Abstract

As the world population grows, food vendors need to be increasingly efficient in the distribution of their products. Products with short shelf-lives need to be distributed as quickly as possible in order to maximize potency. Existing technologies allowing the sale of goods from peer to peer lack key features to optimize usage for perishable goods. GreenPlace provides some of the essential features necessary to distributing perishable goods and managing those orders in an effective manner.
I. INTRODUCTION

Many existing technologies allow the peer to peer sale of goods. Though acceptable for most products, these technologies are difficult for use by vendors of perishable goods. GreenPlace has a simplified design, easily accessible, and provides powerful search methods, communication methods, and a product management system all to facilitate peer to peer transactions between vendors and consumers of products with limited shelf-lives.

II. PROBLEMS WITH EXISTING TECHNOLOGIES

Existing technologies like Craigslist function well for the sale of products that do not expire. These technologies lack the ability to search products by when the vendor is available to conduct a sale. If the fruits on a tree are going to be harvested after a certain date, the farmer would have to add information inside the text post stating when the apples are available. Similarly, users interested in purchasing these apples must individually view each post’s content rather than being able to filter by product availability. Filtering search results based on availability will simplify the purchasing process of products with limited shelf-lives.

When searching Craigslist for a product, it can be difficult to filter results based on location. If you wish to search for a product within a 10 mile radius of your current location, you are forced to switch to map view and individually click on each listing to view the title of the link. Craigslist is also separated into separate subdomains to represent each local area. If searching with a radius extending outside of the local area, the search must be performed under each Craigslist sub-domain. Third party search engines such as www.searchtempest.com exist to solely provide the ability to search all Craigslist for products within a certain distance of your location. Users would benefit through having these map features integrated into a single service.

If a vendor wishes to sell a large quantity of products using existing technologies, they must either post a single listing representing the sale of all products or multiple listings each representing a fraction of the total quantity available. When the vendor is able to sell some of their products, they must manually update the listing to reflect the new quantity available. An ideal marketplace would automatically update the user’s listing as products are sold.

III. TARGET AUDIENCE

GreenPlace is intended for use by primarily perishable good producers and consumers: farmers, grocery stores, merchants, and the general public. Producers of local non perishable products could also benefit from using the site as a means of distributing their goods as GreenPlace flexible in design in order to allow the sale of all legal products.

IV. DESIGN AND IMPLEMENTATION

A. Operating Environment

GreenPlace is accessible from an HTTP server hosted on the cloud using Amazon Web Services (AWS). This allows the software to be accessed by anyone with an internet connection without needing to manually install software locally. Most users are currently aware of how to connect to a website and navigate it with minimal instruction. This operating environment allows GreenPlace to reach a wide audience of users with both internet access and an installed web browser. Node package Forever is used to ensure that the server is restarted in the event of an unexpected crash.

B. Architectural Pattern

GreenPlace follows a model, view, controller architectural pattern. This allows the model and view components to be reused within the software. As shown in Figure 1, the user interacts with the Jade generated view in order to notify the controller of what actions to perform. The controller uses Express to act upon the AJAX requests sent as a result of user pressing buttons or clicking hyperlinks. Depending on the uploaded data, the controller can decide whether to update the PostgreSQL backend using Sequelize or display a new view to the user.

V. FEATURES

A. Tables

Users have the ability to sort each column by ascending or descending values and filter rows based on provided text on any page containing a table (Figures 10, 14, 15, 17, and 19).

B. Authentication

When a user visits GreenPlace, they can choose to log into an existing account, create a new account, or skip authentication and access a limited portion of the site. Creating a new account is as simple as providing a valid email address, username, and password (Figure 2). The provided email address must be validly formatted. Neither the username or email address can be in use by an existing user as they are both used for identifying the user account. Authenticating with an existing account is as simple as typing in the username or email and password associated with the user. Once a user has authenticated, their session is recorded in order to prevent the need for authenticating multiple times during a single usage session.

