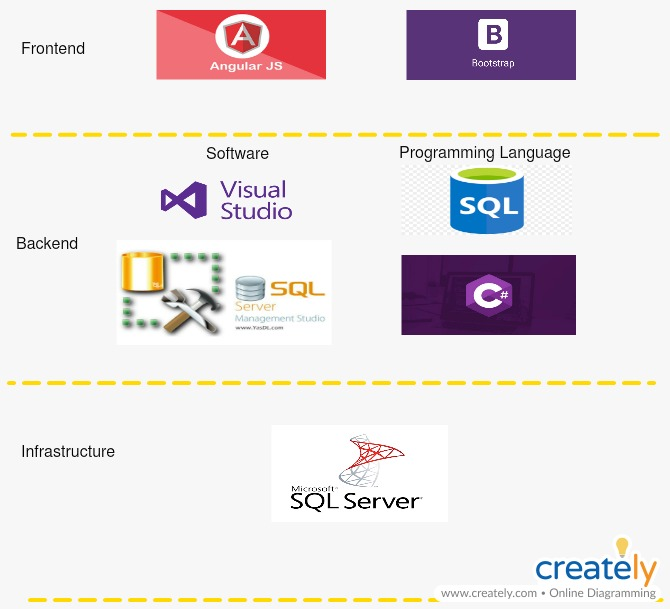
**Buffalo Botanical Gardens**

**Web Application**

Technical Design Document

# **1. Technology Stack**

**

# **Frontend**

# **AngularJS** is a structural framework for dynamic web applications. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application components clearly and succinctly. Its data binding and dependency injection eliminate much of the code you currently have to write.

# **Bootstrap** is a free and open source front end development framework for the creation of websites and web apps. The Bootstrap framework is built on HTML, CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps.

# **Backend**

# **Microsoft Visual Studio** is an Integrated Development Environment(IDE) developed by Microsoft to develop GUI(Graphical User Interface), console, Web applications, web apps, mobile apps, cloud, and web services, etc. With the help of this IDE, you can create managed code as well as native code.

# **Microsoft SQL Server Management Studio (SSMS)** is an integrated environment to manage a SQL Server infrastructure. It provides a user interface and a group of tools with rich script editors that interact with SQL Server.

# **C#** is an object-oriented programming language from Microsoft that aims to combine the computing power of C++ with the programming ease of Visual Basic. C# is based on C++ and contains features similar to those of Java. C# is designed to work with Microsoft's . Net platform.

# **Structured Query Language (SQL)** is a standard computer language for relational database management and data manipulation. SQL is used to query, insert, update and modify data.

# **Infrastructure**

# **Microsoft SQL Server** is a relational database management system (RDBMS) that supports a wide variety of transaction processing, business intelligence and analytics applications in corporate IT environments.

# 

# **2. Accounts and Infrastructure**

## **2.1 Development**

We will be developing assigned modules and features on separate local machines and then after thorough testing on the development station mentioned below and then publish them - one module/feature at a time.

**Development Station :** DESKTOP-EG4AB0S

## **2.2 Production**

Once the modules/features of the Web Application are thoroughly tested and are ready for publishing, the server used for publishing will be 71.186.160.170.

And, the current version of the Web Application can be found at: <http://71.186.160.170/website12>

# **3. Data Sources, Models, Timing**

## **3.1 Data Sources**

This web application is going to act as an interface between the Plant database and the Gardens Staff/Public.

The primary data source are the plant details and images of these plants provided by the Gardens Staff. Only the garden staff can create and edit data in the Plants database.

## **3.2 Data Models and Structure**

## There are 3 Databases and 4 Tables:

## BotanicalApp Database has 2 tables namely LogSheet$ and PlantData.

1. Collections Database has a table named Collections.
2. Locations Database has a table named Locations.

## **TableName : PlantData - 40 columns**

USE [BotanicalApp]

GO

SELECT [ID]

,[PlantID]

,[Location]

,[Collection]

,[Family]

,[ScientificName]

,[Authority]

,[Genus]

,[CommonName]

,[GeographicOrigin]

,[Quantity]

,[Value]

,[YearAdded]

,[VarietyID]

,[VarietyCultivar]

,[Hybrid]

,[LastUpdated]

,[YearRemoved]

,[SeedParent]

,[PollenParent]

,[Description]

,[PlantType]

,[Flowers]

,[BloomTime]

,[Height]

,[Light]

,[Soil]

,[Water]

,[Zone]

,[SoilpH]

,[Propagation]

,[Invasive]

,[Tolerant]

,[GrowingConditions]

,[Medicinal]

,[Diseases]

,[ImageFilename1]

,[ImageFileName2]

,[ImageFileName3]

,[ImageFileName4]

,[ImageFileName5]

FROM [dbo].[PlantData]

GO

## **TableName : LogSheet$ - 8 columns**

USE [BotanicalApp]

GO

SELECT [Counter]

,[Workbook name]

,[Date/time of workbook ]

,[Sheet name]

,[Range Copied]

,[Range Pasted]

