Carbon Footprint Web Portal / Application

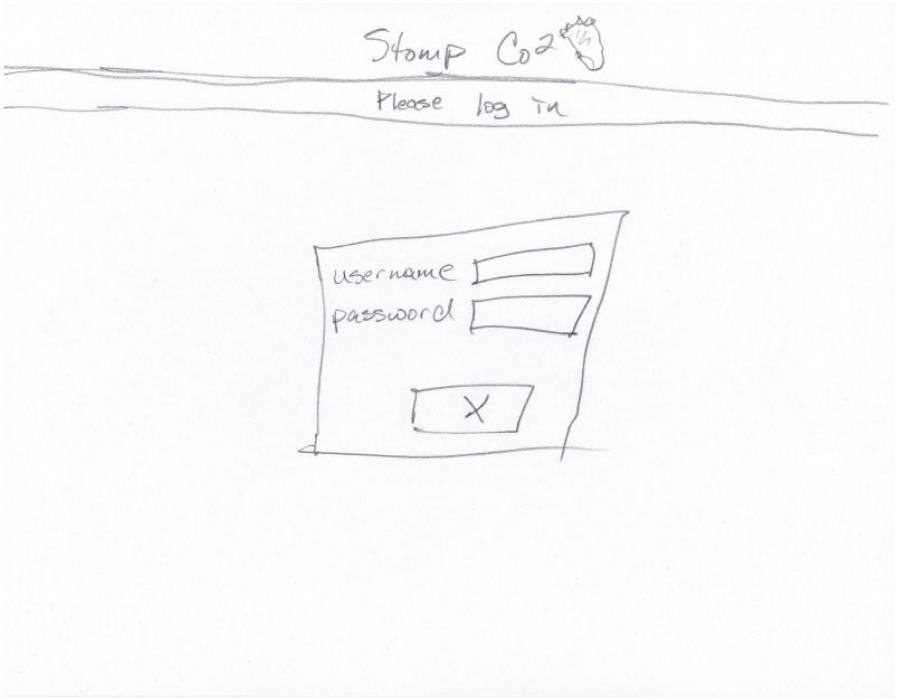
- HW 2: Revised Requirements and Paper Prototypes

CS 361 Group 2 Project
10/26/19

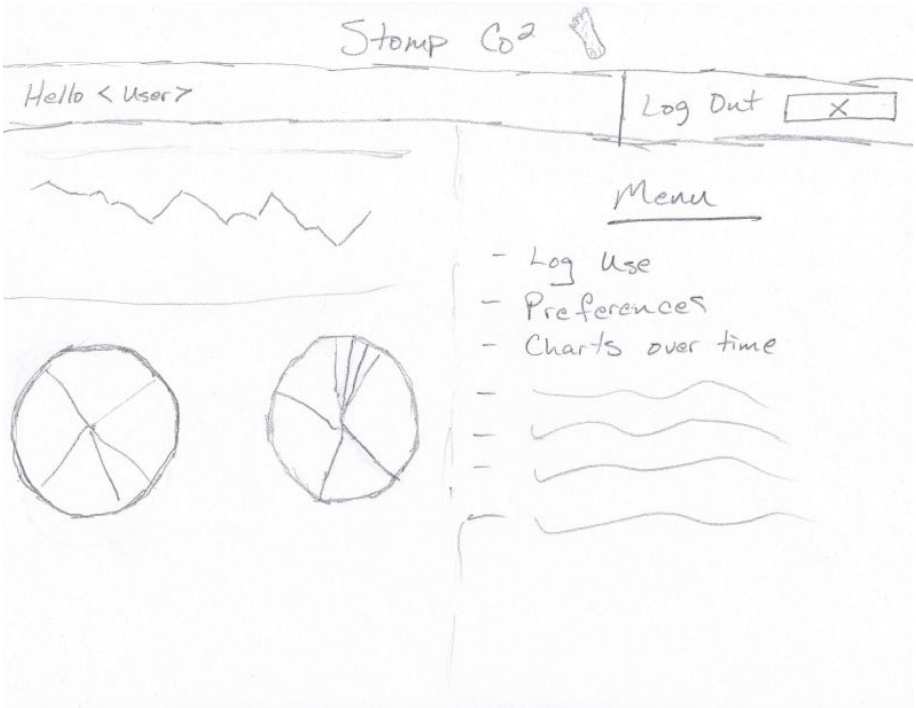
- Alex Densmore
- Anousha Farshid
- Adam Wright
- Ken Wyckoff

Paper Prototypes:

