**Hardware Online**



Table of Contents

1. Title Page
2. Table of Contents
3. Introduction
4. Project Development
5. Requirements
6. User Stories (Post-Its)
7. User Stories (Table)

**Section I**

**Introduction**

1. Team 3
   1. Camerin Figueroa
   2. Christopher Fioti
   3. Chris Danyluk



Our Business is Hardware Online. We are a home-depot like store, providing building supplies, electrical supplies, plumbing supplies, and other hardware requirements for DIY projects. However, we are focused more closely on tools, such as power drills, saws, hammers, wedges, etc.

The goal of this project is to design a website and system for Hardware Online. It allows for the creation of user profiles for customers, employees, administrators, and stakeholders. It will be an inventory tracking system, and allow for a variety of functions on the website to allow for all users to interact with our systems in some forms.

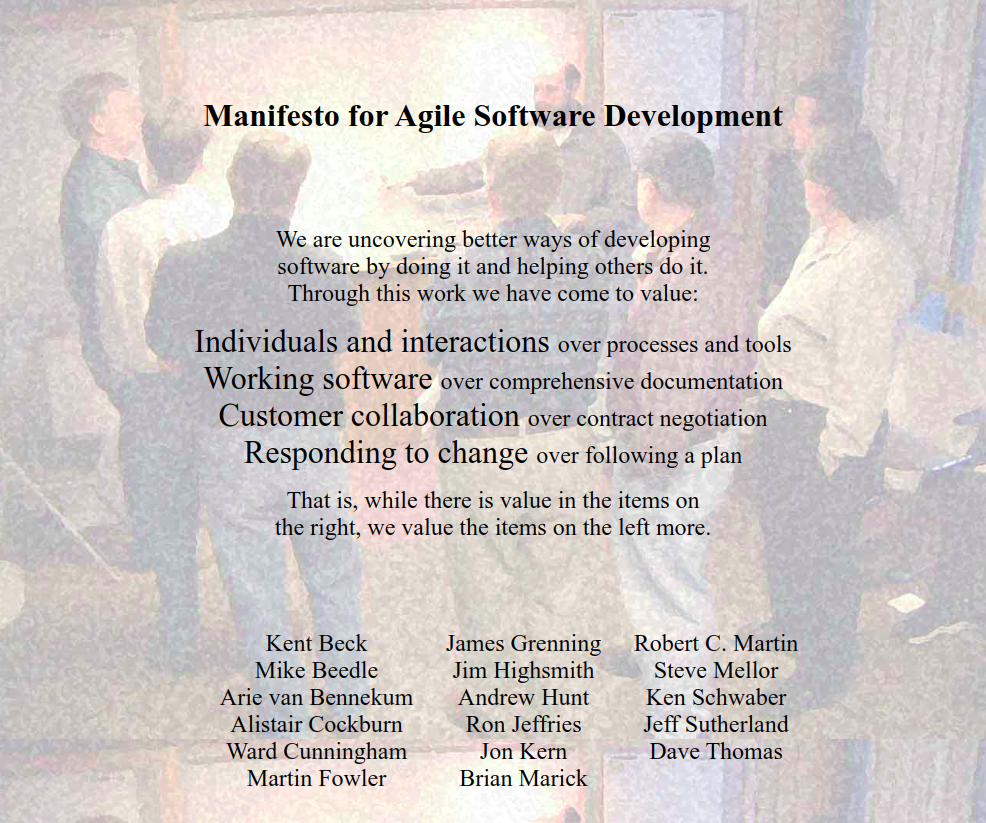
Our userbase will be customers looking to purchase DIY materials, tools, etc., as well as employees looking to update, track, add, and remove stock, and administrators to allow them to track employees' logs, create employee accounts, and act on administrative needs. Stakeholders will also be able to use our system to view the company and ensure their money is not being wasted.

