PROJECT REQUIREMENTS

Table of Contents

t

1. Introduction

1.1 Purpose of Document

This document is a Requirements Specification for an interactive user interface to share the data that is owned by Boulder Apple Tree Project (BATP). BATP is working on sharing a rich history with a vast knowledge of apples to people all around Colorado. The new system will add a map button on the current websites to allow users searching for apple information, apple trees, and their location. This document describes the scope, objectives, and goal of the new system. The requirements specification document also models the functional requirements with use cases, interaction diagrams, and class models. This document is intended to direct the design and implementation of the target system in an object-oriented language using multi-stack technologies.

1.2 Project Summary

Project Name:	Boulder Apple Tree Project
Project Manager:	Amy Dunbar-Wallis
Project Analysts:	Eden Crashers
Responsible Users:	Boulder community

1.3 Background

Colorado is home to a wide variety of apples. Boulder Apple Tree Project (BATP) is currently working on sharing a rich history with a vast knowledge of apples to people all around Colorado. By partnering with the Boulder Apple Tree Project, our senior design team will work on an interactive user interface to share the data that is now owned by the Boulder Apple Tree Project. Together, we can help promote appreciation and preservation of apple trees and their fruit, their genetic lineage, and their connection to the history of Boulder. Products are needed to share knowledge in a meaningful, engaging, and enjoyable way.

The passionate members of the Boulder Apple Tree Project (BATP) have collected a wealth of information, including historical maps, ledgers, and tree data. They are currently making an effort to expand the knowledge base of apple trees within and around the Boulder County area through tagging trees, gathering missing data of the variety of trees and their fruits, and gaining insights into the past through oral and written histories. With this data, the BATP created simple maps representing the locations of apple trees in the past and present.

1.4 Project Scope

The scope of this project encompasses an interactive map as the central product. The interactive map will consist of a display as well as information about apple trees in the Boulder area. The interactive map will also allow for engagement and interactiveness with actions such as hovering, filtering, clicking, and selecting different apple trees. The database of apple trees will be scalable and allow for expansion. Eventually, the app will be upgradeable to be deployed online and in museums. The app and web page will also need to be compatible with mobile displays. The locations of apple trees should not be exact in order not to compromise the tree's safety and to respect the privacy of people who owns the property where a tree is growing.

1.5 System Purpose

1.5.1 Users

Those who will primarily benefit from the new system and those who will be affected by the new system include users, product owners, customer service departments, and information technology departments.

- Users:
  - Upon implementation, the interactive map allows users to search and filter for the location of generalized or specific apple trees, their information, and their heritage information.
- Product Owners:
  - Product owners will be allowed to maintain the data about their products directly.
- Future Project Groups
  - Future project groups will be responsible for putting in new information into the database, maintaining and servicing the website, and maintaining the system

1.5.2 Location

The website and interactive map will be available to users that have access to the internet or visitors that visit the Boulder Museum. In the future, the EBIO department will also be able to update the database whenever needed.

1.5.3 Responsibilities

The primary responsibilities of the interactive map:

- An interactive map as the central product.
- Display information about apple trees on the interactive map.
- Engaging and interactive.

- The website allows for the expansion of the apple tree database.
- The app should be scalable to be deployed online and in museums.
- The web page and application should be compatible with mobile displays.
- Locations of apple trees should not be exact in order not to compromise the apple tree's safety and to respect the privacy of people whose property where a tree is growing.

Other desired out of scope features of the interactive map

- Developing a mobile application for IOS, Android, or any other mobile application platform.
- User credentials for log-in or
- Authorization specific capabilities such as a User credential for log-in.

1.5.4 Need

This system is needed to provide the educational experience of distinct varieties of apple trees to the public, especially in the Boulder museum. When visitors come to the museum, they can learn a rich history with a vast knowledge of apples on the touch screen monitors.