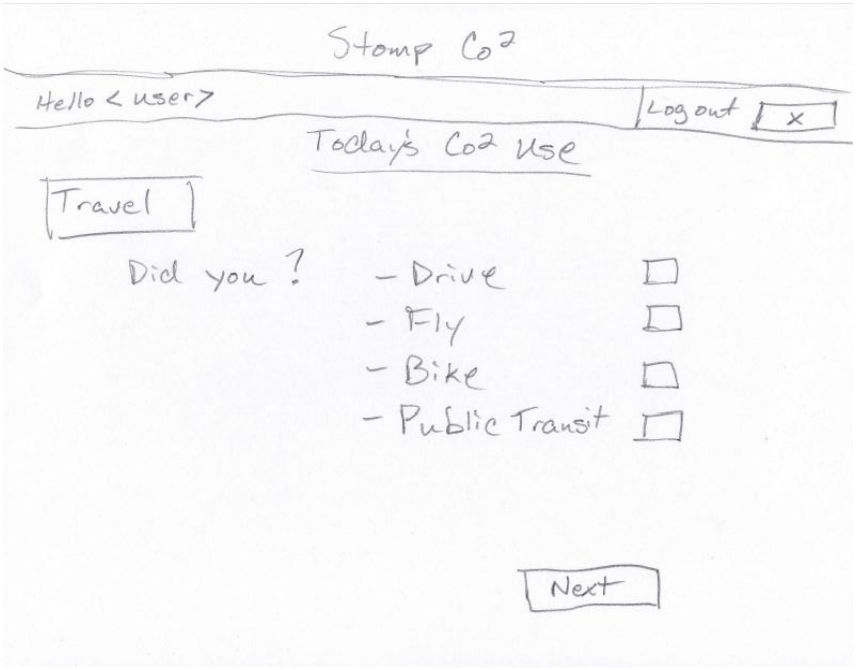
Login screen



Dashboard

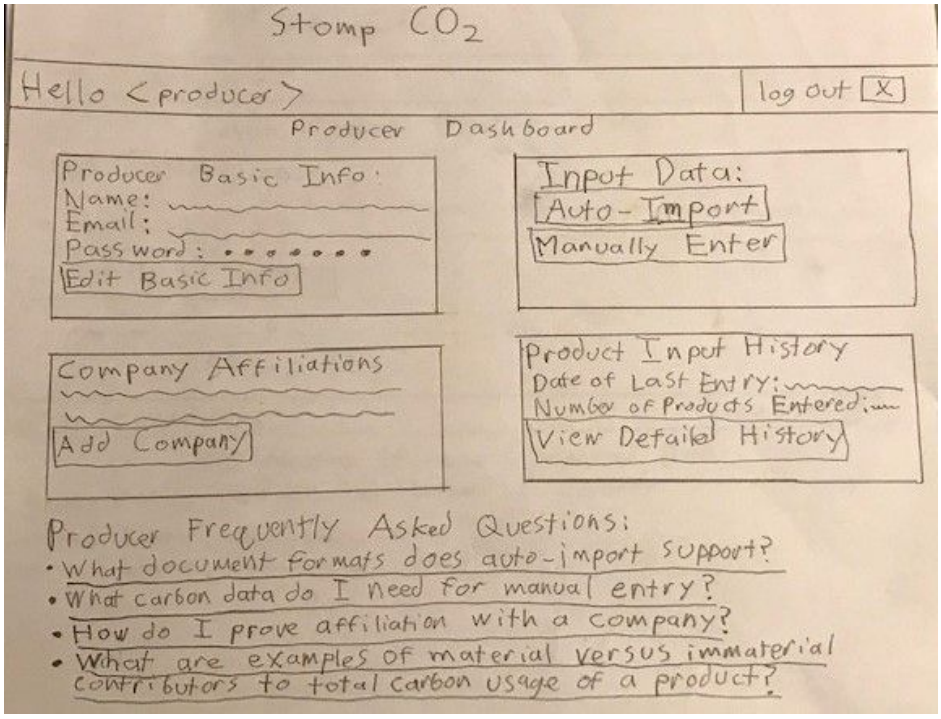


Manual Entry Decision Tree Page

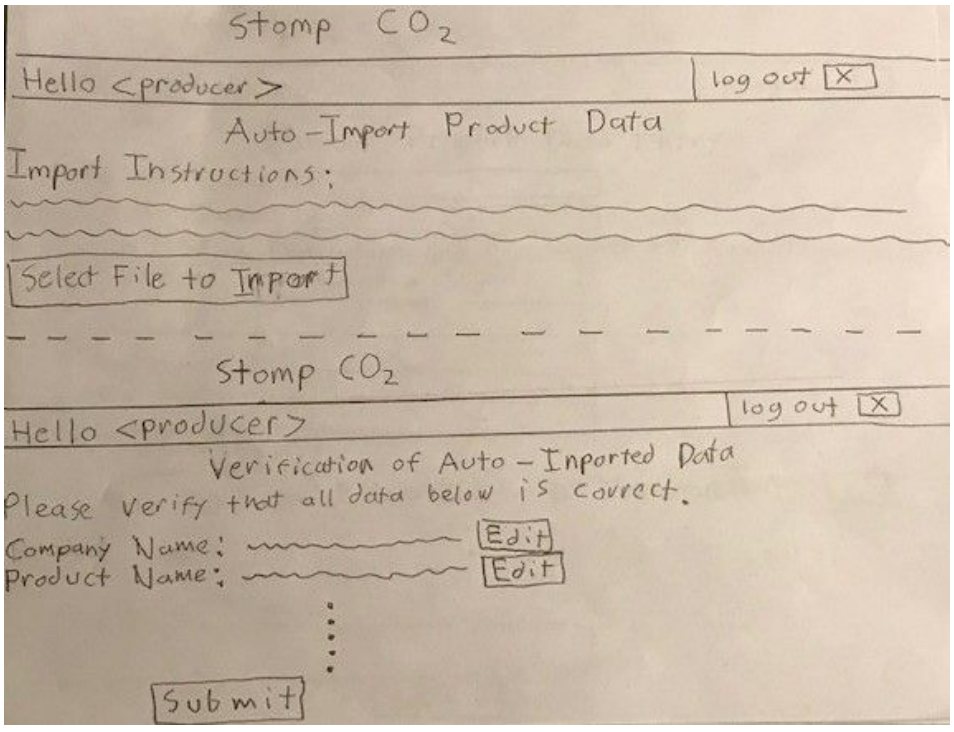


User Preferences

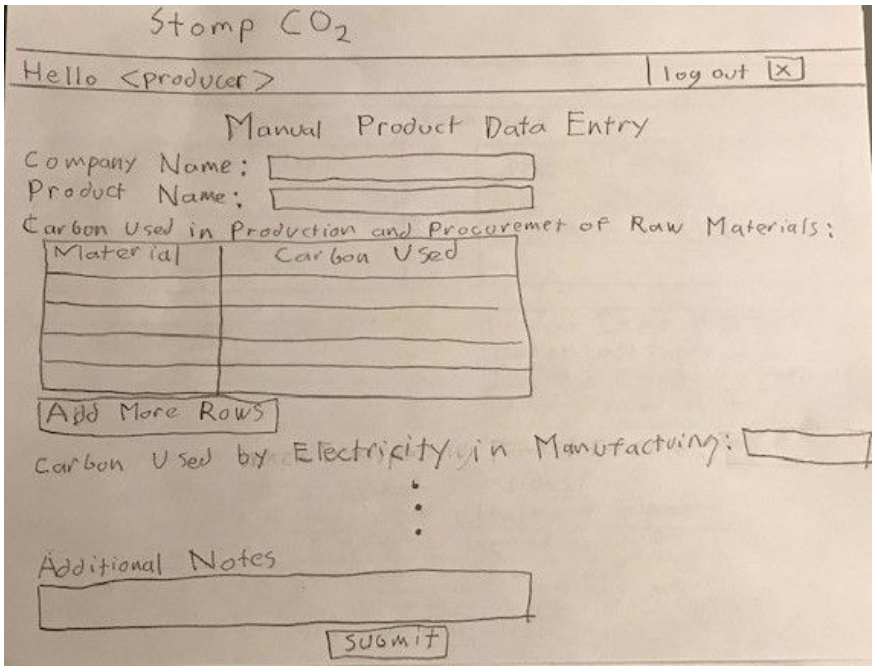
Use Case #2 - Producer Portal



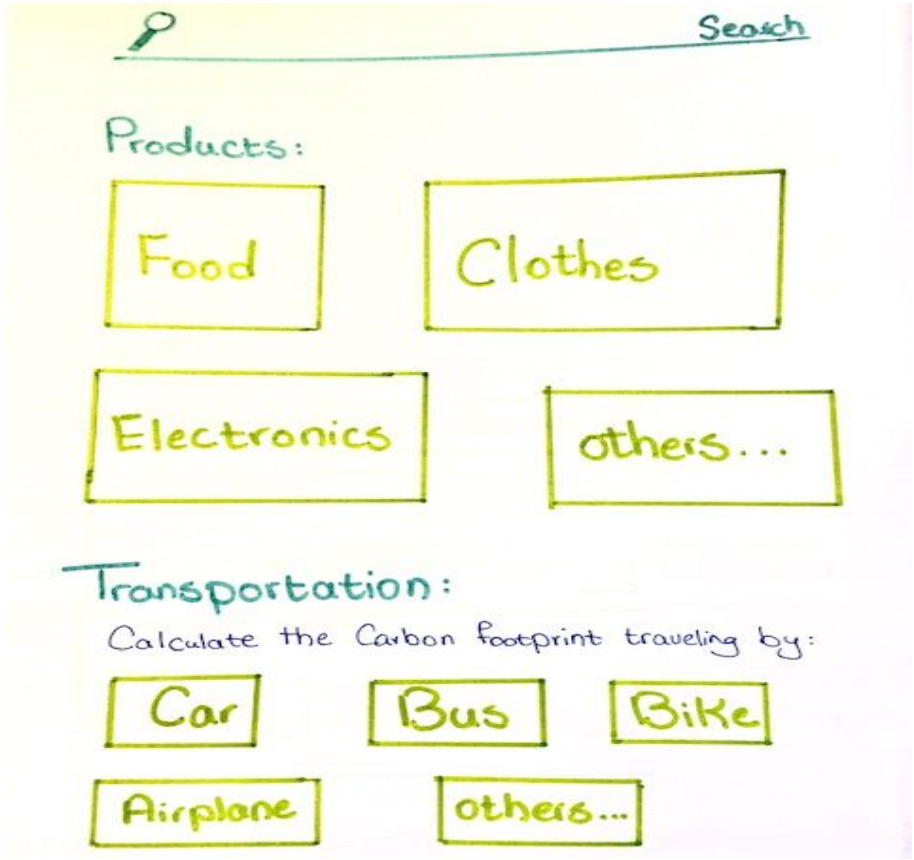
Use Case #2 - Auto-Import Product Data



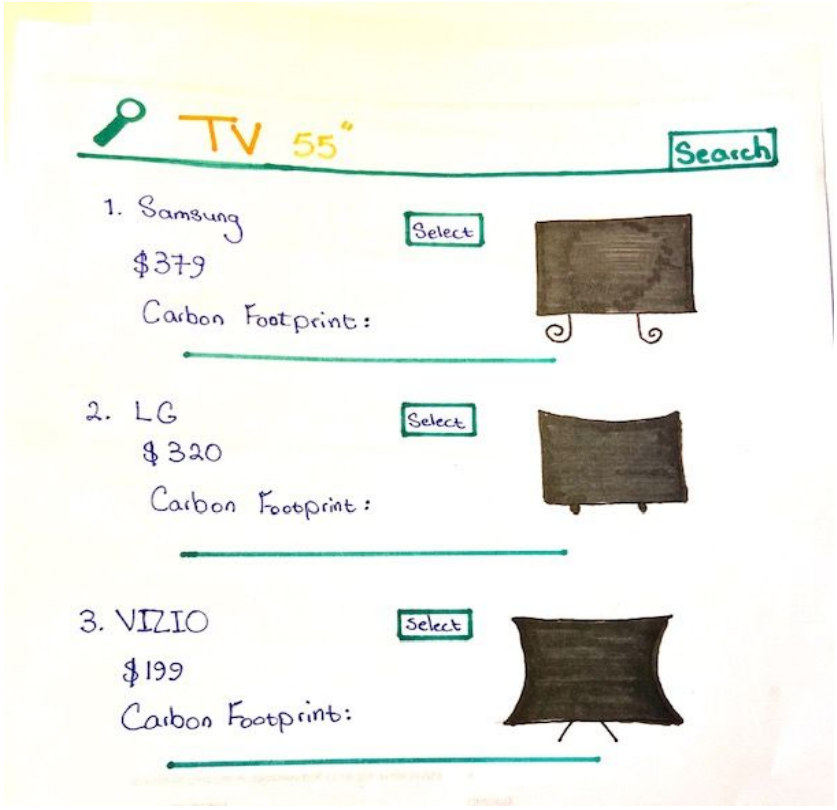
Use Case #2 - Manually Enter Product Data