Our stakeholders include:

* + - 1. Page Lumber
      2. Klein Tools
      3. Makita
      4. Bostitch
      5. Lutron

And more

* Project Development
  + The Agile Manifesto



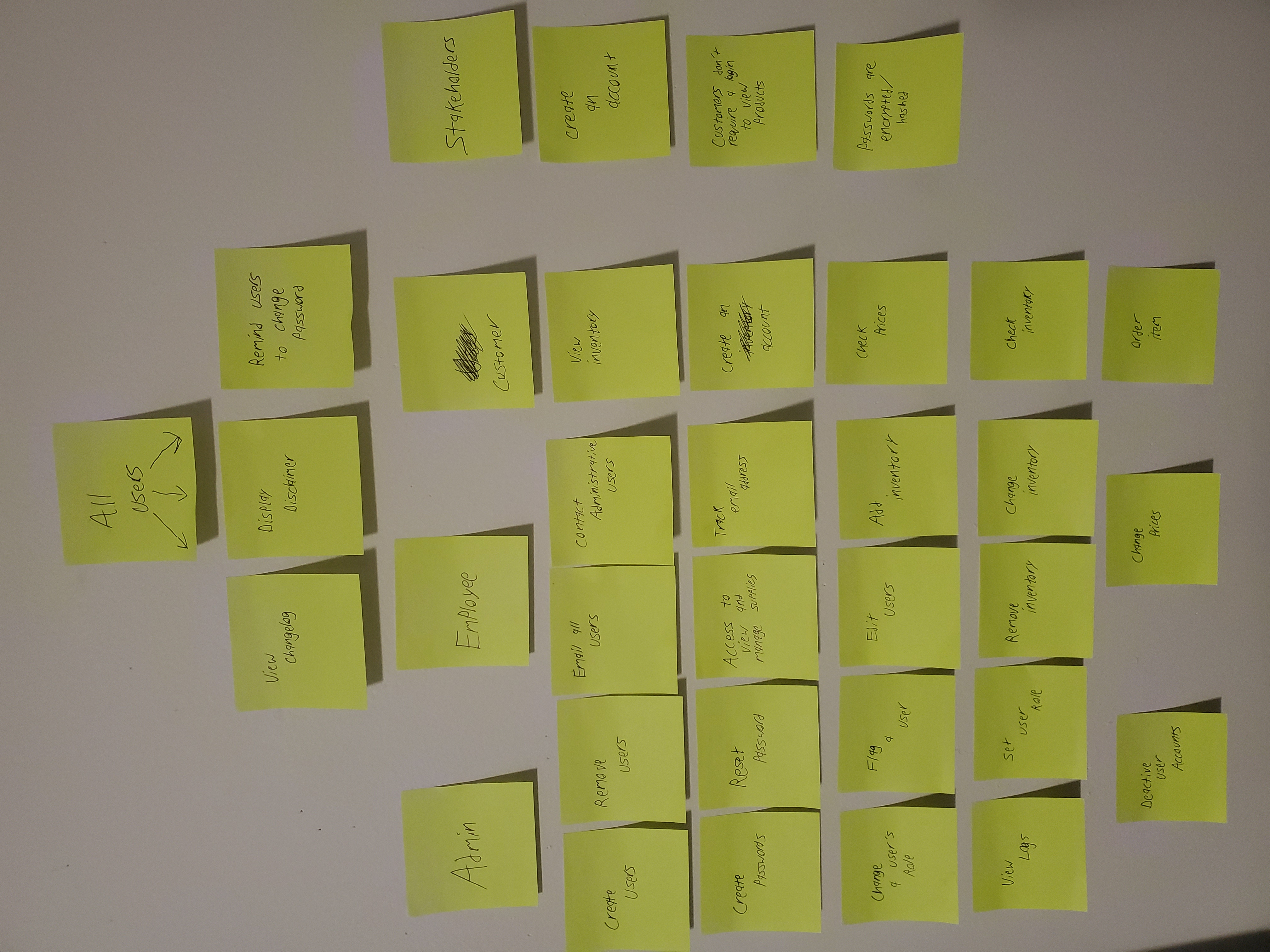
* + What Agile Principles will your team be using
    - Working software is the primary measure of progress.
    - Continuous attention to technical excellence and good design enhances agility.
    - Simplicity–the art of maximizing the amount of work not done–is essential.
    - The best architectures, requirements, and designs emerge from self-organizing teams.
    - The best architectures, requirements, and designs emerge from self-organizing teams.
  + Key Project steps (ex: requirements gathering, feature design, Unit testing,  
     etc.) – describe what they are
    - Gather Requirements and Planning
      * At this step we are tasked with collecting requirements and input from users of the system.
    - Define Requirements
      * At this stage, requirements will be defined and solidified in order for the team to understand how they will be working together.
    - Feature Designing
      * A design plan is drafted up at this stage, and specifications of the system are decided at this stage.
      * Deciding what software will be used, what hardware will be needed that can support our system.
    - Building
      * At this stage, the project is actually built.
      * Scrums will be used to help keep track and organize the order when parts of the project will be built.
    - Unit Testing
      * At this stage, the software is tested for issues, bugs, or anything that needs to be fixed before release.
      * Automated testing or fuzzing will be used at this stage to help find problems with the system and to even determine security risks.
      * Beta testers can be used here to watch the system in it’s natural habitat.
    - Deployment
      * At this stage the system is ready for release and is open to the users that it was meant for.