,[Time/Date of copy]

,[Copy/Paste successful]

FROM [dbo].['Log Sheet$']

GO

## **TableName :Collections - 2 columns**

USE [Collections]

GO

SELECT [ID]

,[Collections]

FROM [dbo].[Collections]

GO

## **TableName :Locations - 2 columns**

USE [Locations]

GO

SELECT [ID]

,[Location]

FROM [dbo].[Locations]

GO

## 

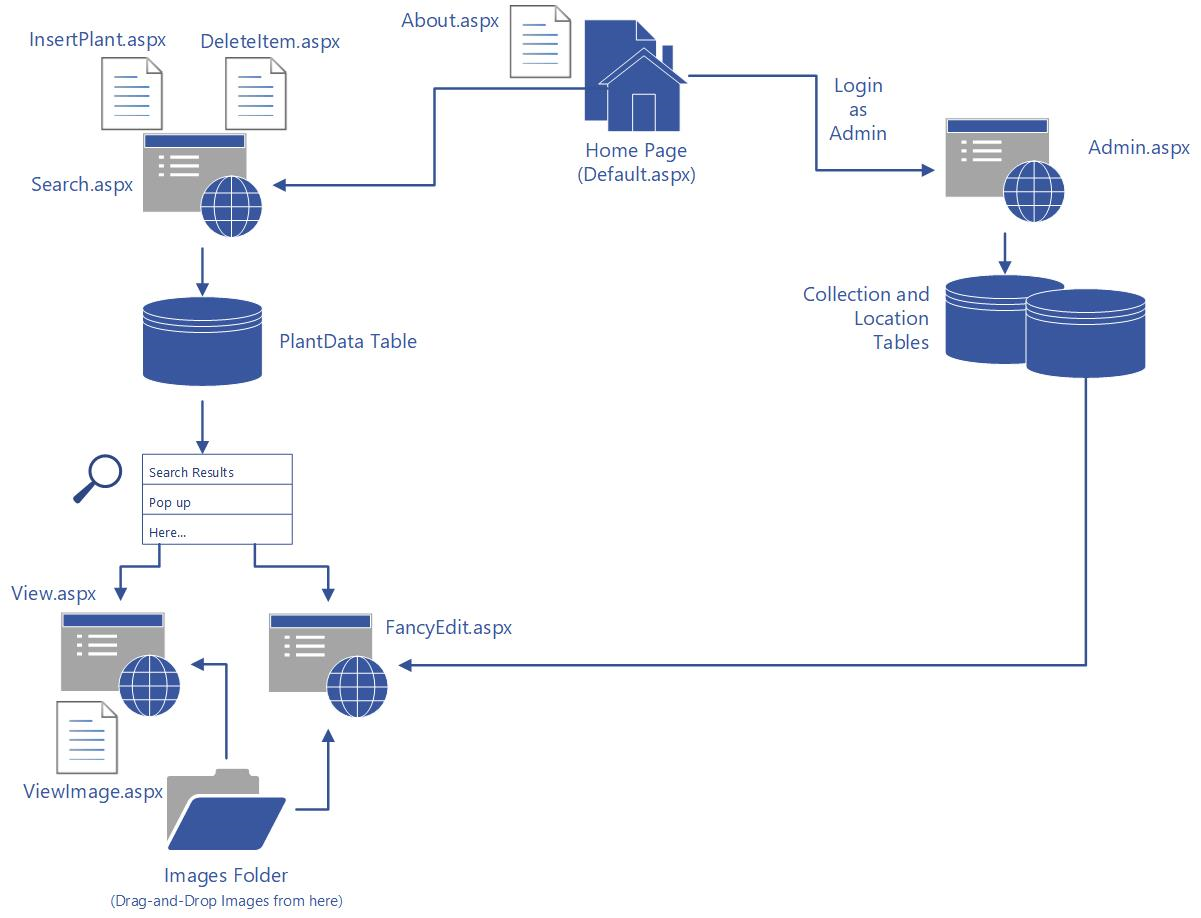
## **3.3 Timing**

Once entered into the Plant Database, data is never deleted. It is flagged as deleted by placing it into a special ‘Invisible Collection’, called ‘Deleted’. (A Collection is just what it says: a collection of plants in any one category like Palms, Ivy, Orchids, etc.). We make sure that Deleted plants are never displayed, although they remain in the database.

As for exports, data can be exported into .xlsx files as and when required.

# 

# **4. System Architecture Diagram**

**

# **5. Deployment Methodology**

**We have a simple step by step Deployment Methodology:**

1. Develop and debug new code independently on separate local machines. One of the machines will serve as the “Development station” - it would be responsible for all the testing and publishing.
2. Test within the Visual Basic environment.
3. “Publish” and test to a local server on the Development station, outside of Visual Studio.
4. Upload the code to the Gardens Server.
5. Create a new web version and debug the code in the Gardens Server Visual Studio environment.
6. Publish to the Gardens Server local server, outside of the Visual Studio environment.
7. Finally, publish to the public as whichever version we are up to.