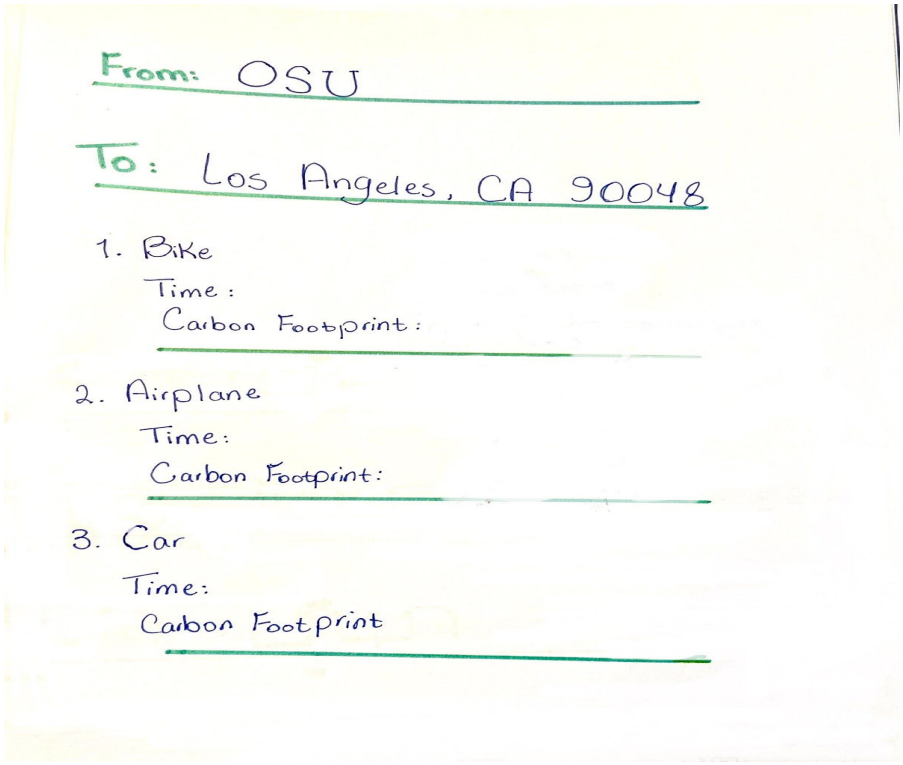
Use Case #3 - Search



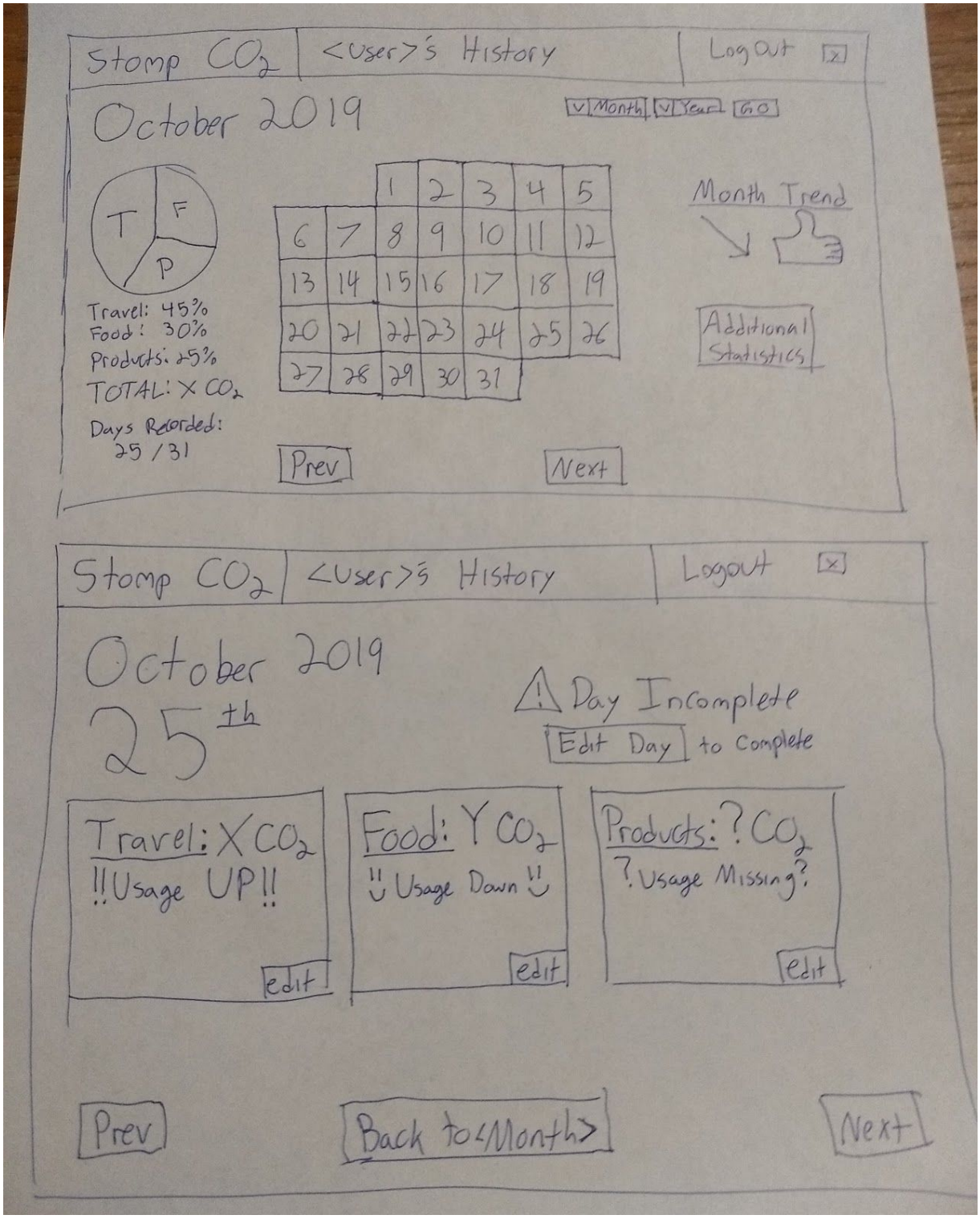
Use Case #3 - Product Comparison



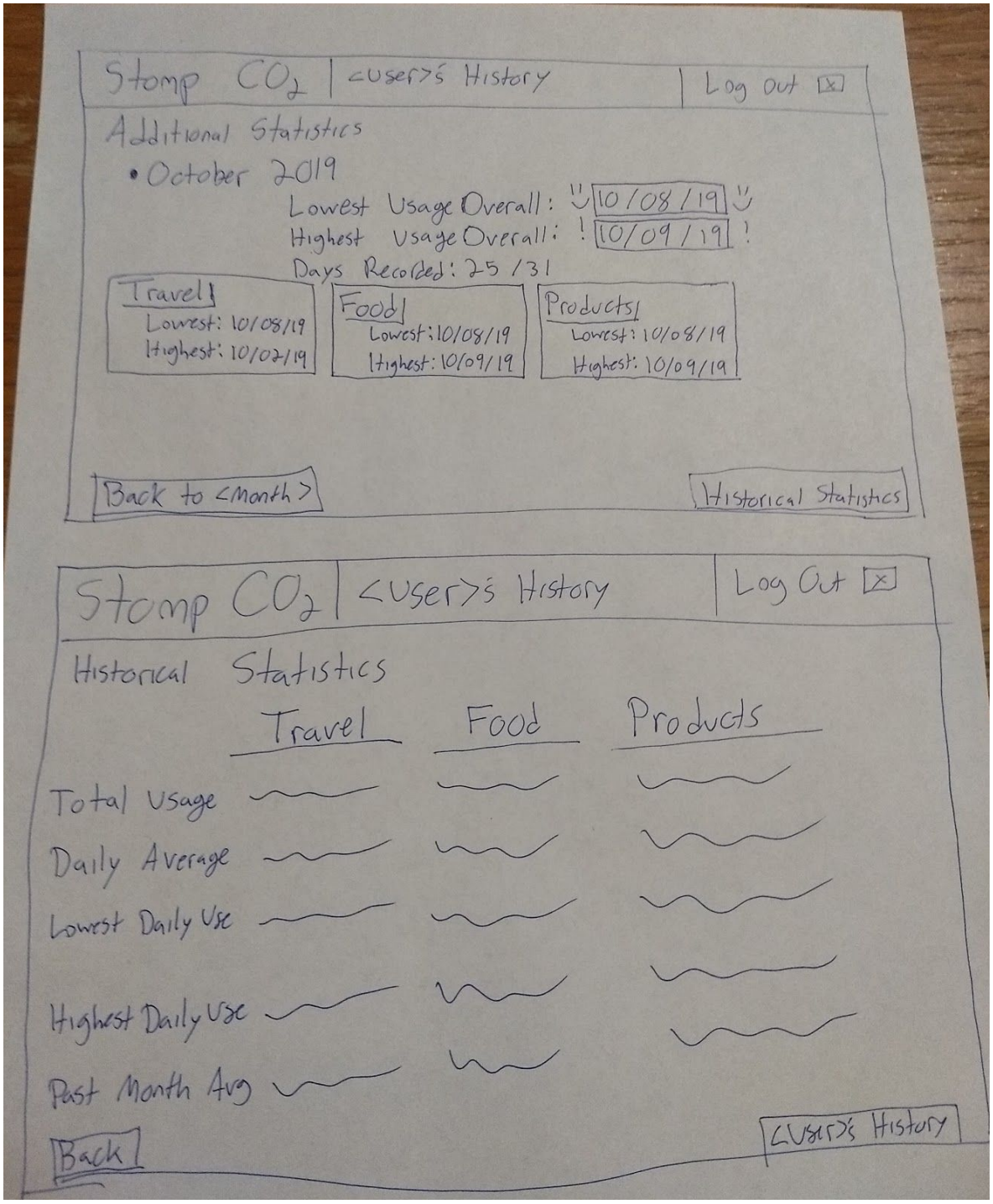
Use Case #3 - Transportation Comparison



Recent User History



Historical User Data



Requirements Definition

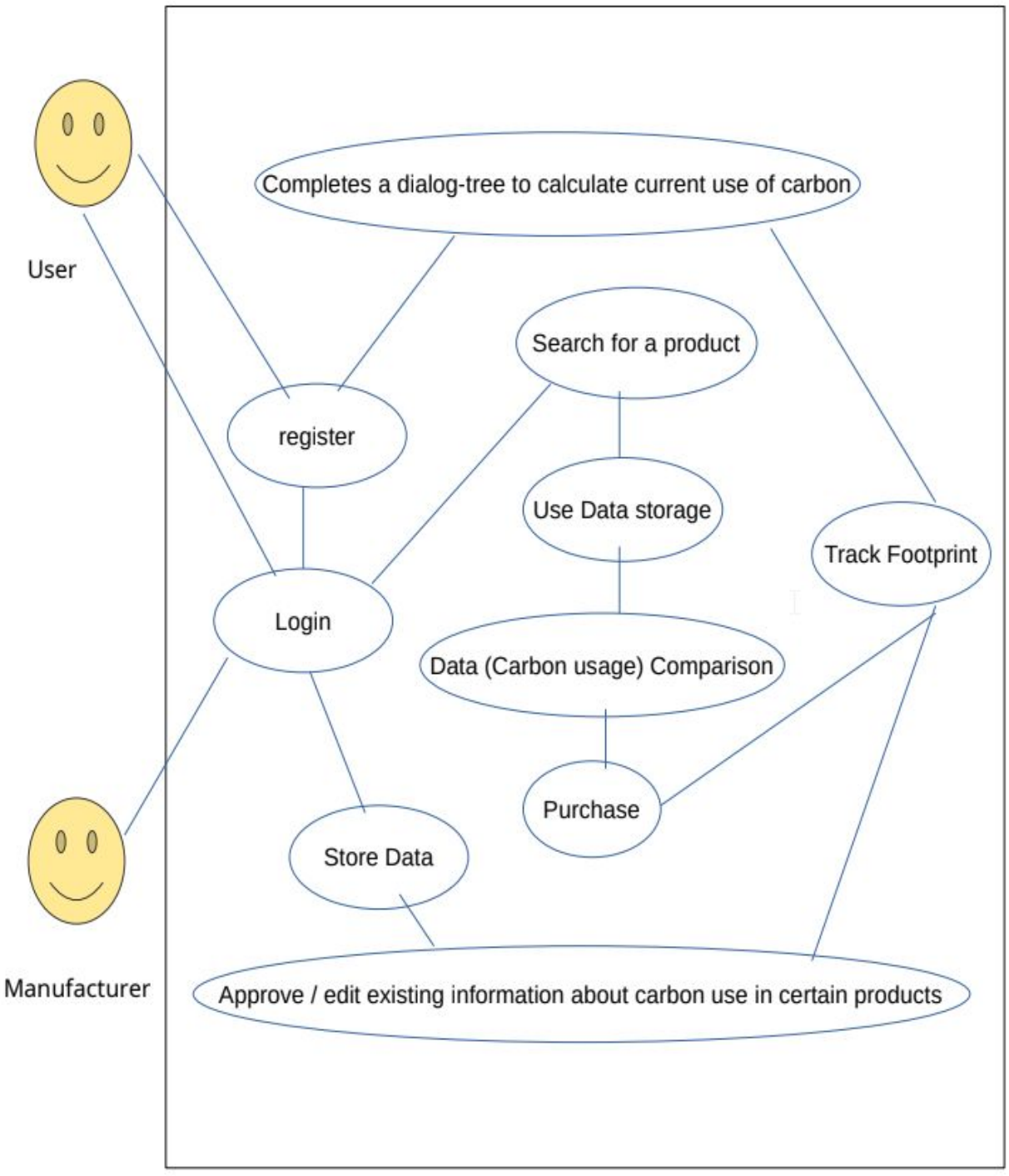
Functional Requirement

- The system shall allow only registered user to access data.
- The system shall allow new users/manufacturers to register.
- The system requires login in order to have access to personal profile.
- The system shall prompt the user to complete a dialog-tree to calculate current use of each user, through either manual input or confirming automatically received data.
- The system shall allow user to update/save new data.
- The system stores the data entered by user.
- The system shall allow manufacturer to contest the validity of existing data.
- The system shall allow the user to have access to stored data from manufactures.
- The system shall allow the user to search for a particular product and display the information about the product provided by different manufacturers.
- The system shall display a report of user carbon footprint.
- The system shall warn the user if their carbon footprint is increasing.
- The system shall allow the user to compare similar products from different manufacturers in a single-view window.
- The system shall display a graph of monthly and yearly report in user profile.

Non-functional Requirement

- The home page will be displayed within 3 seconds of the user navigating to it.
- The system shall process user login within 5 seconds.
- The system shall load user profile within 10 seconds.
- The system shall load the searched product within 10 seconds.
- The producer product auto-import feature will parse and upload product data to the server within 30 seconds per product in the file of the producer uploading the document.
- Stored personal identifying information is kept secure if there has been account activity within the past one year. Otherwise, it is deleted securely.

Carbon Footprint



Use Case #1: Measuring the end user's use of carbon

Actor: Individual and consumer of commercial products

Preconditions:

- The end user has the willingness to allow the application to query any relevant user data from internet of things devices or services to allow for auto-filling of the days carbon usage.
- The end user has the willingness to catalog data about the carbon using activities that they engage in each day when auto-filled information is unavailable.
- The end user possesses and understands the use of either or both a desktop or mobile computing device.

Postconditions:

- The data recorded by the end user is available in the database for that specific user.
- The end user's data is anonymized and added to overall numbers, for the whole of the user base and the general public to search.
- The end user is willing to adjust their behavior to a new pattern which emits less carbon by choosing strategies which lead to a smaller total use.

