Software Requirements Specification

Version 1.0

December 4, 2015

Demeter’s Tome System

Jackie Golden

Tyler Hunnefeld

Laura Ann Truax

Colton Tuck

# Table of Contents

[Table of Contents](#h.1fob9te)

[1.0. Introduction](#h.tyjcwt)

[1.1. Purpose](#h.3dy6vkm)

[1.2. Scope of Project](#h.vps652i9h761)

[1.3. Glossary](#h.w5dvsrij802e)

[1.4. References](#h.2s8eyo1)

[1.5. Overview of Document](#h.xj3zuwimxwzv)

[Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.2.0. Overall Description](#h.3rdcrjn)

[2.1](#h.hzdupuwnmlyo)

[System Environment](#h.26in1rg)

[2.2 Functional Requirements Specification](#h.f6wak7ypi3rt)

[2.2.1 Non-Member Use Case](#h.o6cnsocvp1tf)

[Use case: Non-Member Diagram](#h.vomivp2un3q3)

[Non-member use case: View a plant page](#h.4fu9qkr8znyw)

[Brief Description](#h.46zubzjln96d)

[Initial Step-By-Step Description](#h.iy5xuyeanci0)

[Non-member use case: Sign up for an account](#h.qncywja9zcad)

[Brief Description](#h.lrlly6t8hgjd)

[Initial Step-By-Step Description](#h.r7izccs64ogj)

[2.2.2 Member Use Case](#h.yjxb6lhemqdq)

[Use case: Member Diagram](#h.vapbeq9volam)

[Member use case: Log in](#h.nnhcy1aa2l19)

[Brief Description](#h.cmf53bwoj7f7)

[Initial Step-By-Step Description](#h.1af9ksdcunhm)

[Member use case: Edit a plant page](#h.awnvynp0jes9)

[Brief Description](#h.hp7mpo88qiou)

[Initial Step-By-Step Description](#h.8lbpy1x430na)

[Member use case: Add a plant to a list](#h.c9oahtnbt658)

[Brief Description](#h.whr9p5142clr)

[Initial Step-By-Step Description](#h.svu8sy9zuiy3)

[Member use case: Compare plants for compatibility](#h.mj1dduv6zn50)

[Brief Description](#h.whqhbbg0fb5q)

[Initial Step-By-Step Description](#h.5zcu1n59s6sd)

[Member use case: Assume non-member role](#h.5y8txsr1kf2u)

[Brief Description](#h.n6f3287o3g9t)

[Initial Step-By-Step Description](#h.za3dqvennqxr)

[2.2.3 Web Administrator Use Case](#h.hkgh5f492xv)

[Use case: Web Administrator Diagram](#h.oaqkxxnh5pu8)

[Web admin use case: Log in](#h.9jyqj5p6llvc)

[Brief Description](#h.nryj5qo1f766)

[Initial Step-By-Step Description](#h.875cvju79ox3)

[Web admin use case: Freeze member](#h.63r46p7qkw1q)

[Brief Description](#h.payimclrt7pw)

[Initial Step-By-Step Description](#h.mqr5vf4jr4hy)

[Web admin use case: Edit database areas that are not open for editing to members](#h.dqu4pzkr69ni)

[Brief Description](#h.bc5r3y4chmpq)

[Initial Step-By-Step Description](#h.ud4olexn8sy2)

[Web admin use case: Assume member role](#h.7xi21izgwt53)

[Brief Description](#h.s50oggaw0gti)

[Initial Step-By-Step Description](#h.j8donj940y4p)

[Web admin use case: Assume non-member role](#h.ckjgoy8zg7f8)

[Brief Description](#h.3ti7q0r45k3t)

[Initial Step-By-Step Description](#h.sbbp6zj0myk3)

[2.2.4 System Administrator Use Case](#h.ys9zo3bq1kp0)

[Use case: System Administrator Diagram](#h.ex1ys1butvgh)

[System admin use case: Log in](#h.8ffxmqr5j10c)

[Brief Description](#h.nzqo3tl0xtx8)

[Initial Step-By-Step Description](#h.w149zslgs70j)

[System admin use case: Edit the web interface](#h.eml5r9fmyeai)

[Brief Description](#h.caabjt556nzr)

[Initial Step-By-Step Description](#h.l61w05i7nscu)

[System admin use case: Make changes to the database](#h.9r41i6nwtp8d)

[Brief Description](#h.e7nqopwyvb3f)

[Initial Step-By-Step Description](#h.vhwoxvu79p9v)

[System admin use case: Assume web admin role](#h.68ur1r1i6dve)

[Brief Description](#h.13f25y8e8xbj)