2. Functional Objectives

2.1 High Priority

In Scope:

1. The app shall allow for on-line users to navigate a map based on real data that displays a marker at the location of various, living apple-trees. For users this should be an educational experience.
2. The app shall allow response to the user's intentional manipulation of the map. If the user intends to move the map in a given direction, or zoom in or out, the map shall respond as expected. This should allow the user to control this app in the same way as they are accustomed to with other, more common apps.
3. Each marker displayed within the app's map shall be clickable, allowing the user to learn more about the apple. This should allow the user to learn about apple trees they have specifically chosen to learn more about.
4. The system should have a database that contains all information about every apple tree collected by the Boulder Apple Tree Project. This database must be either secure, or have abstraction built into it to obscure the location of certain, private apple trees. This database will be the only means of data storage this app requires.
5. The app should be able to make calls to the database as requested by app usage.
    a. There must be a way to access all apples within a given region for the purpose of populating the map.
    b. There must be a way to access all non-private information about a given apple tree for the purpose of display of specific data to the user.
    c. There must be a method by which this database can be updated to reflect the acquisition of information by the Boulder Apple Tree Project of new trees.
6. The app shall obscure the specific location of certain apple trees that are either considered in need of protection by Boulder Apple Tree Project, or that are located on private property.
7. The app shall be usable in both mobile and desktop environments.
8. The app shall be capable of responding and fulfilling the requests of multiple users at the same time.
9. The app shall be scalable such that if traffic severely increases, database size severely increases, or serviced map area severely increases, the app's code-base itself would require little to no modification.
10. The app shall allow for a search feature in which the user can input a phrase and have relevant apples and information appear on the map.

11. The app shall allow for a filter feature such that filters can be applied and have an effect on the map in a relevant manner.

2.2 Medium Priority

Stretch Goals:

1. The app's database shall have support for uploading or updating of authorized data such that it can be extended or modified when more data is collected about the apples.
2. The app shall allow for apples to be viewed in a 3D space, a model shall be made for some or all apples so that they can be viewed in a 3D interactive manner.
3. The app shall have an interesting application of data science applied to the apple data stored in the database.
4. The app shall allow for different maps based on historical period to appear when the timeline is scrubbed also updating the map to show only the relevant apples.

2.3 Low Priority

Out of Scope:

1. The app shall have cross-platform support, i.e. be a native application for iOS, Android, or any other mobile platform.
2. The app shall have user credentials for log-in or authorization of specific capabilities.
3. The app shall translate web pages into different languages.

3. Non-Functional Objectives

3.1 Reliability

- The system shall be completely operational at least 50% of the time.
- Down time after a failure shall not exceed 48 hours.

3.2 Usability

- A new user should be able to use the system immediately or after reading the help section of the system.

3.3 Performance

- AWS or servers, code commented, etc
- The system should be able to support 100,000 simultaneous users.

3.4 Security

- Location of trees!
- Inputting tree data must be transmitted in encrypted form.

3.5 Supportability

- Documentation for when we leave the project
- The system should be able to accommodate new features without major reengineering.
- The system should be able to port to different operating systems and the various popular web browsers (Google Chrome, Firefox, Safari, etc.)

3.6 Support

- The system shall provide an integrated help section and an additional web page that explains how to navigate the system.

3.7 Purchased Components

- Figma, AWS
- AWS hosting will be needed.
- Figma will be needed.

3.8 Interfaces

- ex. Data cleaning with python
- The system must interface with the new PostgreSQL / PostGIS database system for tree information.