Flow of Events:

- The end user creates or logs into their account which stores basic information (their name and contact information) by either visiting the web application's URL or downloading the mobile version of the application from the device maker's online application repository.
- On initial setup the user is asked about dwelling size and family composition for later calculations.
- The user will also enter credentials for APIs integrations with any services which can automatically populate user data for the application.
    - Car API for mileage
    - Smart refrigerator API for consumption
    - Airline API for mileage
    - Smart thermostat integration for consumption
- The end user receives a push notification which alerts them that all values have been filled through integration with IOT APIs or that some values remain unfilled and need manual intervention.
- The end user is presented with a web form that asks various questions about the use of specific goods through a dialog-tree based user interface for items which

are not automatically filled in. These questions ask for information about such things as:
    - Current days travel
        - Walking or bicycling
        - Personal Car
        - Car as a service
        - Airplane
        - Ferry
    - The current day's use of heating or cooling of their dwelling
        - Number of hours on for heating
        - Number of hours on for cooling
    - Products consumed today, with the ability to scan receipts and use natural language processing to more quickly process receipts when available
        - Food consumed
        - Number of hours with electric lighting in dwelling
        - Total time in shower or use of bathtub
    - Any use of a fireplace or open fire
    - The number of hours of electronic devices used
        - Television
        - Computer
        - Mobile device
- The end user completes the current days dialog-tree and presses the submit button.
- The system calculates the added linked usage related to their entered data such as:
    - Refrigerator use
    - Lighting use for preconfigured dwelling size
    - The carbon use which is linked from the producer database for each product
        - Transportation of good consumed
        - Electricity used to produce goods
        - ETC. (further described in the producer use case)
- Once the end user submits the form, they are presented with a summary of what they have submitted which will be stored for the current day. The user may edit individual fields, if they wish.
- The user the user is then returned to their home screen, which contains quick data visualizations for their overall use, and how it compares to the average user.

Use Case #2: Measuring producers' use of carbon

Actor: Producer of a commercial product

Preconditions:

- The producer has access to data about the amount of carbon used in manufacturing one or more of their products.
- The producer has distinguished between material uses of carbon (such as the electricity used in their manufacturing process) and immaterial uses of carbon (such as the carbon used in the creation of old software that was created many years ago).

Postcondition:

- The data recorded by the producer is available in the database for all interested parties (other producers and the general public) to search.

Flow of Events:

- The producer creates or logs into their account which stores basic information (their name and contact information).
- The producer selects the option to either upload a preformatted document for auto-importing data or to enter data manually via a form.
- Whether the producer chooses to use the auto-import option or the web form option, the auto-import feature or producer enters information about about such things as:
    - The name of the company
    - The name of the product
    - The role of the person entering the information at the company (useful for later validating data if necessary)
    - The carbon used in the production and procurement of raw materials for the product
    - The carbon used in the generation and usage of electricity for the manufacturing process
    - The carbon emitted from any pollution from the factory
    - The carbon used when transporting the good for sale
    - Any carbon generated by the product when operated
    - Any carbon generated when the product is disposed of
    - Any other relevant notes, such as how the product can be responsibly recycled
- If the producer has chosen the auto-import option:

    - They are presented with the data that was collected and have the opportunity to confirm or edit the data.
    - The system also notifies the producer if there were any problems auto-importing data and prompts them to manually enter any data that could not be auto-imported.
- If the producer chooses the option to manually enter the data via a form:
    - The producer enters all the information they have available into the form and presses the submit button.
    - If any required fields are not filled in, the system alerts the producer, asks them to fill in the missing requirements, and asks them to submit the form again.
- Once the producer confirms the correctness of the auto-imported data or submits the form, they are presented with a summary of what they have submitted which they can print for their records if they wish.
- The producer is also asked if they would like to submit information about another product they produced.
    - If they choose “Yes,” they are presented with the options of auto-importing data or filling out a web form again, and this process is repeated.
    - If they choose “No,” they are returned to their account homepage.

Use Case 3

- Users can compare the carbon used by different products.

Actors: Manufactures, Users

Preconditions:

- Administrators add data about carbon usage for a list of products.
- Manufacturers approve/correct the collected data about their carbon usage.
- Manufacturers add unlisted products and the amount of carbon used.
- Users have access to the list of products and their accurate carbon data.
- Users can add a list of their purchases and find out their amount of carbon usage.

Postconditions:

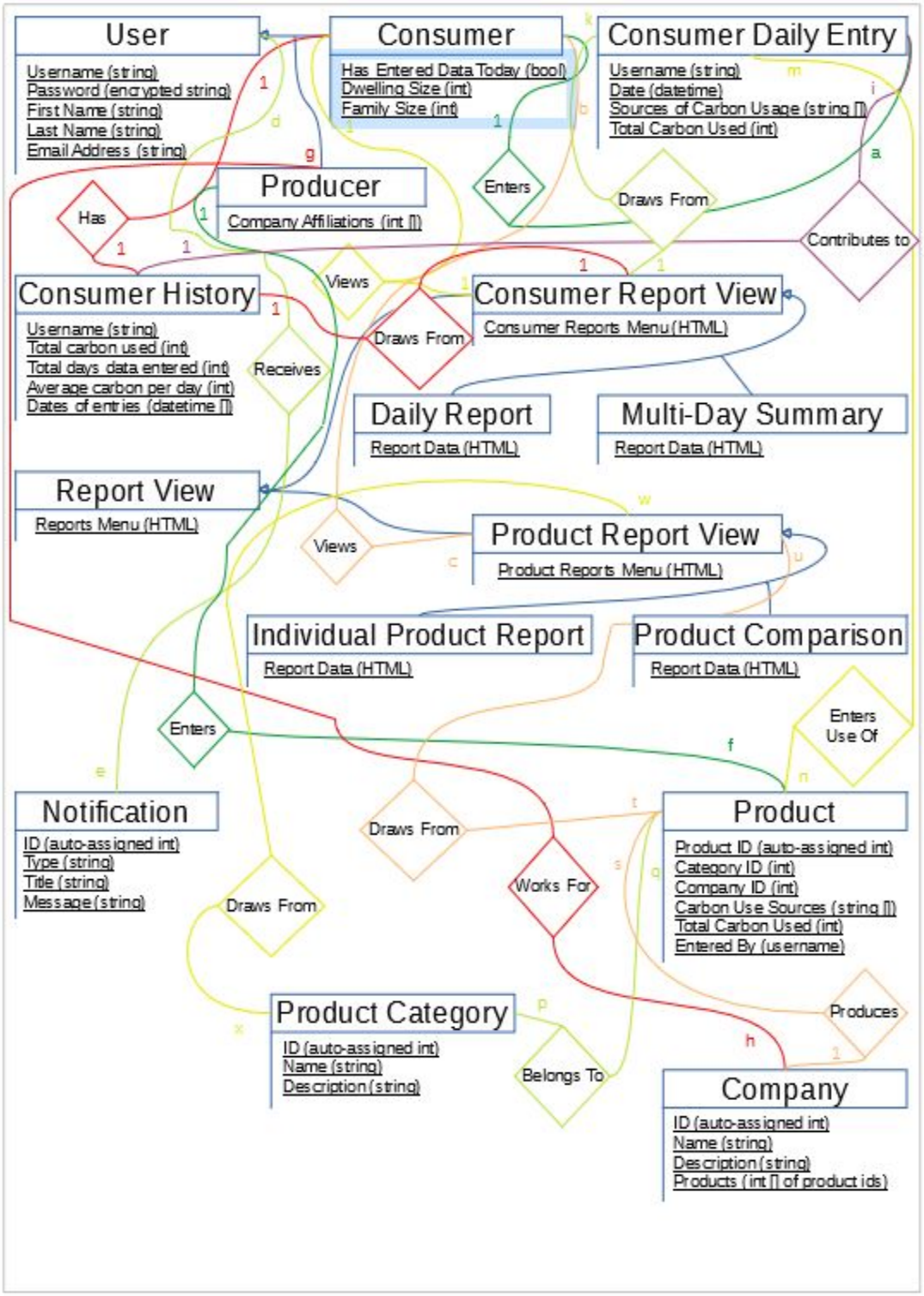
- User could compare different products.
- User could make a better decision purchasing a product with less carbon used.
- Carbon usage will be a factor in purchasing a product.
- Manufactures decrease carbon usage in order to compete together.