[Initial Step-By-Step Description](#h.gp9m5qth218f)

[System admin use case: Assume member role](#h.albv8xh0ycfg)

[Brief Description](#h.28ng90buif6q)

[Initial Step-By-Step Description](#h.raql6cb32i9g)

[System admin use case: Assume non-member role](#h.ai7879kaaqad)

[Brief Description](#h.wza0ut3yxzyj)

[Initial Step-By-Step Description](#h.3b2fbir0an9v)

[2.3 User Assumptions](#h.9ok5lnodi36y)

[2.4 Non-Functional Requirements](#h.dd0czbr8ghbp)

[3.0. Requirements Specification](#h.4i7ojhp)

[3.1 External Interface Requirements](#h.2xcytpi)

[3.2 Functional Requirements](#h.p1h20rk2lvye)

[3.3 Detailed Non-Functional Requirements](#h.qsh70q)

[3.3.1 Logical Structure of the Data](#h.afq772llujm8)

[3.3.2 Security](#h.49x2ik5)

[Appendix A. Index](#h.rlp4rf1nu1p5)

# 1.0. Introduction

## 1.1. Purpose

The purpose of this SRS is to provide a comprehensive description of the intended purpose and environment for the Demeter’s Tome (DT) web application.

## 1.2. Scope of Project

The DT system will be a database and accompanying web application to be used for organic companion planting information for the 48 contiguous states. This project is being done in hopes that it can be part of an interactive garden planner program. The database will include information about specific plant varieties, plant families, pests, diseases, growing zones, fertilizers, soil types, sun exposures, companion pants, and opponent plants. The web application will provide an elegant user interface for users to access the database.

The DT system is being developed to organize and simplify the companion planting garden planning process as well as to facilitate the sharing of a user’s knowledge with other users. It will serve as a centralized place to search and browse for plants, add and edit plants, store a user-specific list of plants, check the compatibility of two or more plants, and make a garden plan.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Plant Group | The family of plants that a cultivar belongs to. |
| Cultivar | A specific variety of plant. |
| Database | Collection of all the information monitored by this system. |
| Editor | Person who has edited a cultivar page. |
| Field | A cell within a form. |
| Non-Member | A person who can search and browse the database and sign up to be a member. |
| Member | A person who can log in and out, search and browse the database, edit specific fields in the database through the web interface, save a list of plants, add and remove plants from their lists, and use a list to compare the compatibility of plants. |
| Web Administrator | A person who has all of the capabilities of a member and can also freeze users and edit all fields in the database through the web interface. |
| Web Admin | Abbreviation for “web administrator”. See the ”Web Administrator” entry for more information. |
| System Administrator | A person who has access to all aspects of the DT system and can assume any role in the system. |
| Demeter’s Tome | The system that is being developed. |
| DT | Abbreviation for “Demeter’s Tome”. See the “Demeter’s Tome” entry for more information. |
| Software Requirements Specification | A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. |
| SRS | Abbreviation for “Software Requirements Specification”. See the “Software Requirements Specification” entry for more information. |
| User | Any system administrator, web administrator, member, or non-member. See corresponding entries for more information. |
| Plant profile page (cultivar profile page) | The page on the DT web application that displays all of the information in the database about a specific cultivar. |

## 1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

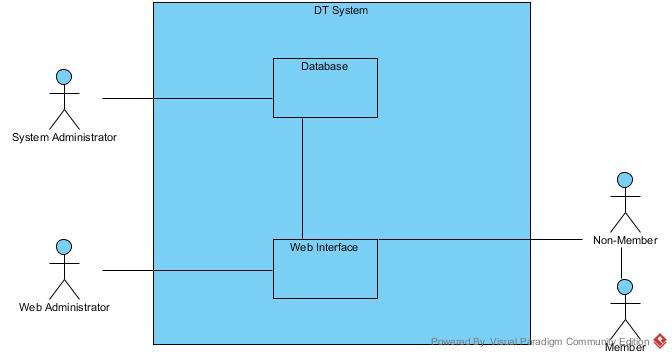
## 1.5. Overview of Document

The first chapter, the Introduction section, of this document contains information about the document.

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

2.0. Overall Description



**Figure 1 - System Environment**

## 2.1 System Environment

The DT system has the following actors: Non-member, member, web administrator, and system administrator. Any communications between users will be through their private email, but a messaging system may be added in later.

A non-member can access the web interface for the DT system without logging in. They may browse or search for plants as well.