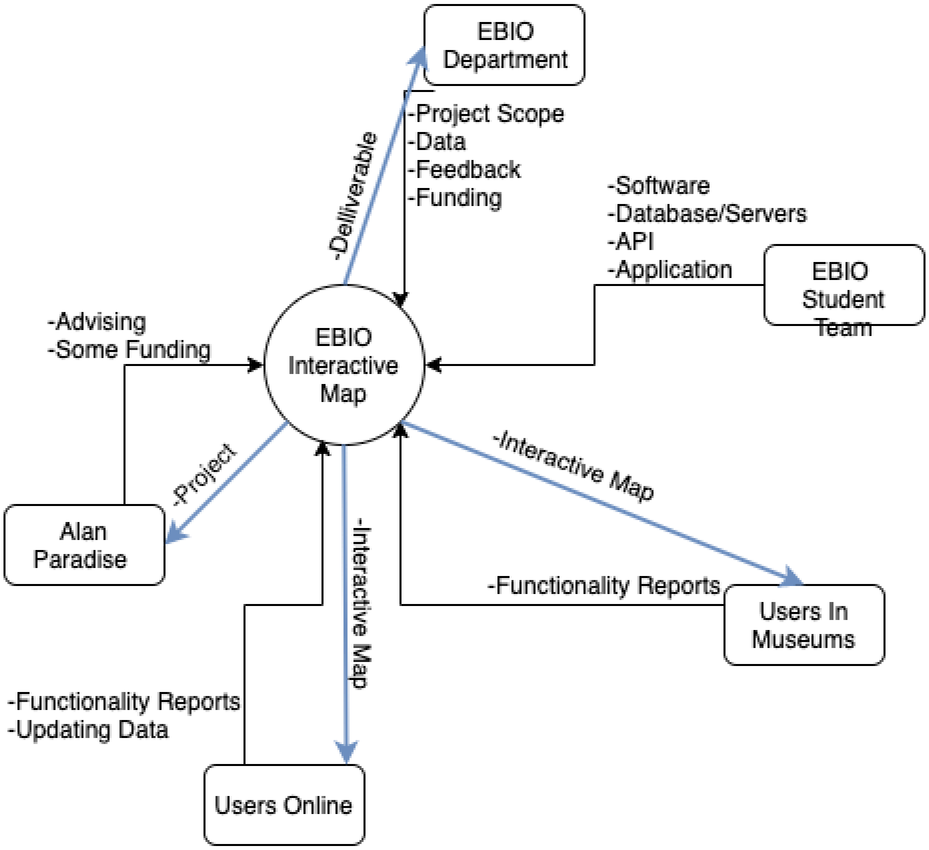
4. The Context Model

4.1 Goal Statement

The goal of our system is to allow the EBIO department to share a rich and vast history of the apple trees around Colorado. The purpose is to promote information and the appreciation of apple trees and their fruit.

- Providing users with a fast, intuitive, interactive access to the information of the apple trees.
- Allowing users to interact with the map that will display all the apples in the city in their appropriate location and all their corresponding information and images of the trees and their fruits.

4.2 Context Diagram



4.3 System Externals

Alan Paradise
Alan Paradise will be the person in charge of advising and some funding for the project. He will receive the project information/deliverables to approve of what’s being done.

EBIO Student Team
Students are responsible for creating the software to deliver an interactive map. Also responsible for creating and developing a scalable and clean database that can be easily managed by the EBIO department. Students will work with API, servers, among other softwares to fulfill this purpose.

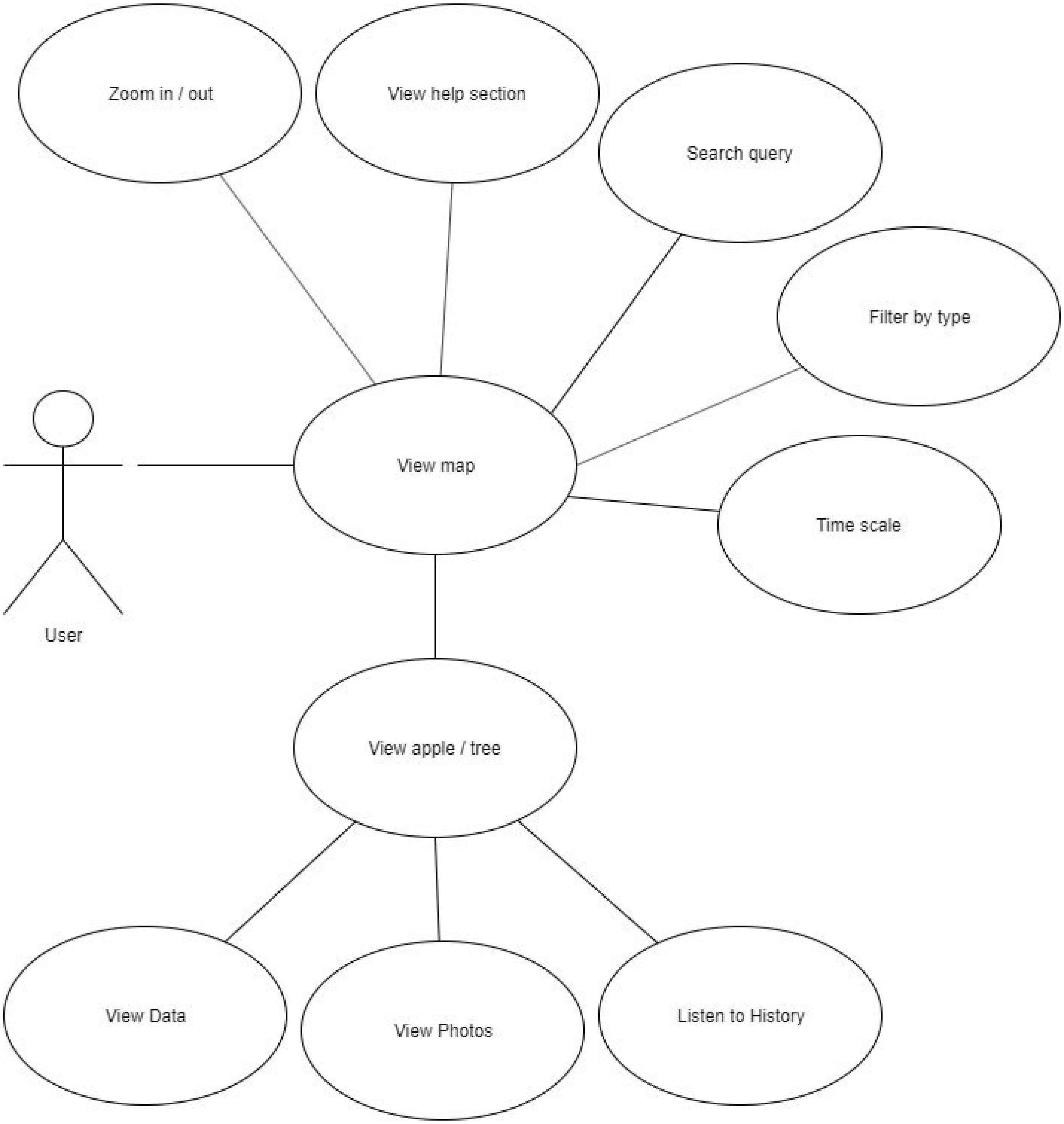
Users in Museums/Online
A user will be every user of the web application who is interested in knowing more about the history, properties and uses of the apple trees. They will use the interactive map to adquiere this information.

