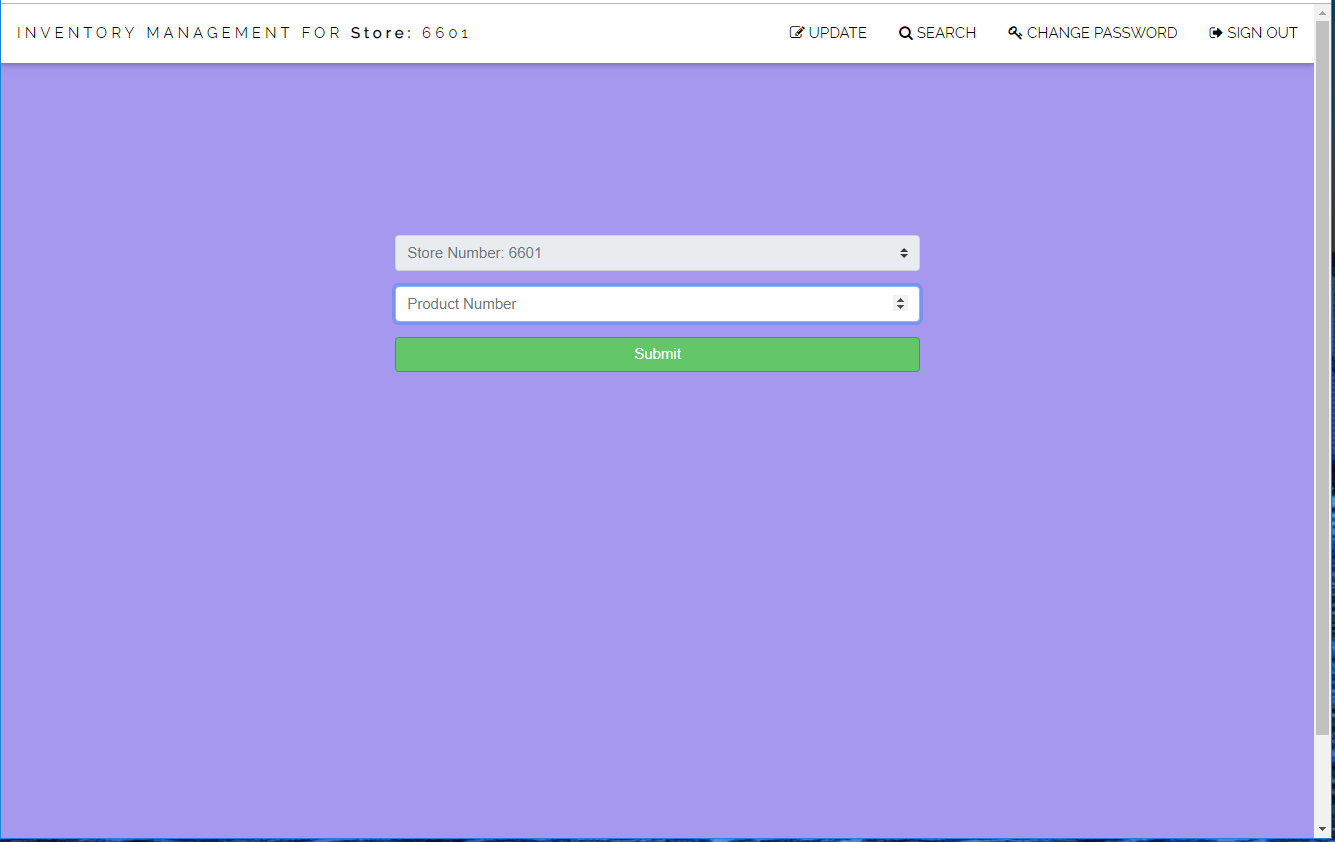


Search By Store Number:

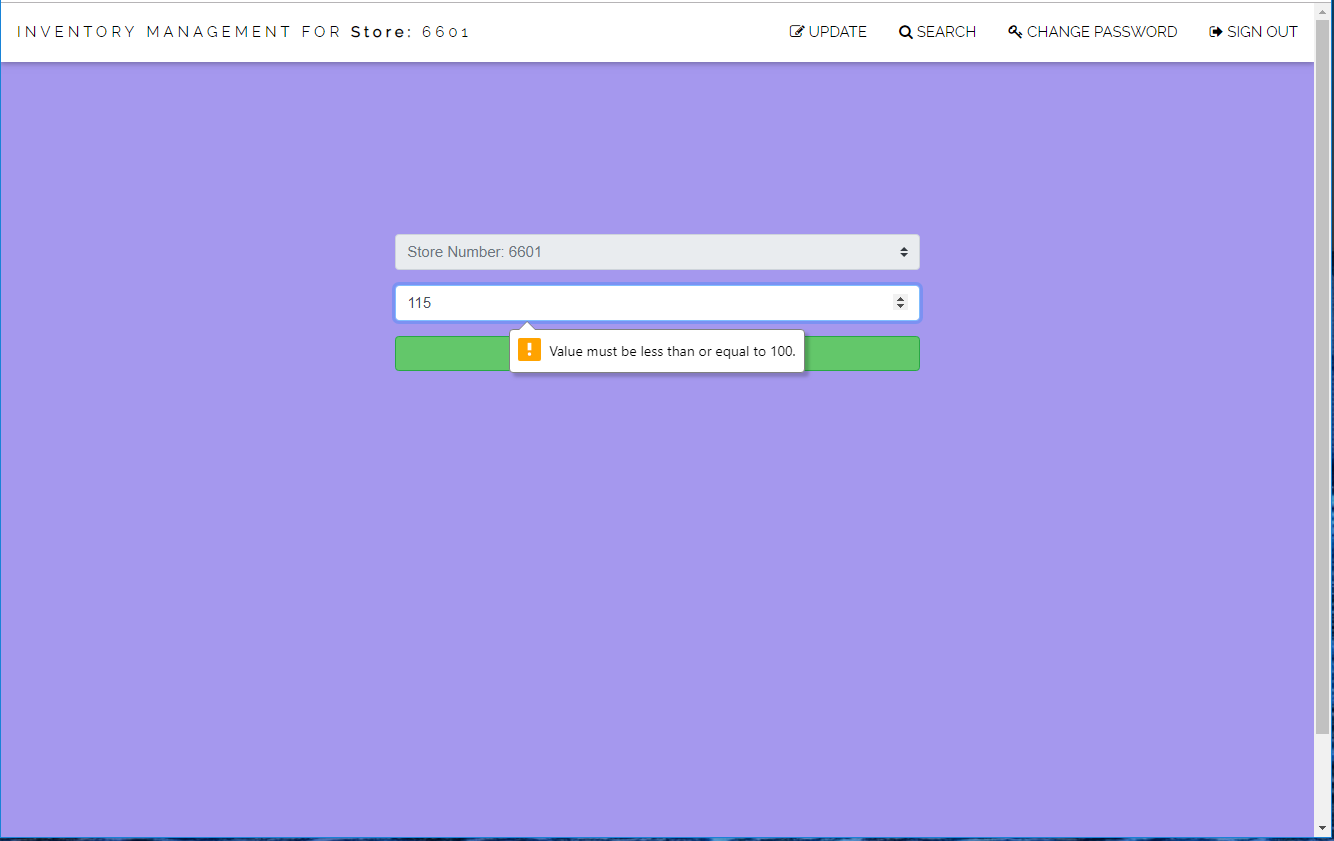