A member can assume the role of a non-member by not logging in to or logging out of, their account. Upon logging in, a member will be able to add plants to the database or edit existing plant pages through the web interface. Additionally, a member can add plants to one or more custom lists. These lists can be used to check the compatibility of two or more plants. Upon compatibility checking, a report will be produced showing which plants are compatible, which are neutral, and which are not compatible. Additional functionality to produce a suggested planting order or garden plan may be added in the future.

A web administrator or web admin can assume the role of a non-member by not logging in to or logging out of the system. Upon logging in, a system administrator can edit or add information to any area of the database through the web interface and freeze users who are deemed to be an issue. This is in addition to having all of the capabilities of a member.

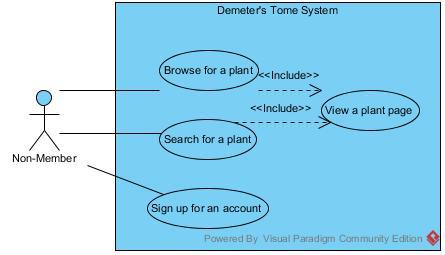
A system administrator can access all aspects of the DT system. A system administrator can also assume the role of a non-member by not logging in to or logging out of the system. In addition, a system administrator also has all of the capabilities of a web admin and therefore a member as well. They can also access the database directly, without the use of a web interface, for editing or additions and can edit the web interface.

## 2.2 Functional Requirements Specification

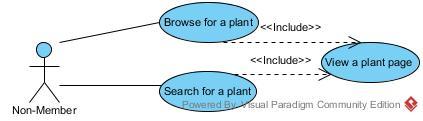
This section outlines the use cases for each of the types of user separately. Functionality increases for each type of user, starting with a non-member, then a member, a web administrator, and finally, a system administrator.

### **2.2.1 Non-Member Use Case**

#### Use case: Non-Member Diagram



#### Non-member use case: View a plant page



##### Brief Description

The non-member either searches for or browses for a plant.

##### Initial Step-By-Step Description

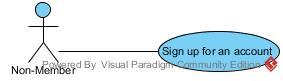
Before this use case can be initiated, the non-member has already connected to the Demeter’s Tome website.

1. The non-member selects the browse option
2. The DT system retrieves the plant categories from the GenusSpecies table and display them on a new page.
3. The non-member selects a category to view.
4. The system displays the cultivars in the category.
5. The non-member selects a cultivar to view.
6. The system displays the cultivar profile page (plant profile page).

Or

1. The non-member enters a string into the search bar.
2. The DT system displays a list of matching cultivars. If there are no matching cultivars, a message explaining that there are no matching cultivars is displayed instead.
3. If there are cultivars available, the non-member selects a cultivar to view.
4. The system displays the cultivar profile page (plant profile page).

#### Non-member use case: Sign up for an account



##### Brief Description

The non-member either searches for or browses for a plant.

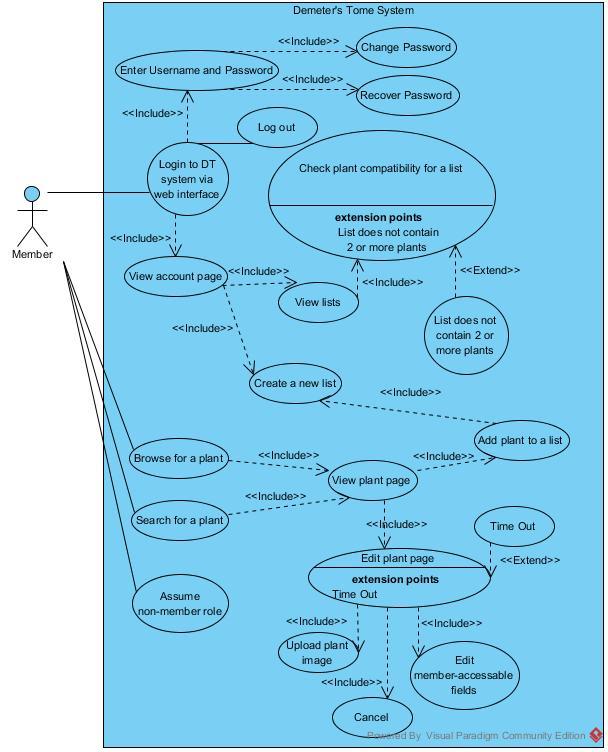
##### Initial Step-By-Step Description

Before this use case can be initiated, the non-member has already connected to the Demeter’s Tome website.