If authentication is skipped, the user can still access GreenPlace by browsing and searching for posted products. If an unauthenticated user attempts to access any site feature other than browsing, searching, or viewing a product, a prompt for authentication will occur as all other features require an authenticated account for use.

If a user forgets their password, their password can be reset by following the password reset link at the bottom of the authentication page (Figure 2). The user can then input their username or email address and generate a reset url sent to their email address (Figures 4 and 5). This url is only valid for thirty minutes, during which the user can set a new password to be associated with their account. If the user attempts to use the reset url outside the reset time window, their password will not be changed.
Note that passwords are hashed using MD5, compared, and stored in the database. Once a password has been stored it is never returned to plain text.

C. Searching for Products

Users can find and view products on GreenPlace by using one of three methods: browsing, searching, or sharing. All of these methods are designed so that first time consumers view the benefits of becoming a regular user.

Browsing for a product is done by selecting a location on an embedded map (Figure 6). Products are placed on the map as pins as the user pans to new locations or decreases the map scale. When a pin is clicked a box expands showing the product name, description, and vendor. The name and vendor are both hyperlinks that open tabs to the product and vendor pages respectively.

More specific products can be found through performing a search. Searching is done through initially selecting a single location on a map and specifying a radius around that area where the product must exist (Figure 7). After specifying a location and radius the user must provide keywords to look for in the product name and description. Additional criteria which can be provided include time periods in which the vendor has marked as available for sale, the minimum quantity of units available, and a minimum or maximum price for the desired quantity (Figures 8 and 9). Once the search criteria is filled and submitted, the user is shown all results on a table containing product name, vendor, quantity, and price (Figure 10).

It is worth noting that when searching, the provided availability is assumed to be provided in the respective timezone in which the product is being sold. That is to say that if the user is located in Los Angeles and searching for products is located in New York, the provided availabilities are not converted from Pacific Standard Time to Eastern Standard Time, but rather assumed to be provided in Eastern Standard Time. This is to prevent the need for users to manually convert provided availabilities when searching for products in other time zones.

D. Purchasing

Once a consumer has found a product they wish to purchase, a button on the product page that when clicked redirects to an order form (Figure 12). On this form the consumer specifies the desired quantity, price, time of sale, and optional message to send to the vendor. Once an order has been placed the consumer cannot edit the order. If the order was placed by mistake or is no longer valid, the consumer can “Rescind” in order to delete the order (Figure 15).

E. Product Management

Vendors submit their products to GreenPlace by providing the product name, description, image, total quantity, pricing, availability, and pickup location (Figure 11). The product description can be formatted using markdown to include hyperlinked, bolded, italicized, underlined, and sized text. These fields exist to organize the information for consumers browsing GreenPlace. After submission, all product fields can be modified in case a change needs to be made (Figure 13). Before finalizing changes to an existing or submitting a new product, the vendor is shown a page where the post can be previewed. This feature is useful to ensure that the markdown formatted text appears as intended before submission.

If a vendor uses other advertising methods to distribute products, they can include a hyperlink to the product page to direct traffic toward their post. The hyperlink allows possible consumers to view the product without authentication. The product page includes information regarding the name, description, initial quantity, pricing, availability, pickup location, and remaining quantity (Figure 12).

Note that when setting the availability of a product the times will be set to the respective timezone the product is sold in.

All products listed on the marketplace by the authenticated user can be viewed from they “My Items” page (Figure 14). This page provides functionality to view, edit, or delete products currently listed on the marketplace for sale.

Once the product is submitted to the marketplace, the vendor is responsible for managing orders placed by other users. The vendor can accept or decline the order signifying whether the sale will be completed on the requested date for the provided quantity and price (Figure 17). If the order is accepted, the remaining quantity will automatically update relative to the quantity sold in the accepted order. The consumer can see the acceptance status of their order set by the vendor. This establishes a standard means of communication regarding the sale and purchases of products along with the built in messaging system (Figure 19). When changes are made to a product, its orders are not modified in any way. Accepted orders will remain accepted even if the total quantity is reduced below the number purchased in the accepted order.