EBIO Department
EBIO Department is responsible for providing the scope desired for this project, in addition to funding, feedback on the implementation, and the data EBIO Student Team is going to use. In addition to historians or people who are going to add details about each tree.

---

5. The Use Case Model

5.1 System Use Case Diagram



5.2 Use Case Descriptions

Notes:

- For all use cases, the user can cancel the use case at any step that requires user input. This action ends the use case. Any data collected during that use case is lost.
- For all use cases that require a logged in user, the current login session is updated during the use case to reflect the navigation paths through the use case.

Map Display

Use Case Name:	Map Display
Summary:	In order to move around and concentrate on certain areas, users can interact with the map to zoom in, zoom out or general up, down, left, or right movement.
Basic Flow:	1. The use case starts when a user indicates that they want to change the map view. This can be accomplished on mobile by moving the map around with a finger, pinching inwards for zooming in, or pinching outwards for zooming out. This can also be done using the respective UI buttons. 2. The maps view updates depending on the action of the user. I.e. if they zoomed in the map would get larger, zoomed out in would get smaller, etc. 3. Apples and icons update respective to the new look of the map.
Alternative Flows:	● Step 2: If the user reaches the edge of the map or the full zoom in/out capacity the map provides no update, go back to step 1.
Extension Points:	None
Preconditions:	none
Postconditions:	The user now has a less or more focused view of the map cemented in the intended position wanted by the user.
Business Rules:	● Each time the map is opened it starts at the same vantage point. I.e. 50% zoom, center of Boulder. ● User is restricted to only moving around in Boulder and zooming in/out is restricted to a certain viewpoint of Boulder.

Apple Interaction

Use Case Name:	Apple Interaction
Summary:	The system allows the user to click on the apple to view more information about the specific apple tree (type, photos, etc.).
Basic Flow:	1. The use case starts when a user clicks on an apple. 2. The system opens a display window with tree information after click. 3. The user can click on the various tree data e.g. photos to see more.
Alternative Flows:	Step 3: if there is no more additional data, the use case goes back to step 2.
Extension Points:	*None*
Preconditions:	The apple or tree icon exist
Postconditions:	The user can now view tree data.
Business Rules:	Some tree data may be hidden.

Information Hover

Use Case Name:	Information Hover
Summary:	Allows the user to hover over an apple or an apple tree, and then the interactive map will create a context menu about the item.