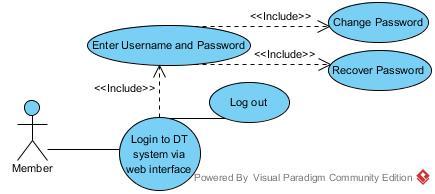
1. The non-member clicks the “Sign up for an account” link.
2. The DT system displays the sign up form.
3. The non-member fills out the form and submits it.
4. The DT system checks for errors in the form input.
   1. If the data is accepted, the system adds the user to the database.
   2. If the data is not accepted, the system displays an error message and allows the non-member to correct the entries. This will loop until input is accepted or the non-member exits the system, whichever happens first.
5. The DT system displays the new user profile page.

### **2.2.2 Member Use Case**

#### Use case: Member Diagram



#### Member use case: Log in



##### Brief Description

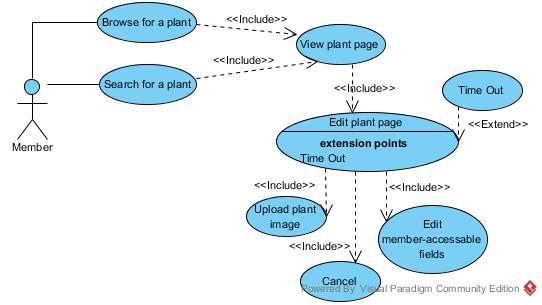
The member logs in to the DT system through the web application.

##### Initial Step-By-Step Description

Before this use case can be initiated, the member has already connected to the Demeter’s Tome website.

1. The member clicks on the “Sign in” link.
2. The DT system displays the sign in page.
3. The member enters their credentials and submits them.
4. The system checks the submitted data.
   1. If the credentials are accepted, the member profile page is displayed.
   2. If the credentials are not accepted, steps 3 and 4 loop until the credentials are accepted or until the member disconnects from the website, whichever happens first.

#### Member use case: Edit a plant page



##### Brief Description

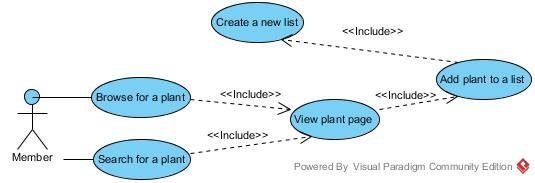
The member changes or adds to the information in the database about a specific cultivar through the web application.

##### Initial Step-By-Step Description

Before this use case can be initiated, the member has already connected to the Demeter’s Tome website and logged in.

1. The member searches or browses for a plant in the same way a non-member searches or browses.
2. Upon the display of a cultivar page, the member clicks the “Edit information about this plant” link.
3. The DT system displays the populated form for the cultivar.
4. The member edits or adds to the data in the form and submits the changes or cancels the transaction.
   1. If the data is submitted, the system validates the data and either accepts or denies the changes.
      1. If the changes are accepted, the transaction is written to the database and the user is redirected to the plant profile page.
      2. If the changes are not accepted, the user is redirected back to the form and an error message is displayed.
   2. If the transaction is canceled, the user is redirected to the previous plant profile page.

#### Member use case: Add a plant to a list



##### Brief Description

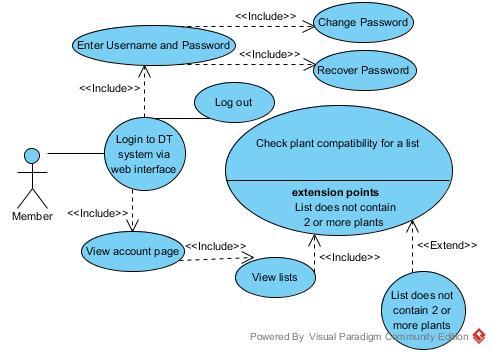
The member changes or adds to the information in the database about a specific cultivar through the web application.

##### Initial Step-By-Step Description

Before this use case can be initiated, the member has already connected to the Demeter’s Tome website and logged in.

1. The member views a plant page in the same fashion as a non-member.
2. The member clicks the “Add to a list” button.
3. The DT system displays all lists available to a user and an option to create a new list.
4. The member either selects the list(s) that they want to add the plant to, selects the “Create a new list” option, or cancels the transaction.
   1. If the user has elected to create a new list,
      1. The system displays a window with the “create a new list” form.
      2. The user inputs the name of the list and submits the data.
      3. The system validates the data, and either accepts or denies the list creation.
      4. If the data is accepted,
         1. The system adds a new list to the member’s account.
      5. If the data is not accepted,
         1. The system redirects the window to the previous form and displays an error message. This will repeat until the member has created a new list or cancels the transaction.
   2. If the member selects one or more lists to add the cultivar to and submits the data,
      1. The system adds the cultivar to each list that the member has chosen.
      2. The system redirects the member to the previous plant profile page.
   3. If the member elects to cancel the transaction,
      1. The system cancels the transaction and redirects the member to the previous plant profile page.