Once an order has been placed it’s acceptance status is set to “Pending”. The Orders that are pending can be rescinded by the consumer and accepted or declined from the vendor only if the order’s status is “Pending”. Once a change has been made by the consumer or vendor the status cannot be changed. Both the vendor and consumer will receive an email indicating a new order has been placed or a change in acceptance status. Users are encouraged to share more specific information regarding the purchase such as contact information and directions once an order has been placed.

Note that an order cannot be placed on a product owned by the same user.

F. Interfacing with Other Users

Each account has a customizable profile (Figure 18). This profile page allows other users to view public information posted by the account owner. The profile page can include a profile picture and text formatted with markdown.
Users can message one another for any reason by clicking a button at the bottom of the user’s profile page (Figure 18). A message must contain both a subject and description (Figures 20, 21, 22). All users have an inbox where messages can be viewed and a responses can be sent to the user that sent the message (Figure 19). Once a message is sent, the recipient will receive an email notification indicating that a message has been received. This promotes users to return to GreenPlace and to communicate without soliciting their contact information.

G. Site Comments / Feedback

As GreenPlace is in the early development process, feedback from users is important in order to determine features and modifications to consider making. A "Give Feedback" hyperlink at the bottom of each page allowing users to send messages to my personal GreenPlace account would be sufficient for gathering feedback from users.

VI. PROBLEMS ENCOUNTERED

A. Database Change

Originally, GreenPlace was constructed using a combination of MongoDB and Mongoose for the model component. These are relatively new technologies that allow data to be stored in documents rather than the commonly used relational tables. It was a pitfall to assume that newer technologies are always better than the old. With MongoDB, aggregation queries were difficult and impractical to write as they require a lot of boilerplate code. An example of this is where MongoDB in its current state does not allow performing a query to determine the distance from a location and verifying that it’s values are within inside of an array in a single query. This requires two queries to perform a single action and then only displaying the results inside of both queries. This query became resource intensive and showed the weaknesses in the application of MongoDB.

In the second quarter, MongoDB and Mongoose were completely replaced by PostgreSQL and Sequelize. By switching to PostgreSQL, aggregation queries became easier to write and less resource intensive.

VII. FUTURE WORK

A. Subscriptions

Users should be able to subscribe to specific vendors in order to receive email notifications when the vendor places an item for sale on the marketplace. This would encourage users to return GreenPlace to conduct business.

B. Flagging Posts

As with all online services, some users post inappropriate or illegal content. A method should exist to flag these users and products for removal.

C. View Upcoming Transactions

When a vendor has a large number of orders for a product, it would be beneficial to display them on a calendar. This would allow the vendor to view their schedule easily without having to navigate through each order. It may also be beneficial to export this data to an existing calendar service such as Google Calendar.

D. Sharing Contact Information

After a vendor accepts an order placed by the consumer, both users should be able to access private information regarding each other such as contact information and first and last name. This will allow both users to communicate outside of GreenPlace before completing a transaction.

E. Usage Terms and Privacy Policy

Links to usage terms and the privacy policy should be visible at the bottom of each page in order to inform the user of how the site should be used legally and what is being done with user provided data.

VIII. CONCLUSIONS

GreenPlace is a peer to peer marketplace designed for the sale of perishable goods. The software allows users to find products within a radius from a provided location, communicate with other users through an internal messaging service, and manage their product information and orders in a single location. GreenPlace is a digital farmers market with the tools essential to selling perishable goods.
### IX. APPENDIX