**Section III**

**Requirements and Software Used**

For this project we will be using a combination of HTML, CSS, JavaScript, PHP, and SQL on a Windows OS. Specifically, we will be developing the HTML and PHP in Visual Studio Code, while our SQL will be produced using MySQL. In addition, the PHP server will be hosted through Abyss, another Microsoft product. This will be a client/server application and will make use of HTTP requests.

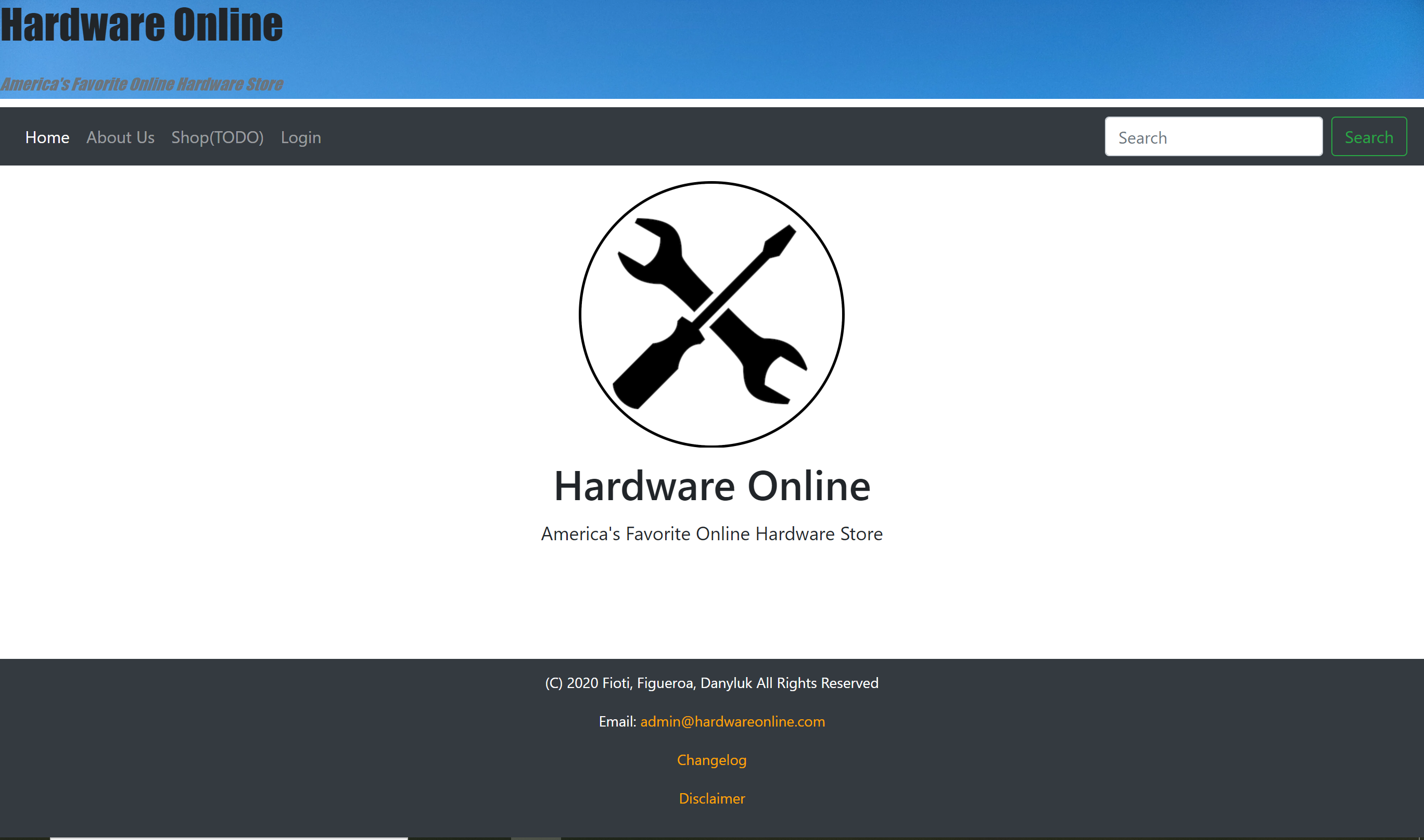
User stories are within the table on the next page, as well as an image of our stories on post-its.