Basic Flow:	1. The use case starts when the user is on the interactive map page. 2. On mouse hover over an apple or tree icon, the interactive map displays a context menu 3. The context menu brings up quick information about the apple such as its genetic variety 4. When the user hovers off of the apple and the information disappears
Alternative Flows:	*None*
Extension Points:	*None*
Preconditions:	There has to be at least one apple or tree icon on the map.
Postconditions:	The user hovers off of the apple or tree icon, and the context menu disappears.
Business Rules:	None

Tree Location

Use Case Name:	Tree Location
Summary:	This system allows the users to find the location of specific apple tree.
Basic Flow:	1. This use case starts when users click on maps, it will pop up the interactive map that have apple and tree on that map. 2. When users click on apple or tree, users could find specific location of apple tree
Alternative Flows:	The users can type the name of apple tree that they look for in the search bar to get its location.
Extension Points:	*None*

Preconditions:	There has to be at least search bar on the interactive map.
Postconditions:	The users can find the location of apple tree
Business Rules:	*None*

Time Scale

Use Case Name:	Time Scale
Summary:	The interactive map will have a time scale bar to interact between historical maps.
Basic Flow:	1. This use case starts when a user clicks on the time scale bar. 2. When users click on the time scale bar they can move the bar to the left and right indicating different time periods.
Alternative Flows:	*None*
Extension Points:	*None*
Preconditions:	There has to be a time scale bar.
Postconditions:	The users gets to the time period chosen.
Business Rules:	*None*

Filtering

Use Case Name:	Filtering
Summary:	The system allows the user to filter the database to display the tree associated with a particular property desired, for example user wants to display only trees that are more than 28ft tall.

Basic Flow:	1. The use case starts when a user clicks on the filtering box. 2. The system drop down all the fields the database is filtered by. 3. The user selects one field. 4. The user inputs the conditions of the field selected, e.g. heigh selected, user inputs 28ft.
Alternative Flows:	*None*
Extension Points:	*None*
Preconditions:	Successfully load all the database
Postconditions:	System was able to retrieve all the tree data matching the criteria the user specified.
Business Rules:	*None*