#### A. Technologies Used

<table>
<thead>
<tr>
<th>Technology</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deploy.io</td>
<td>Tool for automatically deploying Server hosting GreenPlace</td>
</tr>
<tr>
<td>Amazon Web Services</td>
<td>Server hosting GreenPlace</td>
</tr>
<tr>
<td>NodeJS</td>
<td>JavaScript library used to run the HTTP server</td>
</tr>
<tr>
<td>Forever</td>
<td>Node package used to keep a node process running without interruptions</td>
</tr>
<tr>
<td>Jade</td>
<td>Node template engine used to render HTML files</td>
</tr>
<tr>
<td>Passport</td>
<td>Node package used for authenticating users and managing usage sessions</td>
</tr>
<tr>
<td>MongoDB</td>
<td>Document database completely replaced by PostgreSQL in the final version of the website</td>
</tr>
<tr>
<td>Mongoose</td>
<td>Node package used to communicate to MongoDB. Completely replaced by Sequelize in the final version of the website</td>
</tr>
<tr>
<td>PostgreSQL</td>
<td>Relational database used to user information</td>
</tr>
<tr>
<td>Sequelize</td>
<td>Object relational mapping used for communicating with PostgreSQL database using JavaScript</td>
</tr>
<tr>
<td>Busboy</td>
<td>Node package used to parse multipart form-data POSTed by users</td>
</tr>
<tr>
<td>Body-Parser</td>
<td>Node package used to parse non multipart form-data POSTed by users</td>
</tr>
<tr>
<td>Underscore</td>
<td>JavaScript utility package</td>
</tr>
<tr>
<td>Lodash</td>
<td>JavaScript utility package</td>
</tr>
<tr>
<td>Crypto-Js</td>
<td>Node package used for hashing user passwords</td>
</tr>
<tr>
<td>Nodemailer</td>
<td>Node package used to send templated emails</td>
</tr>
<tr>
<td>Email-Templates</td>
<td>Node package used to template emails</td>
</tr>
<tr>
<td>Moment</td>
<td>Node package used for manipulating JavaScript Date objects</td>
</tr>
<tr>
<td>Moment-Timezone</td>
<td>Node package for converting moments from one timezone to another</td>
</tr>
<tr>
<td>Tzwhere</td>
<td>Node package to determine the timezone of gps coordinates</td>
</tr>
<tr>
<td>Marked</td>
<td>Node package used to parse markdown text into HTML</td>
</tr>
<tr>
<td>Express</td>
<td>Handle user AJAX requests</td>
</tr>
<tr>
<td>Express-Session</td>
<td>Node package used to handle usage sessions without reauthenticating with each page load</td>
</tr>
<tr>
<td>Express-Validator</td>
<td>Node package used to validate user input and sanitize uploaded data</td>
</tr>
<tr>
<td>PureCSS</td>
<td>CSS library used to style web pages</td>
</tr>
<tr>
<td>JQuery</td>
<td>JavaScript utility library</td>
</tr>
<tr>
<td>JQuery DatePicker</td>
<td>JavaScript library allowing the selection of multiple dates on a calendar</td>
</tr>
<tr>
<td>Isotope</td>
<td>JavaScript library used to allow sorting of HTML tables</td>
</tr>
<tr>
<td>Google Maps JavaScript v3 API</td>
<td>JavaScript library used to allow user map interaction</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource URL</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://dploy.io/">http://dploy.io/</a></td>
<td></td>
</tr>
<tr>
<td>aws.amazon.com/free</td>
<td></td>
</tr>
<tr>
<td><a href="https://nodejs.org/">https://nodejs.org/</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://github.com/everys/everys">https://github.com/everys/everys</a></td>
<td></td>
</tr>
<tr>
<td><a href="http://jade-lang.com/">http://jade-lang.com/</a></td>
<td></td>
</tr>
<tr>
<td><a href="http://passportjs.org/">http://passportjs.org/</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.mongodb.org/">https://www.mongodb.org/</a></td>
<td></td>
</tr>
<tr>
<td><a href="http://mongoosejs.com/">http://mongoosejs.com/</a></td>
<td></td>
</tr>
<tr>
<td><a href="http://www.postgresql.org/">http://www.postgresql.org/</a></td>
<td></td>
</tr>
<tr>
<td><a href="http://docs.sequelizejs.com">http://docs.sequelizejs.com</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/busboy">https://www.npmjs.com/package/busboy</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/body-parser">https://www.npmjs.com/package/body-parser</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/underscore">https://www.npmjs.com/package/underscore</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/lodash">https://www.npmjs.com/package/lodash</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/crypto-js/">https://www.npmjs.com/package/crypto-js/</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/nodemailer/">https://www.npmjs.com/package/nodemailer/</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/email-templates/">https://www.npmjs.com/package/email-templates/</a></td>
<td></td>
</tr>
<tr>
<td><a href="http://momentjs.com/">http://momentjs.com/</a></td>
<td></td>
</tr>
<tr>
<td><a href="http://momentjs.com/timezone/">http://momentjs.com/timezone/</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/tzwhere/">https://www.npmjs.com/package/tzwhere/</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/marked">https://www.npmjs.com/package/marked</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/express">https://www.npmjs.com/package/express</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://www.npmjs.com/package/express-validator/">https://www.npmjs.com/package/express-validator/</a></td>
<td></td>
</tr>
</tbody>
</table>
B. Data Dictionary