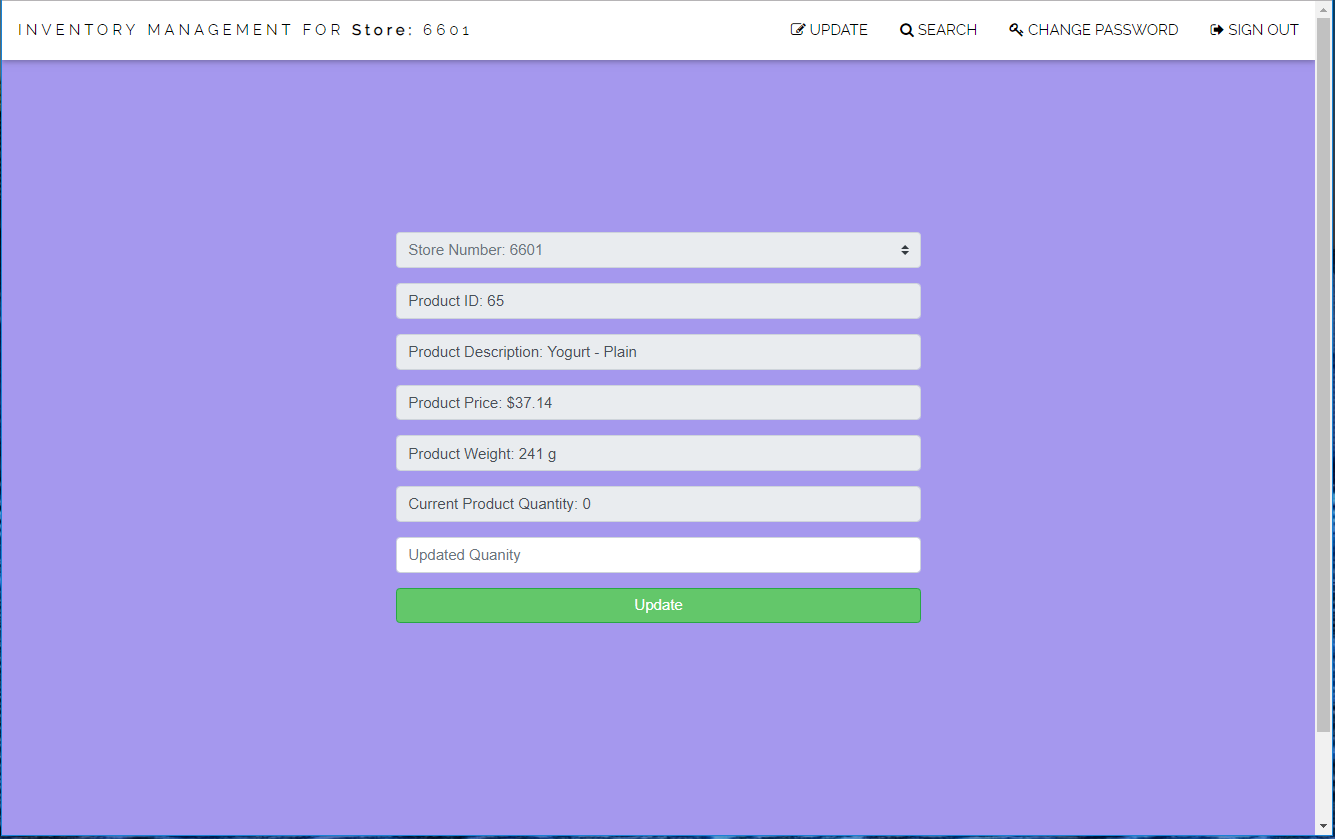
***Update Page (Logged in as 