Flow of events:

- User can make a profile on the website/app and add purchases.
- Each manufacturer has a profile and a list of products including detailed information about each product.
- Administrators add collected data about products which has no participating company.
- Each company can edit or approve the information collected by administrators and start participating by making a profile.
- System uses information provided by companies and data collected by administrator to rank the products based on their carbon usage.
- User has the ability to search for a certain product and get a list of that product made by different companies and get access to the ranking based on carbon usage.
- User can make better decisions every day by being able to compare different products and procedures of each company making that product.
- The purchase will be added to the user's profile and user can track how much carbon has been saved by picking the better product.
- User will also get notified about how much carbon has been wasted (could be saved) if other choices would be made.

- User has access to daily, monthly, and yearly report of their profile activity. The amount of carbon they have saved and the amount of carbon they could save by purchasing from other companies.

ERD


User
Username (string)
Password (encrypted string)
First Name (string)
Last Name (string)
Email Address (string)
Consumer
Has Entered Data Today (bool)
Dwelling Size (int)
Family Size (int)
Consumer Daily Entry
Username (string)
Date (datetime)
Sources of Carbon Usage (string [])
Total Carbon Used (int)
Producer
Company Affiliations (int [])
Has
Enters
Draws From
Contributes to
Consumer History
Username (string)
Total carbon used (int)
Total days data entered (int)
Average carbon per day (int)
Dates of entries (datetime [])
Views
Receives
Consumer Report View
Consumer Reports Menu (HTML)
Daily Report
Report Data (HTML)
Multi-Day Summary
Report Data (HTML)
Report View
Reports Menu (HTML)
Product Report View
Product Reports Menu (HTML)
Individual Product Report
Report Data (HTML)
Product Comparison
Report Data (HTML)
Enters
Enters Use Of
Notification
ID (auto-assigned int)
Type (string)
Title (string)
Message (string)
Draws From
Works For
Product
Product ID (auto-assigned int)
Category ID (int)
Company ID (int)
Carbon Use Sources (string [])
Total Carbon Used (int)
Entered By (username)
Product Category
ID (auto-assigned int)
Name (string)
Description (string)
Belongs To
Produces
Company
ID (auto-assigned int)
Name (string)
Description (string)
Products (int [] of product ids)

Requirements Specifications

Functional Requirement

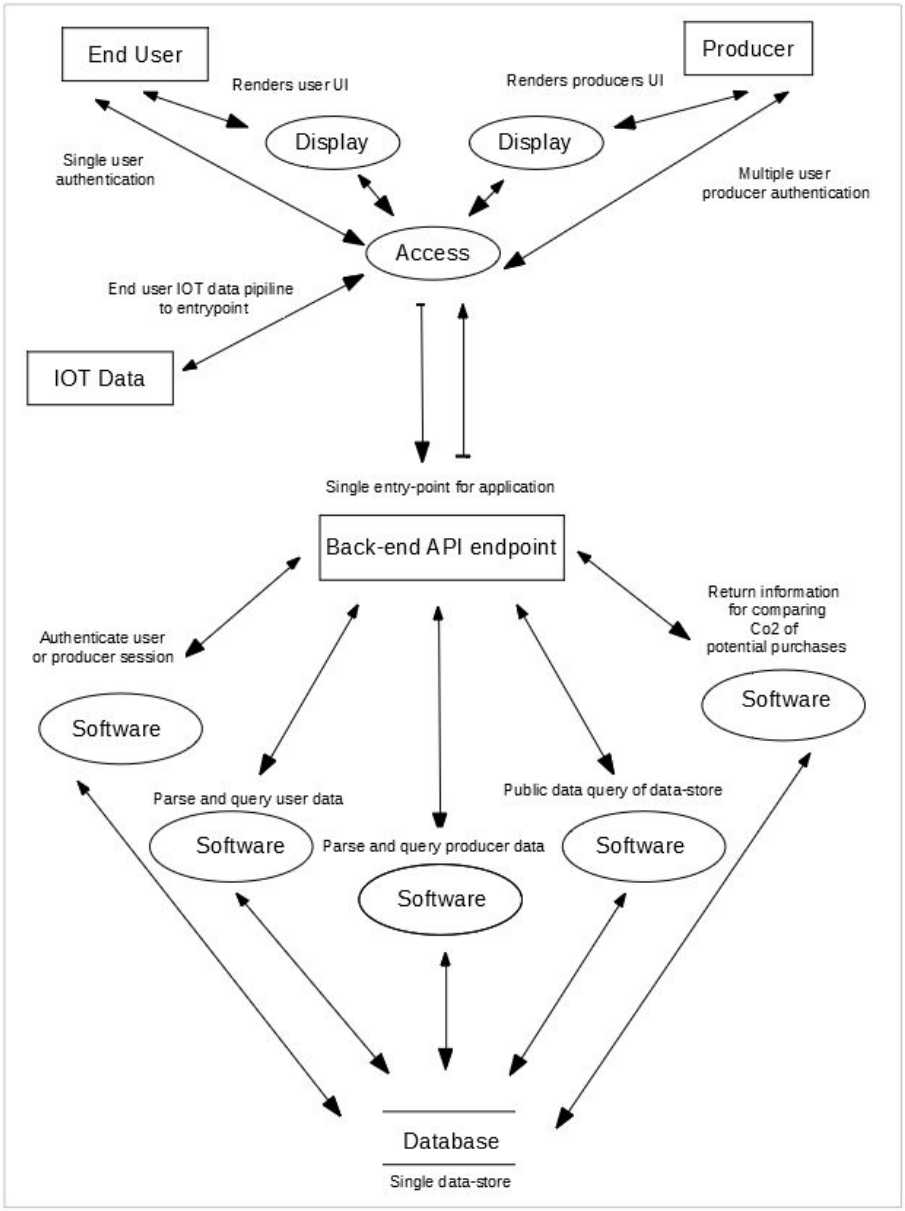
- The user interface for a consumer or producer login will pass user credentials over an encrypted connection to the user account database to be verified against those in database and either authorize or reject the login request.
- The application allows creation of new, unique user accounts as either a consumer or producer and stores the credentials in the database.
- After login, user data is passed from the database to the user interface.
- After account creation, software will serve the user a dialog-tree and send user responses to the database.
- The user interface allows the user to confirm automatically retrieved data, manually submit new data or update data. The user interface then sends that data to the database, which maintains data integrity.
- If the user is a producer, the system allows the user to contest the validity of data in the database.
- The user interface displays data from the database, regardless of the data's origin.
- The software allows the user to input the name of a particular product, which the software queries the database for. The database returns product information to the software, which displays that information to the user.
- The software displays a summary of user carbon footprint data stored in the database based on data previously entered by the user.
- When the software displays a summary, it will indicate to the user if their carbon footprint has increased based on recent data from the database.
- The software allows the user to input the name of a manufacturer, which the software queries the database for. The database returns product information to the software, which displays that information to the user in a comparison view of multiple products from that manufacturer.