#### Member use case: Compare plants for compatibility



##### Brief Description

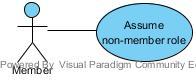
The member chooses a list with two or more plants to check the compatibility of the plants in the list. The system then displays the generated report to the member.

##### Initial Step-By-Step Description

Before this use case can be initiated, the member has already connected to the Demeter’s Tome website and logged in.

1. The member is either on their profile page or clicks the “View your profile” link.
2. The DT system displays the member’s profile page (which includes a list of the member’s lists).
3. The member selects a list to compare with itself and submits the data.
   1. If the list contains two or more cultivars, the DT system checks the cultivars on the list for compatibility and generates a report.
   2. If the list contains less than 2 cultivars, the system will display an error message until a valid list is selected or the member leaves the member profile page.
4. The system displays the report.
5. The member either saves the report as a PDF, prints the report, or closes the report.
   1. If the member elects to print the report, the system will send the document to the default printer.
   2. If the member elects to save the report, the system will export the document as a PDF.
   3. Steps a. and b. will repeat until the member decides to close the report.
6. The system closes the report and displays the member’s profile page.

#### Member use case: Assume non-member role



##### Brief Description

The member assumes the role of a non-member either by logging out or not logging in.

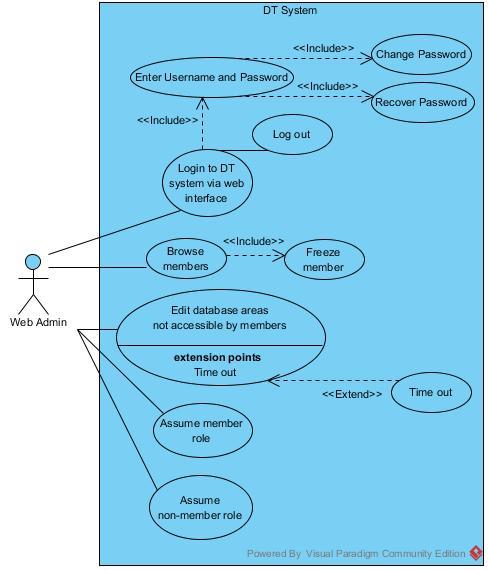
##### Initial Step-By-Step Description

Before this use case can be initiated, the member has already connected to the Demeter’s Tome website.

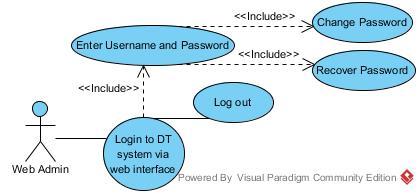
1. If the member has not logged in yet, they automatically use the system as a non-member until they either log in or disconnect from the DT system.
2. If the member has already logged in, they click the “Log out” link.
   1. The system logs the member out and they can automatically use the system as a non-member until they either log in again or disconnect from the DT system.

### **2.2.3 Web Administrator Use Case**

#### Use case: Web Administrator Diagram



#### Web admin use case: Log in



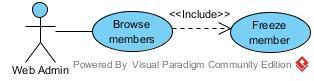
##### Brief Description

The web admin logs in to the DT system.

##### Initial Step-By-Step Description

See the Step-By-Step Description for the member use case: Log in (section 2.2.2)

#### Web admin use case: Freeze member



##### Brief Description

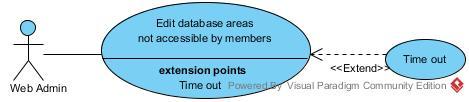
The web admin freezes a user for violating the rules for membership.

##### Initial Step-By-Step Description

Before this use case can be initiated, the web admin has already connected to the Demeter’s Tome website and logged in.

1. The web admin accesses their personal profile page.
2. The web admin clicks on the “Freeze a user” link.
3. The system displays a list of all users.
4. The web admin chooses a user and submits the data.
5. The system freezes the corresponding user account and redirects the web admin back to their personal profile page.

#### Web admin use case: Edit database areas that are not open for editing to members



##### Brief Description

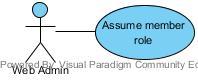
The web admin edits parts of the database other than the Cultivars section.

##### Initial Step-By-Step Description

Before this use case can be initiated, the web admin has already connected to the Demeter’s Tome website and logged in.