1. GreenPlace | SaaS created to facilitate peer to peer sales of perishable goods
2. Peer to Peer | Direct sale and purchase of a product from one user to another
3. Product | Anything that is produced and intended for sale to another party
4. Vendor | A person or company that sells products to another party
5. Consumer | A person or company that purchases products
6. User | A person or company that uses the website [vendor, consumer]
7. Perishable Good | A product which has a fixed lifespan before it is no longer valid
8. Shelf-Life | The amount of time a product can be left unused before it is no longer able to function properly
9. Potency | The effectiveness of a product to perform its intended purpose
10. Technology | Other marketplaces intended for peer to peer sales: Craigslist, Ebay, Amazon, etc.
11. Account | Digital information that represents the user when using GreenPlace
12. Usage Session | Period of time from which a user uses GreenPlace
13. Unit | The unit used to quantify a product: litres, kilograms, and each individual product aka unit
14. Acceptance Status | Whether an order has been rescinded, accepted, declined, or pending approval
15. Markdown | Language that allows formatting of text
16. SaaS | Software as a service
17. Availability | The time which a sale can be conducted

C. Figures

Fig. 1: MVC Diagram
Fig. 2: Authentication
This is the default screen to an unauthenticated user. From this page the user can login to an existing or create a new account, or browse the marketplace without authenticating.

![Authentication Screen]

Fig. 3: Change Password
Accessed by clicking the “Forgot Password” hyperlink on Figure 2. Users can change their password by filling in their username or email, old password, and new password.

![Change Password Form]

Fig. 4: Reset Password
Allows users to reset their password by providing either their username or email address. It is accessed by clicking the “Reset Password” hyperlink on Figure 2. After submission, an email is sent to the user’s email address depicted by Figure 5.

![Reset Password Form]
Fig. 5: Reset Password Email

Represents a reset password attempt generated by Figure 4. The user can click the reset url in order to change their password.

Fig. 6: Browsing the Marketplace

Default page for an authenticated user. The user is displayed all items for sale in their area. Users can center the map on their location by pressing the “Go to Current Location” button.
**Fig. 7: Search Location**

Allows users to set their location and radius to find items in.

**Fig. 8: Search - Selecting Availability**

Select one or more days at once to be added to the search criteria.
Fig. 9: Search - Selected Availability to Add
Set the time period for the selected days to add to the search criteria.

Fig. 10: Search - Results
Results when searching for “Avocado” near San Luis Obispo.
Fig. 11: New Item

Template to be filled by the user to represent the product for sale.
Fig. 12: Item

Represents a product for sale.

**Milk**

18 litres remaining of 20 litres

Sold by: **Farmer**

Milk for sale!

Come and buy a few liters!