6601)***



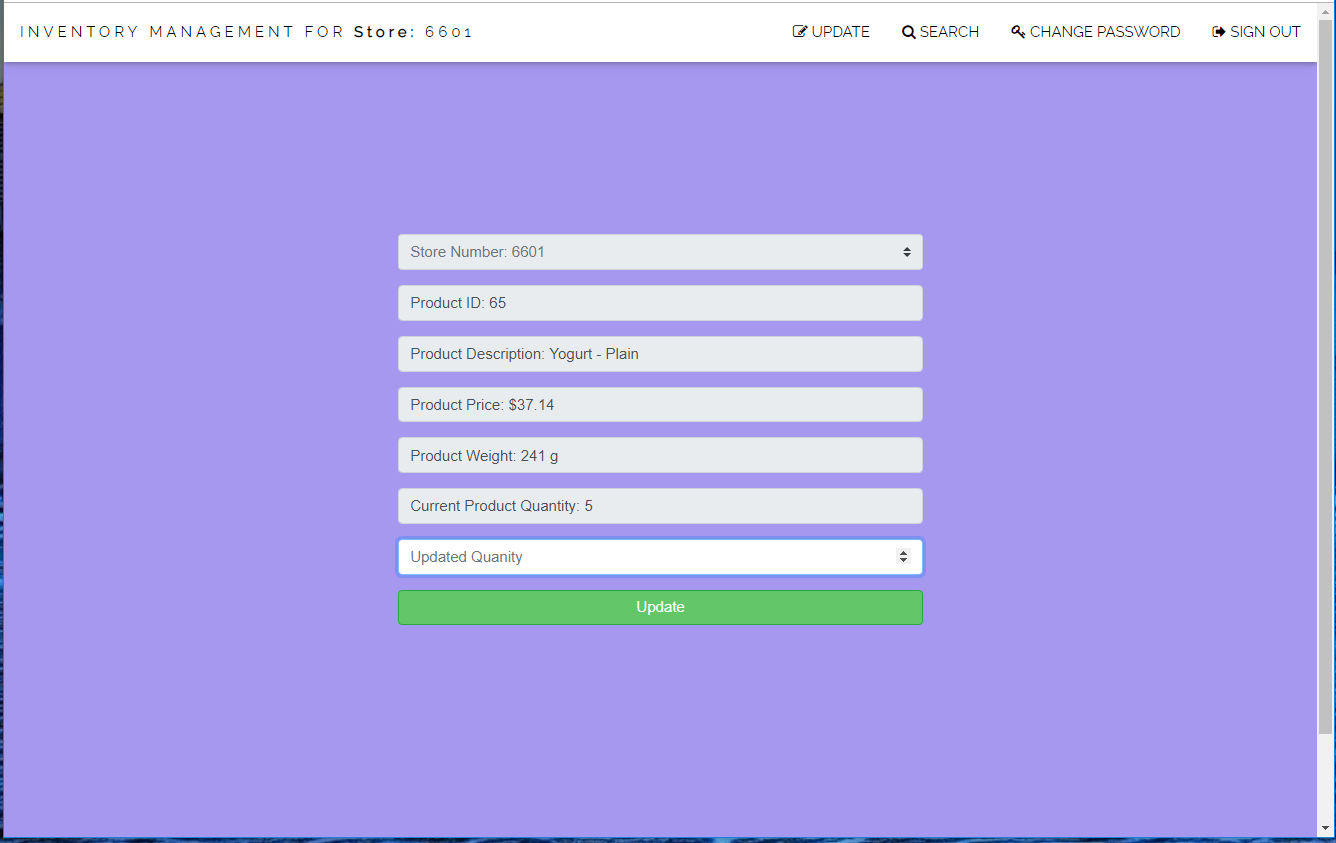
Invalid Product ID



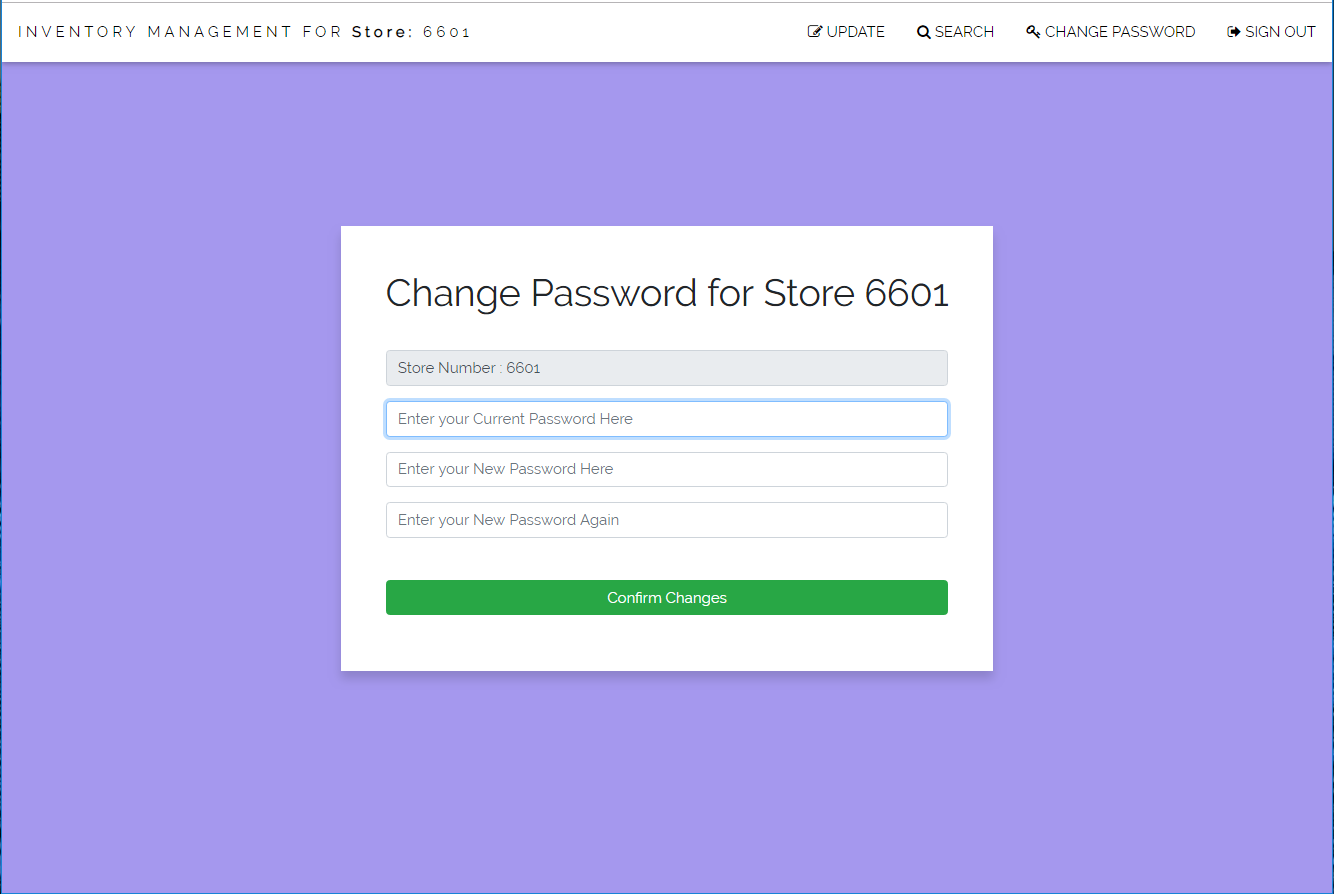