1. The web admin navigates to their personal profile page.
2. The web admin clicks the “Edit database tables” link.
3. The system displays all tables in the database.
4. The web admin chooses a table to edit.
5. The system displays all fields in the table, along with the option to edit any field (next to each field) and the option to add a new field.
6. If the web admin chooses to add a new field, the system will display a form for a new table entry.
   1. The web admin fills out the form and submits the data.
   2. The system validates the data and either accepts or denies it.
      1. If the data is accepted, the system adds the new field to the table.
      2. If the data is not accepted, the system redirects the web admin to the form and displays an error message. This repeats until the transaction times out, the transaction is canceled, or the transaction is accepted.
   3. The web admin is redirected back to the previous page in Step 5.
   4. If the web admin chooses to edit a field, the web admin clicks the “Edit field” link next to the chosen field.
   5. The system displays a form populated with the current data.
   6. The web admin either changes the data or cancels the transaction.
   7. If the data is submitted, the system validates the data and either accepts or denies it.
      1. If the data is accepted, the system updates the field.
      2. If the data is not accepted, the system redirects the web admin to the form and displays an error message. This repeats until the transaction times out, the transaction is canceled, or the transaction is accepted.
   8. The web admin is redirected back to the previous page in Step 5.

#### Web admin use case: Assume member role



##### Brief Description

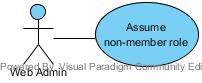
The web admin assumes the role of a member automatically upon logging in to the DT system.

##### Initial Step-By-Step Description

Before this use case can be initiated, the web admin has already connected to the Demeter’s Tome website.

1. The web admin logs in to the DT system
2. The web admin automatically has all member permissions as the set of permissions for a web admin is a proper superset of the member permissions.
3. Upon logging out, the web admin automatically assumes the role of a non-member.

#### Web admin use case: Assume non-member role



##### Brief Description

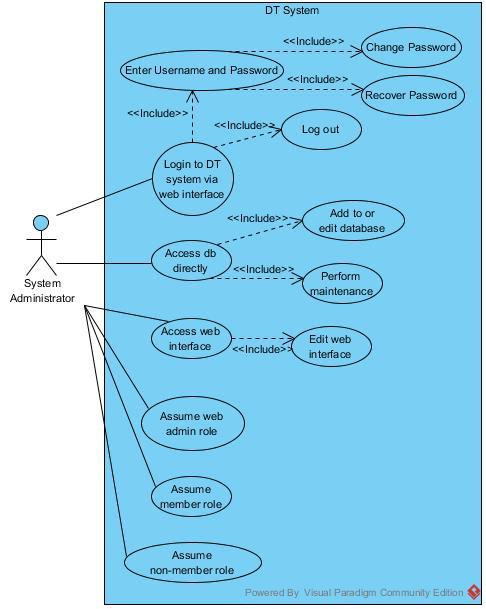
The web admin assumes the role of a non-member either by logging out or not logging in.

##### Initial Step-By-Step Description

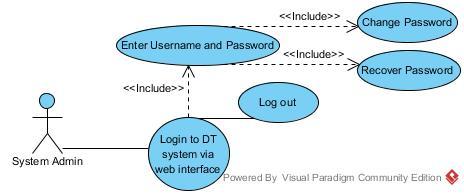
See the Step-By-Step Description for the member use case: Assume a non-member role (section 2.2.2)

### **2.2.4 System Administrator Use Case**

#### Use case: System Administrator Diagram



#### System admin use case: Log in



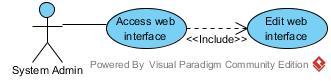
##### Brief Description

The system admin logs in to the DT system.

##### Initial Step-By-Step Description

See the Step-By-Step Description for the member use case: Log in (section ).

#### System admin use case: Edit the web interface



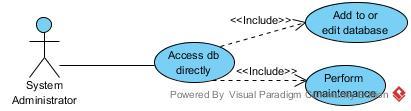
##### Brief Description

The system admin accesses the source of the web interface and makes changes to it.

##### Initial Step-By-Step Description

1. The system admin accesses source files for the DT system, makes changes, and tests the changes. When a new version on the web site is released, the system admin will upload the appropriate files.

#### System admin use case: Make changes to the database



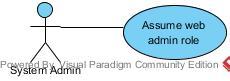
##### Brief Description

The system admin accesses the source of the database and makes changes to it.

##### Initial Step-By-Step Description

1. The system admin accesses the DT database directly, makes changes, and tests the changes.

#### System admin use case: Assume web admin role



##### Brief Description

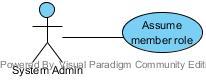
The system admin assumes the role of a web admin automatically upon logging in to the DT system.

##### Initial Step-By-Step Description

Before this use case can be initiated, the system admin has already connected to the Demeter’s Tome website.