Non-functional Requirement

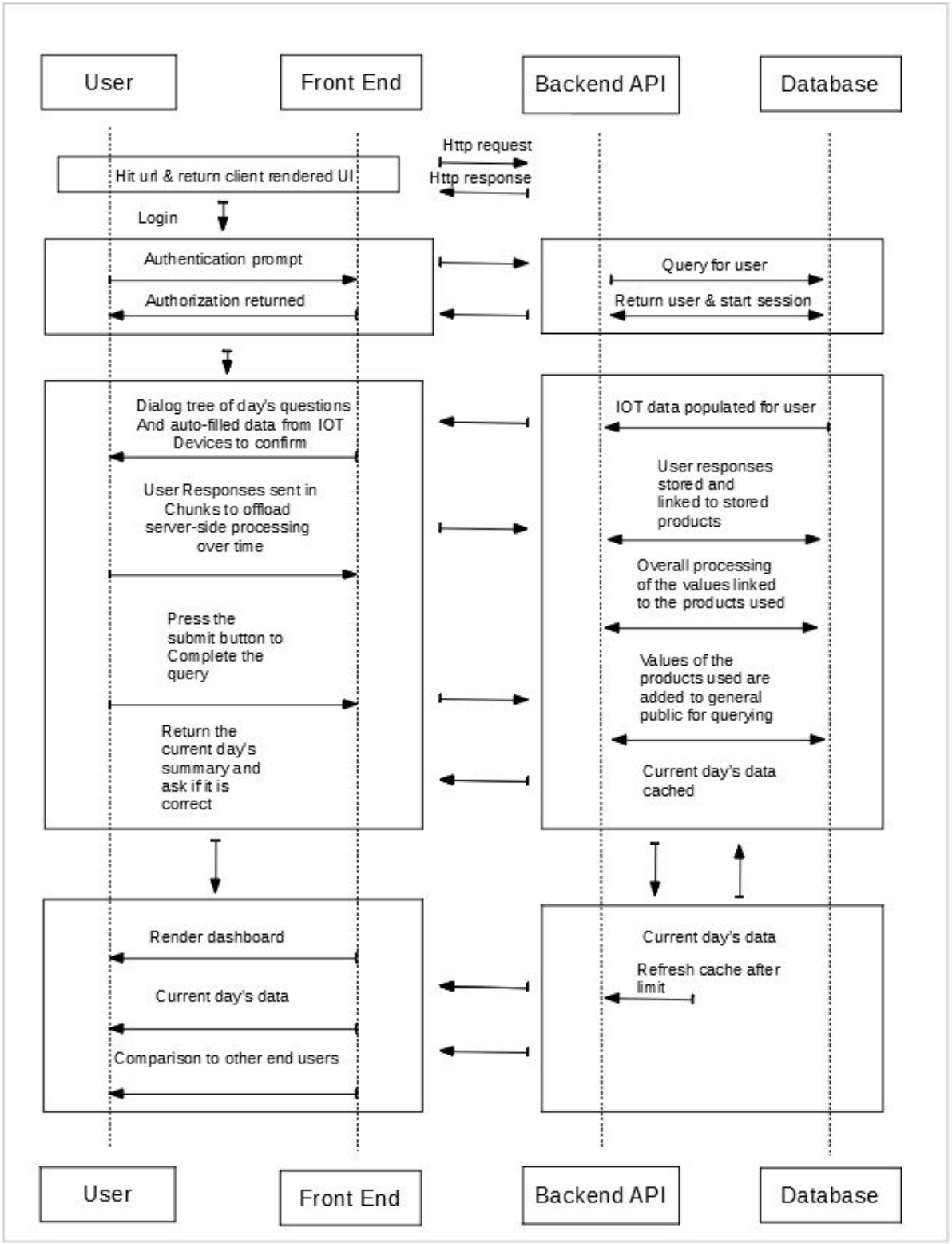
- The system will begin to render the home page UI within 3 seconds of a client request.
- The system queries the database with user login credentials and responds within 5 seconds.
- The system queries the database for user profile data and returns that data within 10 seconds.

- The system queries the database for searched product data and responds within 10 seconds.
- When the system receives a product data file, it will parse and upload product data to the server within 30 seconds per product in the file.
- The system stores personal identifying information securely using the latest best-practice encryption standards for at most one year since the user has last been active.