Initial listing quantity: 20 litre
Selling for $2 per 1 litre ($2 per litre)

Available on the following dates:

<table>
<thead>
<tr>
<th>Su</th>
<th>Mo</th>
<th>Tu</th>
<th>We</th>
<th>Th</th>
<th>Fr</th>
<th>Sa</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
</tr>
<tr>
<td>28</td>
<td>29</td>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Pickup Location
Fig. 13: Edit Item
Template with prefilled information to streamline making changes to an existing item.

Fig. 14: My Items
Table displaying all products for sale by the authenticated user available for purchase.
Fig. 15: My Orders

Orders that have been placed by the logged in user on products interested in purchasing.

![Table of orders with vendors and items]

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Item Name</th>
<th>Quantity</th>
<th>Units</th>
<th>Price</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Higuera40</td>
<td>Organic Herbs</td>
<td>0.3</td>
<td>kg</td>
<td>5</td>
<td>Pending</td>
</tr>
<tr>
<td>Farmer</td>
<td>Milk</td>
<td>2</td>
<td>litre</td>
<td>4</td>
<td>Accepted</td>
</tr>
</tbody>
</table>

Fig. 16: View Order

Detailed information regarding a potential product purchase. In this case, the order was declined by the vendor.

Order for **Farmer's Milk**

Placed by: Businessman

Status: Declined

Seeks to purchase **15 litres** of Milk for **$10**

Order message:

I hope we can do business together

Sale time for **2015-06-11T09:00:00-07:00**

Pickup Location
Fig. 17: Customer Orders
Orders that have been placed on one of the logged in user’s products.

**Viewing orders for Milk**

<table>
<thead>
<tr>
<th>Placed By</th>
<th>Quantity</th>
<th>Units</th>
<th>Price</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nittylico</td>
<td>2</td>
<td>litre</td>
<td>4</td>
<td>Accepted</td>
</tr>
<tr>
<td>Businessman</td>
<td>15</td>
<td>litre</td>
<td>10</td>
<td>Declined</td>
</tr>
</tbody>
</table>

18 litres remaining of 20 litres

Fig. 18: Profile
A customizeable profile displaying relevant merchant information about the user. Changes can be made by the authenticated user by clicking the pencil.

**Farmer’s Profile**

**Items I sell frequently**
- Milk
- Eggs
Fig. 19: Messages Received
Displays messages sent from other users to that which is currently authenticated.

Inbox - Received Messages

<table>
<thead>
<tr>
<th>Sent By</th>
<th>Time Sent</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buyer</td>
<td>05/22/2015 7:14 PM</td>
<td>Hey there</td>
</tr>
</tbody>
</table>

Fig. 20: View Message
A message sent to the currently authenticated user.

Hey there
Sent By: Buyer at Fri May 22 2015 19:14:25 GMT+0000
I want to buy stuff from you!
;

Reply

Fig. 21: Compose Message
Compose a message to be sent to another user using markdown formatted text.

Compose message
Sending to: Buyer

Note: section uses Markdown, a webpage formatter. You can learn the basics here.
Looking forward to make the sale
![moneymoney](http://mewarnoi.us/images/200g8-money-clip-art.png)

Preview Message
Fig. 22: Preview Message

Preview message to verify properly formatted text before sending.

Ok!

Looking forward to make the sale

Send Message

Fig. 23: Send Feedback

Send a message to the system administrators regarding bugs, feature requests, and opinions.

Send Feedback

Thank you for taking the time to help improve the site.
Please address any bugs, feature requests, and any other site related inquiries here.

Sending to: niftynico

Title

Note: section uses Markdown, a webpage formatter. You can learn the basics here

Message

Preview Message

D. Special Thanks

- Adam Currie for creating the query using MongoDB that searches the database for matching products given keywords, location, availability, and minimum or maximum quantity.
- Justin Fujikawa for adding functionality to allow users to edit their description using markdown text.
- Dr. Clark Savage Turner and Dr. Phillip Nico for guidance during the project.