1. The system admin logs in to the DT system
2. The system admin automatically has all member permissions as the set of permissions for a system admin is a superset of the web admin permissions.
3. Upon logging out, the system admin automatically assumes the role of a non-member.

#### System admin use case: Assume member role



##### Brief Description

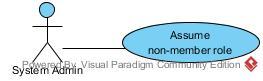
The system admin assumes the role of a member automatically upon logging in to the DT system.

##### Initial Step-By-Step Description

Before this use case can be initiated, the system admin has already connected to the Demeter’s Tome website.

1. The system admin logs in to the DT system
2. The system admin automatically has all member permissions as the set of permissions for a system admin is a proper superset of the member permissions.
3. Upon logging out, the system admin automatically assumes the role of a non-member.

#### System admin use case: Assume non-member role



##### Brief Description

The system admin assumes the role of a non-member either by logging out or not logging in.

##### Initial Step-By-Step Description

See the Step-By-Step Description for the member use case: Assume a non-member role (section )

## 2.3 User Assumptions

1. All users should be computer and internet literate and be able to use a search engine.
2. A web administrator should also have some gardening experience.
3. A system administrator should also have experience in MySQL, PHP, and HTML.

## 2.4 Non-Functional Requirements

The DT system will be on a server with high speed Internet capability. The software developed here assumes the use of a tool such as xampp for connection between the Web pages and the database. The speed of a user’s connection will depend on the hardware they use to access the system rather than characteristics of this system.

# 3.0. Requirements Specification

## 3.1 External Interface Requirements

There will be no links to an external system at this time.

## 3.2 Functional Requirements

In addition to the basic functions (insert, update, and delete records, searching and printing returned results) the application will be able to do the following:

1. The database can generate a list of suggested plants for a garden. To use this database in this way, the user will have to know one or more of their garden’s growing conditions.

* A user will be able to enter in their specific growing conditions (sun exposure that the garden gets, their USDA growing zone), and receive a list of compatible plants for their garden and information on how to plant them.

So far, the only similar databases that have been found can only search for plants by their scientific or common names. An example of a similar database is at the website Dave’s Garden (<http://davesgarden.com/guides/pf/adv_search.php?Search=Click+here+to+search+for+plants>). Garden planners are also available but do not offer the ability to customize the garden based on more than one factor, if any at all.

1. The database can check the compatibility of two or more plants.

* A user will be able to choose from 2 to 20 plants. The application will display if one is beneficial to the other, not beneficial to the other, or if there is no effect either way.
* It will also display why an interaction is either beneficial or not (if such information is in the database).

1. The database can suggest garden plans based on selected compatible plants.

* A user will choose up to 20 plants from the database. The plants will be sorted based on compatibility. If a plant is not compatible with any other plant in the list, it will not be included, but will be shown on a list of plants that were not included. Compatible plants will be given to the user in a table and a graphical representation of a row garden will be displayed.

1. The database can give a table of planting dates, growing times, and/or harvest dates for selected plants.

* A query will be run to find first and last frost dates, how many days before or after frost the plant should be planted, and whether or not the plant is a summer plant or winter plant.

1. The user can save lists of plants to use for searches.
2. Common plants will be included, but the user can add individual plants through a form. The user can also add additional common names to plants (as there may be several).