Search for Product 65:



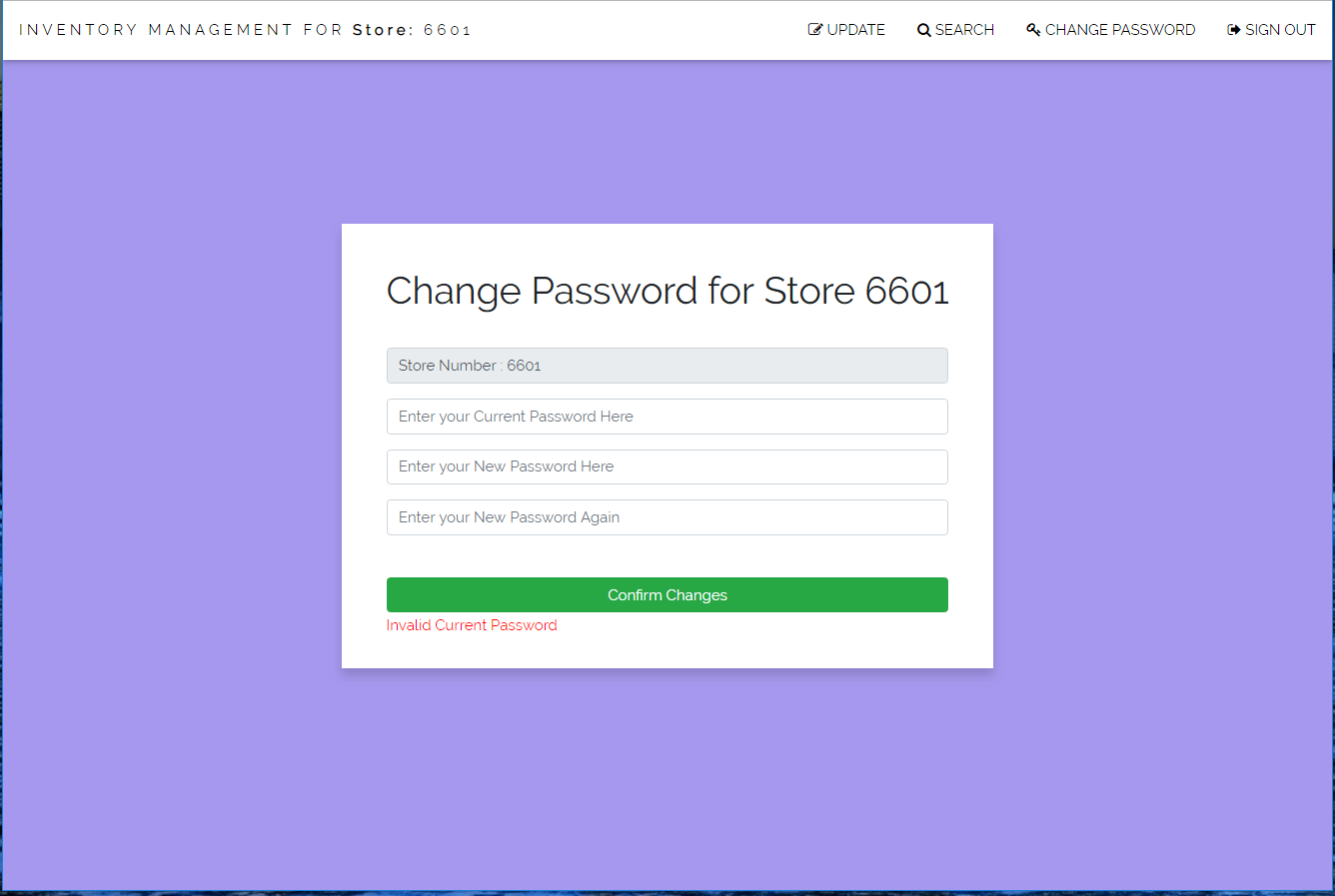