|  |  |  |
| --- | --- | --- |
| Individual | I want to... | So that I can... |
| Admin | Create Users | Add people to the system |
| Admin | Remove Users | Remove people from the system |
| Admin | Create Passwords | secure the system |
| Admin | Reset passwords | Allow employees who forgot their password back in |
| Admin | Change a user’s role | Promote and demote people |
| Admin | Flag a user | Review the profile later |
| Admin | View Logs | Monitor user activity, and diagnose issues. |
| Admin | Set User Role | Allocate role to appropriate users |
| Admin | Deactivate User Accounts | Prevent malicious users from accessing the website. |
| Employee | Email all users | Send out Promotional Emails |
| Employee | Contact Administrator Users | Contact Administrators to notify them of issues, questions, etc. |
| Employee | Access to view and manage suppliers | Add, edit, deactivate, or view suppliers and their info. |
| Employee | Track Email Address | Find a user’s email address in order to contact them |
| Employee | Edit Users | Can edit passwords and usernames |
| Employee | Add Inventory | Update when we receive new products |
| Employee | Remove Inventory | Update when we no longer sell products |
| Employee | Change Inventory | Update certain products |
| Employee | Change prices | Update prices for items |
| Customer | View Inventory | Buy things |
| Customer | Create an Account | Keep my information loaded for my next purchase |
| Customer | Check Prices | See what is the lowest cost |
| Customer | Check Inventory | See what is in stock |
| Customer | Order Item | Purchase Item |
| Stakeholder | Create an account | See what is happening in the business |
| Stakeholder | Customers don’t need a login to view products | Easier access to what we have without needing to take time to create account. |
| Stakeholder | Passwords are encrypted/hashed | Prevent view of passwords in database breaches. |
| All Users | View Changelog | Understand what has changed on the website and to see that it is being actively worked on. |
| All Users | Display Disclaimer | In order to cover the company in case someone wants to sue us. |
| All Users | Remind user to change password | To keep the website secure |

**Section IV**

**Design**



Software Used

We will be using HTML, CSS, JavaScript, PHP, and MySQL to achieve our desired end result

* HTML
* CSS
* JavaScript
* PHP
* SQL

Tables

Users

|  |  |  |
| --- | --- | --- |
| Field | Attribute | Definition |
| username | TEXT, UNIQUE | User’s ID for login |
| UID | PRIMARY\_KEY, UNIQUE, INT | Account ID |
| pwHash | TEXT | Password Hash Code |
| hashType | TEXT | Hashing algorithm used with this user’s hash. This is useful so we can update our password hashes later on to more secure algorithms. |
| rankID | ENUM | Connects to the Permissions table to determine what permissions this user has |

Product

|  |  |  |
| --- | --- | --- |
| Field | Attribute | Definition |
| vendorID | INT | Unique ID for vendor |
| productID | UNIQUE, PRIMARY\_KEY, TEXT | Unique Product ID Number |
| model | TEXT | Model number of product |
| product | TEXT | Name of product |
| stock | INT | Quantity of product in stock |
| desc | TEXT | Product Description |
| price | FLOAT | Price of product |

Vendors

|  |  |  |
| --- | --- | --- |
| Field | Attribute | Definition |
| vendorID | PRIMARY\_KEY, UNIQUE, INT | Unique ID for vendor |
| name | TEXT | Name of vendor |
| address | TEXT | Address of vendor |
| phone | INT | Phone number for contact of vendor |

Permissions

|  |  |  |
| --- | --- | --- |
| Field | Attribute | Definition |
| rankID | ENUM | Id for rank |
| permission | TEXT | Identification for the permission |
| UID | INT, AUTO\_INCREMENT, PRIMARY\_KEY | Unique ID for each permission row |