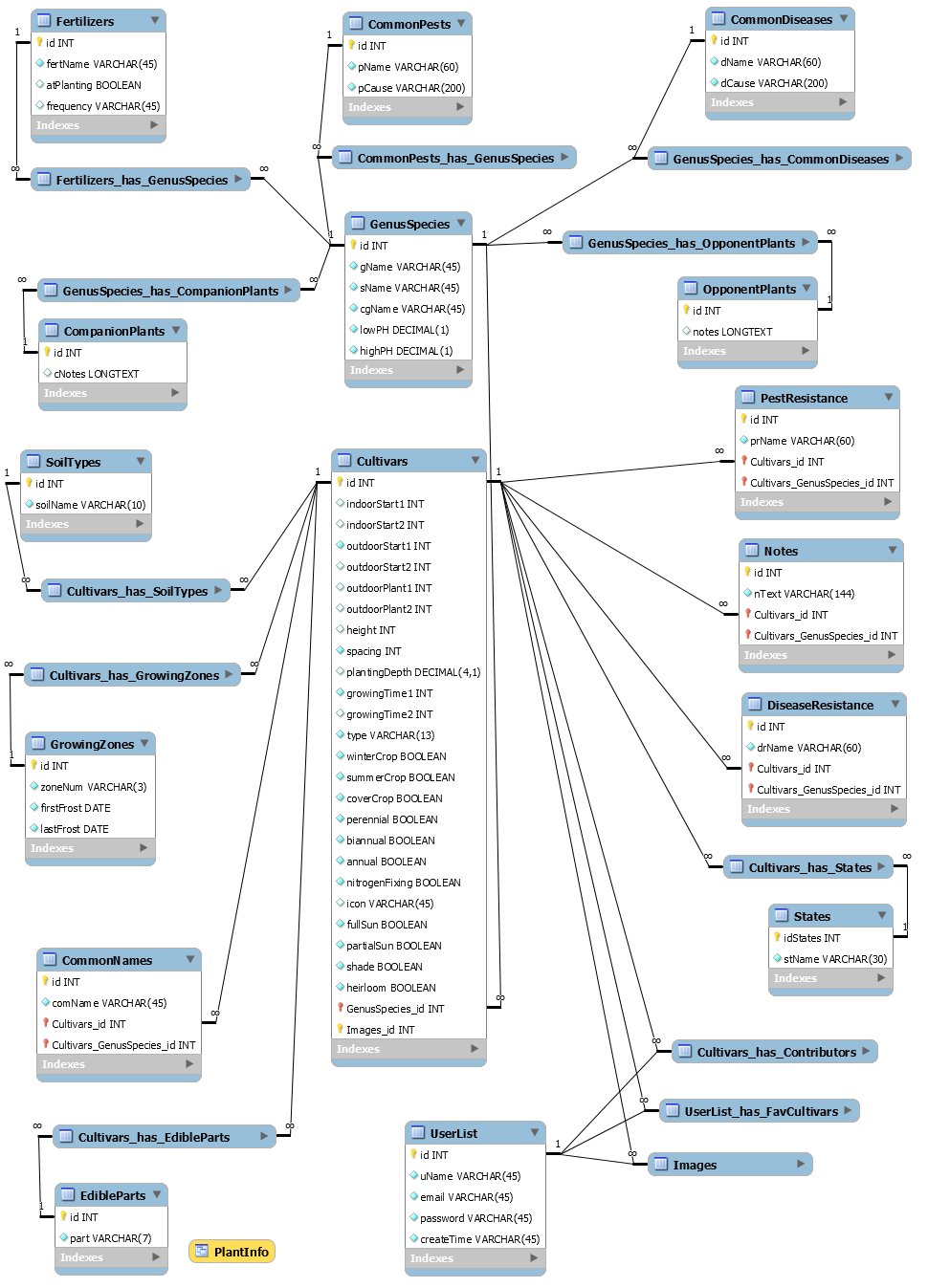
**Data notes:**

* Frost dates will be used to calculate when to plant seeds or transplants.
* Icons and images will be used to see what the plants actually look like and for the garden plans.
* Planting depth will be how deep in the soil a seed is to be planted.
* Growing times will be used to calculate when the plant will be ready for harvest.
* Winter crop and summer crop designations will be used to compute planting dates.
* Whether the plant is an annual, biannual, or perennial will be used for short-term and long-term garden planning.
* Whether the plant is nitrogen-fixing or not will determine if it is a good cover crop or companion for a garden that uses a substantial amount of nitrogen to grow.
* Common and scientific names will be used for searching.
* Common pests/diseases and individual plants’ pest/disease resistance will be used to determine whether a plant is well-suited for a garden.
* The plant “family” will actually be the genus. Gardeners often refer to these groups as the families.

## 3.3 Detailed Non-Functional Requirements

### 3.3.1 Logical Structure of the Data

The ER Diagram for the database portion of Demeter’s Tome is given below.



### 3.3.2 Security

The server on which the DT system resides will have its own security to prevent unauthorized *write*/*delete* access. There is no restriction on *read* access.

The PC on which the Article Manager resides will have its own security. Only the Editor will have physical access to the machine and the program on it. There is no special protection built into this system other than to provide the editor with *write* access to the Online Journal to publish an article.

# Appendix A. Index

|  |  |
| --- | --- |
| **Term** | **Page Number** |
| Actors  Assume Role  Cultivar - Add  Cultivar - Details  Database  Log in  Log Out  Schema  Search  Sign Up  System Admin  System Environment  User  User - Freeze  Web Admin  Web Administrator | **7**  **13**  **14**  **26**  **7**, 19, 26, 23  **7,** 12, 17, 18, 21, 22  **12**  **28**  5, 7, 9, 25, **26**  **10**  **21**, 22  **7, 8**  **5**, 8, 10  **10, 18**  **5**, 7, 8, **17**, **18**, **19**, 20, 22, 24  **5**, 7, 8, **17**, **18**, **19**, 20, 22, 24 |

Note: **Bold** page numbers feature the term heavily or help to define it