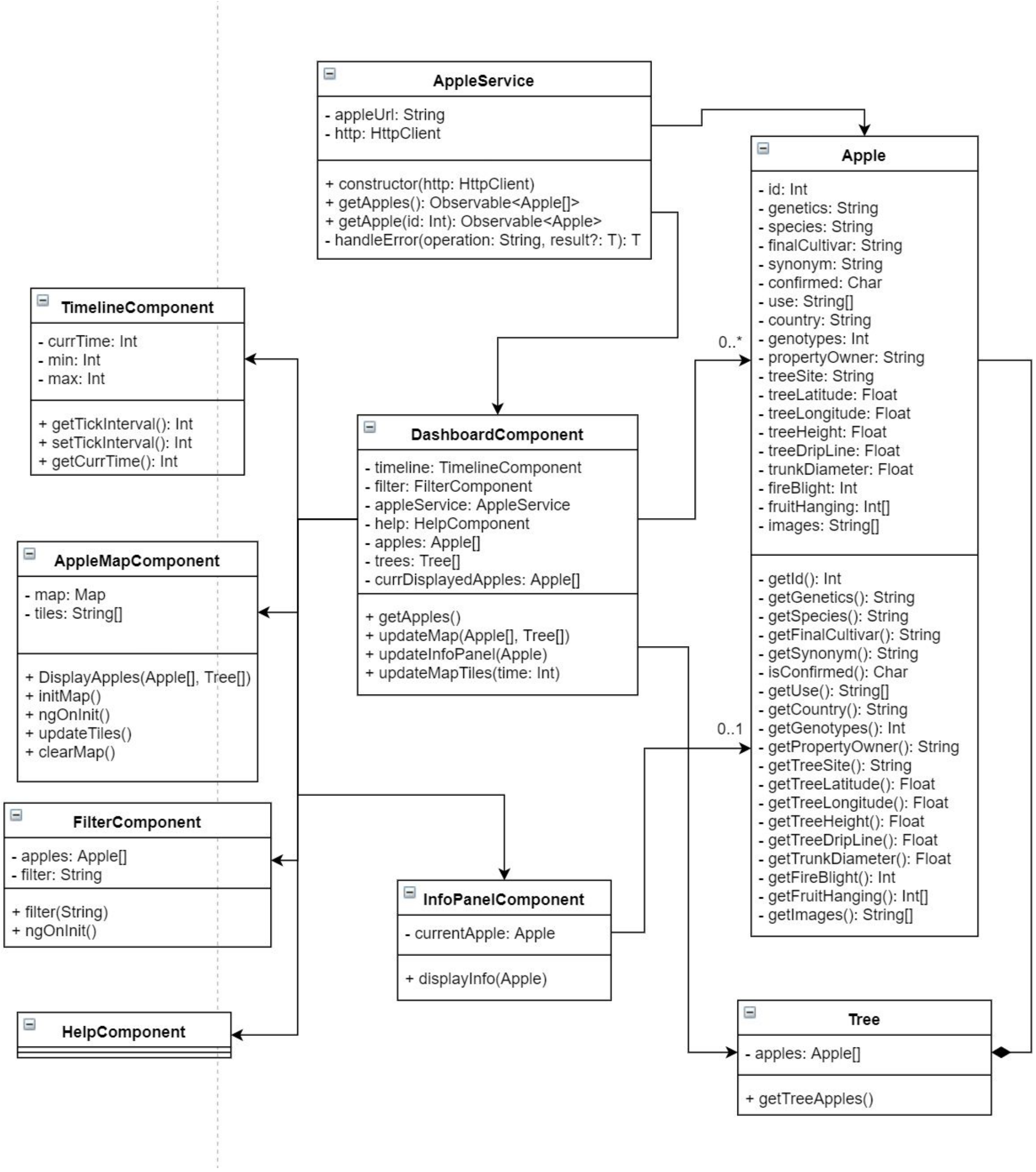
Help Section and Video Tutorial

Use Case Name:	Help section and video tutorial
Summary:	Provides details on how to navigate the application.
Basic Flow:	1. When using the web application, there will be a header link ‘Map’ that will redirect the user to the interactive apple map. 2. Users can zoom in and out of the map by using their mouse wheel, pinching or expanding on mobile, or using the (+) and (-) icons on the bottom right corner of the screen. 3. User can zoom out which will create an icon of a tree in replacement of multiple apples that are geographically close together. 4. User can click on a tree will bring the user into a zoomed mode and display the apple trees that were very close to each other as apple icons once more. 5. User can then select any of the marked apple tree locations by clicking or tapping on one of the apple icons. 6. User can hover on an apple icon will bring up a small information card about the apple tree selected.

	7. User can click on the information card to bring up a side bar that expands fully on all the information available for that specific apple tree. 8. User can select the filter button on the top right of the screen to filter apple trees. 9. User can select a desired filter from the drop down menu and the applied filter will only show those trees that fit the selected filter requirements specified by the user.
Alternative Flows:	*None*
Extension Points:	*None*
Preconditions:	Application must be fully loaded on a computer or phone device with an internet connection.
Postconditions:	Application successfully loads on device for user to interact with the application.
Business Rules:	*None*

6. The Class Model and DB Schema

6.1 The Class Model



6.2 DB Schema

DB Schema: Our entire DB schema is created with the command found in the glossary. This is the only table we foresee ourselves needing, unless we need to have administrative users.

1. Postgres Table Creation Statement:

```
create table applesprimary(
  TreeTagId serial PRIMARY KEY,
  genetics VARCHAR (50),
  species VARCHAR (50),
  finalCultivar VARCHAR (50),
  synonymText VARCHAR (300),
  isConfirmed VARCHAR (50),
  use VARCHAR (50),
  country VARCHAR (50),
  genotypes float,
  property VARCHAR (50),
  location VARCHAR (500),
  lattitude float8,
  longitude float8,
  height real,
  dripline real,
  diameter real,
  fireBlight real,
  fruitHanging varchar (50)
);
```

7. Appendix

Glossary

1. Detailed Design Specifications:
    a. Backend:
        i. AWS - currently testing with free tier
            1. EC2
                a. Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0dacb0c129b49f529 (64-bit x86) with Python3.6 and Django 3.0 stable, connected to RDS
            2. RDS
                a. PostgreSQL 12.1 Stable without AWS IAM
            3. S3 - filesystem
    b. Frontend:
        i. Angular
        ii. HTML5
        iii. CSS3
        iv. Leaflet