Updated Quantity to 5:



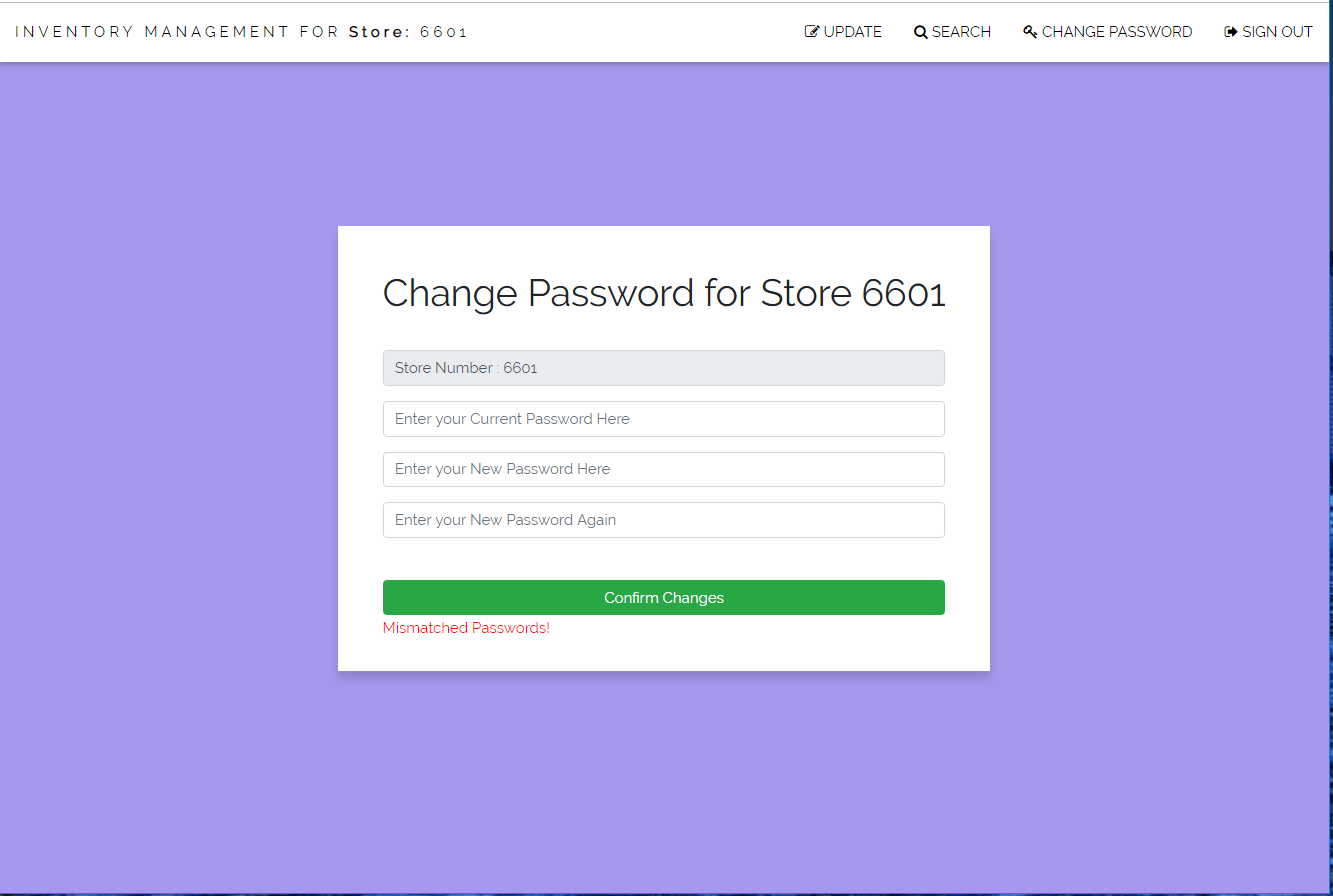
***Change Password Page (Logged in as 6601)***



Wrong Current Password



Passwords Don’t Match



Password Change Successful