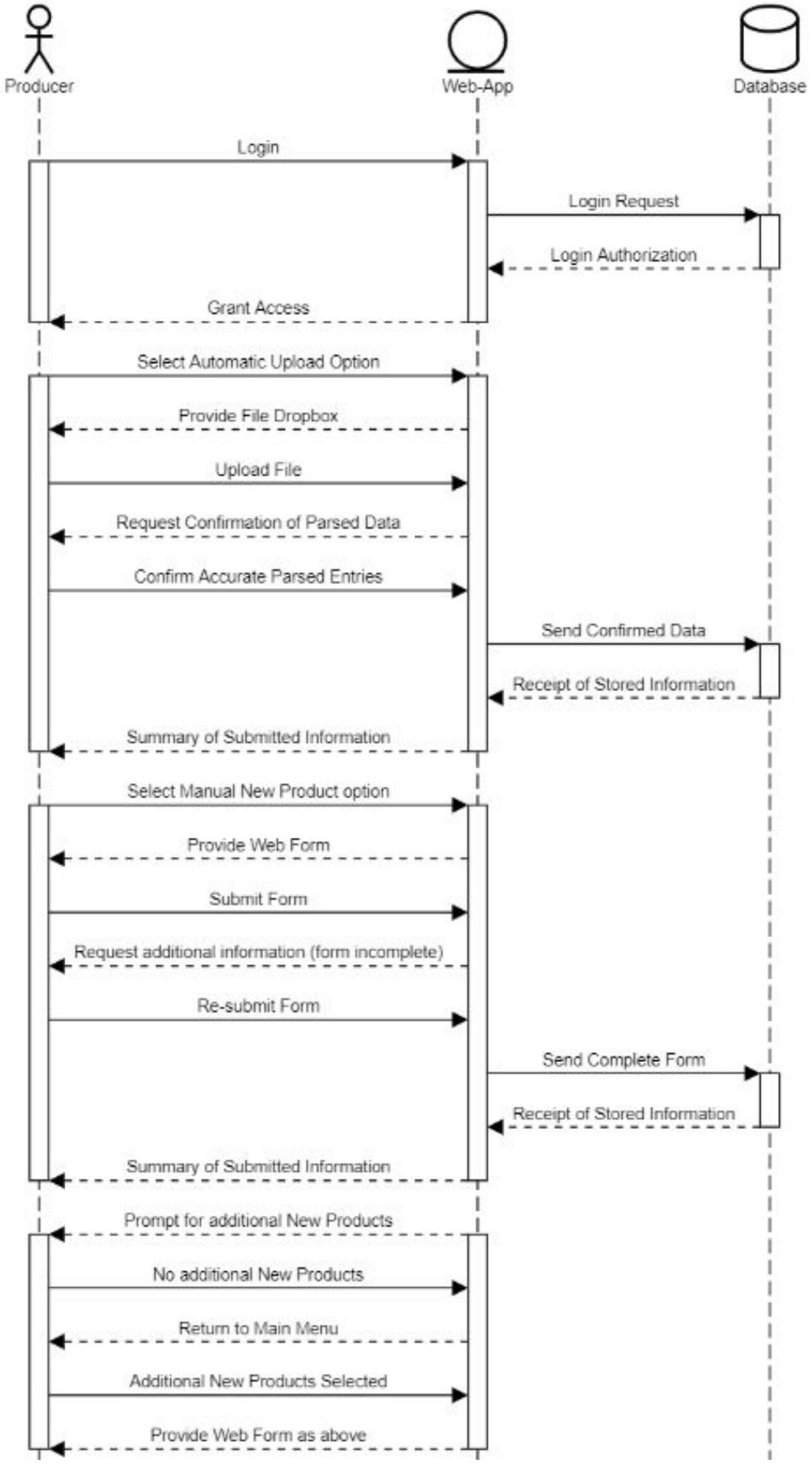
Dataflow Diagram

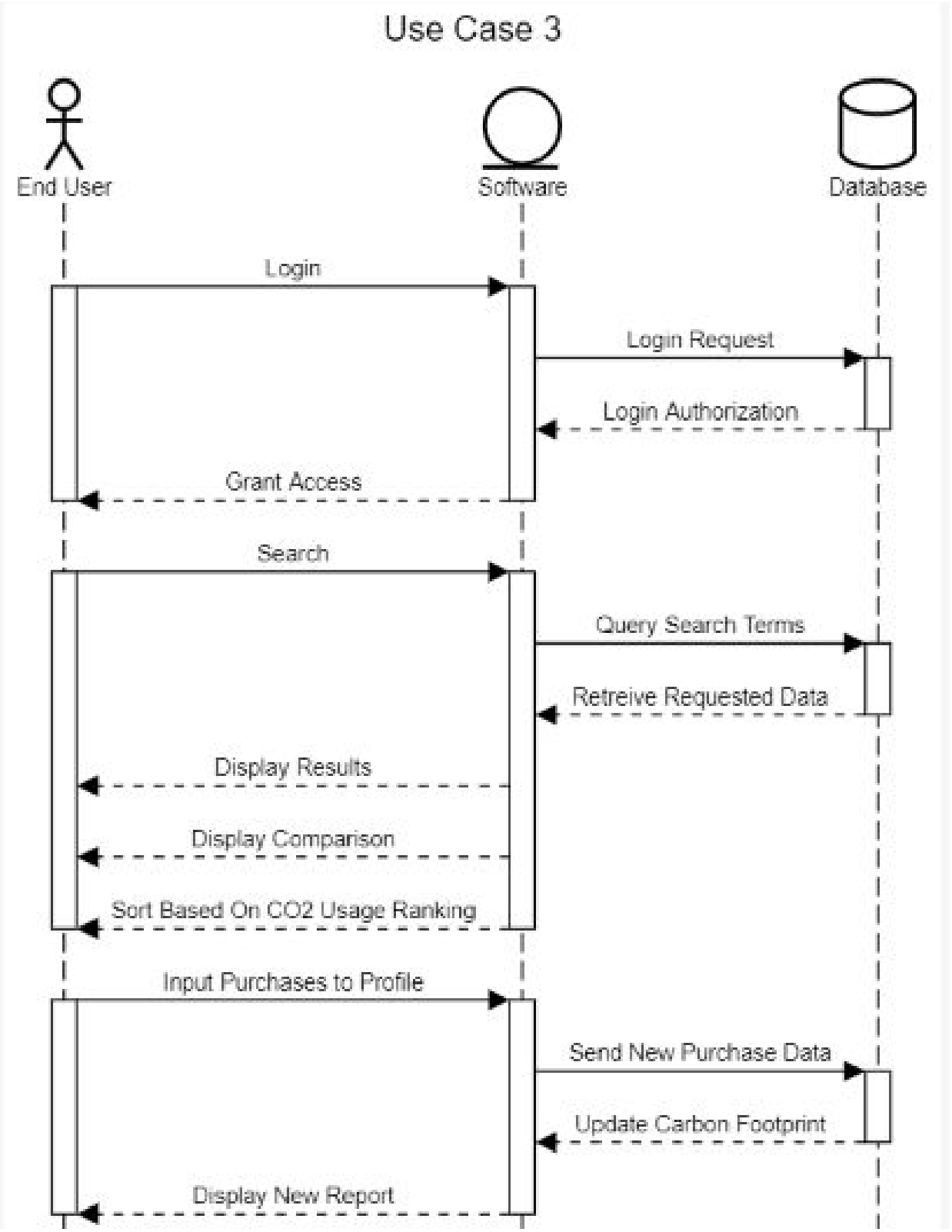


Sequence Diagrams
Use Case 1



Use Case 2


Producer
Web-App
Database
Login
Login Request
Login Authorization
Grant Access
Select Automatic Upload Option
Provide File Dropbox
Upload File
Request Confirmation of Parsed Data
Confirm Accurate Parsed Entries
Send Confirmed Data
Receipt of Stored Information
Summary of Submitted Information
Select Manual New Product option
Provide Web Form
Submit Form
Request additional information (form incomplete)
Re-submit Form
Send Complete Form
Receipt of Stored Information
Summary of Submitted Information
Prompt for additional New Products
No additional New Products
Return to Main Menu
Additional New Products Selected
Provide Web Form as above


Use Case 3
End User
Software
Database
Login
Login Request
Login Authorization
Grant Access
Search
Query Search Terms
Retreive Requested Data
Display Results
Display Comparison
Sort Based On CO2 Usage Ranking
Input Purchases to Profile
Send New Purchase Data
Update Carbon Footprint
Display New Report

Summary of Changes Based on Feedback from Homework 1:

When working on updating our Homework 1 documentation this week, we considered feedback from both our customer and our grading TA. Our customer, Shawn, joined a video meeting with our group and gave a significant amount of helpful feedback. He also followed up with more detailed feedback by means of comments within a Google Doc copy of our Homework 1 submission. We then made a copy of the annotated Homework 1 to begin revising it for Homework 2. That way, we were all able to make changes using Shawn's comments as a guide and then resolve comments in the Homework 2 document while still preserving the original feedback in the annotated Homework 1 document. The overarching theme of his feedback was that more automation should be included in data collection. He imagined this product to be released in the near future with increased capabilities in Internet of Things devices. This, he said, would allow data to be pushed to our system without the user having to manually and arduously enter each point of data personally. He also gave some feedback about other miscellaneous details, such as data security.

On the title page of the paper, before the subtitle giving a description of all that is included in the document, we decided to add an indication of the particular assignment number. This label of "Homework 2" will make it clearer in the future when referring back to this document exactly which particular assignment this document pertains to, which will greatly help us with staying organized. Organization is, after all, a very important part of software engineering.

The Requirements Definitions and Specifications were updated. For functional requirements, our customer requested that we make it clear that data would be automatically imported whenever possible, so we updated the language to reflect this. For non-functional requirements, our customer requested that we add information about how users' personally identifying information would be kept secure. To expand our non-functional requirements, we also added information about how quickly the home page would be rendered when requested and how quickly auto-imported product data would be parsed and pushed to the database.

The English description for Use Case #1 was updated to allow for a few additional specific data pipelines requested by the customer.The hope is that as more Internet of Things devices come onto the market that data pipelines will increasingly supplement and eventually replace the need for manual entry of any data by the user, over time. With such extensive data, the customer noted the need for security regarding personal information. We included a step to anonymize personal data while including such data to a searchable portion of the database for general search, use, and research from interested parties.

The English description for Use Case #2 was updated to reflect the customer's request that data collection be automated whenever possible. Now, instead of a producer being required to manually enter all data about the carbon emitted during the production of a product, the producer will have the option to either upload a preformatted document from which data can be automatically pulled or manually enter the data via the form. Clarification was added for how the process would differ between these two options. For example, if the producer auto-imports data, they are then presented with the imported data and given the opportunity to edit or confirm the auto-imported data. In addition, the system will perform error checking and prompt the producer to manually enter any data that could not be auto-imported successfully. Clarification was also added to indicate that, if the producer chooses to enter additional data from the summary screen describing the data they just entered, they will again be asked whether they would like to auto-import or manually enter the new data.

The dataflow diagram was updated to reflect that Internet of Things devices are involved whenever possible. A rectangular "actor" box was added to represent Internet of Things devices, since they would actively contribute to the data input process as opposed to simply serving as a function (which would have meant they would have been in an oval) or serving as a database (which would have meant they would have been placed within an open rectangle).

The message sequence diagram for Use Case #3 was updated, correcting the "Producer" label to be "End User" instead. A few minor changes were also made, despite the customer not requesting any significant alterations. The Use Case #2 message sequence diagram was updated to include a portion related to automatic data entry, as the customer requested. This was differentiated with the previous version by adding "Select Manual New Product Option" to the web form section. Similarly, the Use Case #1 message sequence diagram has been updated with Internet of Things data being automatically generated for End User confirmation during the dialog-tree process.

Our customer reviewed, and was happy with our paper prototypes and did not request any changes. We drew our prototypes after having previously met with the customer earlier in the week and having discussed his requested changes to the requirements definitions and specifications. Therefore, we incorporated his feedback into the prototypes. For example, the prototypes reflected the requested ability of data to be auto-imported when possible and to optimise for the minimum amount of manual user input.

We all enjoyed the opportunity to reflect upon and update our requirements documentation from Week 1. The simulation of receiving customer feedback, responding to it professionally, and making the required changes to evolve our project organically based on the customer's desires will be invaluable as we prepare for careers after graduation.

Customer Meeting Summary

We met with our customer, Shawn, via Google Hangouts on Wednesday evening. Shawn looked over our HW1 documents and gave initial overall feedback about the importance of having data collection be automated whenever possible. He then annotated our HW1 Google Doc with comments on Thursday with more detailed information he would like to see changed. We agreed to send him drawings of our paper prototypes by Friday evening for him to review.

Team Member Contributions

The following list indicates team member contributions during the preparation of HW2:

- All - Direct team meetings and continued communication via Google Hangouts and email; contributed to 2-page summary of changes to HW1 documents based on customer's feedback; revised English sections of Requirements Definitions and Requirements Specifications based on customer's and grader's feedback.
- Alex Densmore - Revised Use Case #2 text based on customer's feedback; drew paper prototypes for producer homepage and data entry screens.
- Anousha Farshid - Revised Use Case #3 text; drew paper prototypes for product comparison, travel comparison, and product search.
- Adam Wright - Revised Use Case #1 text, Dataflow Diagram, and Use Case #1 Message Sequence Chart; drew paper prototypes for login screen, home dashboard, manual dialog tree, and user preferences.
- Ken Wyckoff - Revised Use Case #2 Message Sequence Chart; drew paper prototypes for recent user data and historical user data.

Each team member was also involved in creating this document from the various parts completed individually, along with reviewing said documents for accuracy and assurance.