|  |  |
| --- | --- |
|  | **GALLERIA E-ART**  **Technical Design Document** |
| |  |  |  |  | | --- | --- | --- | --- | |  | **Prepared By / Last Updated By** | **Reviewed By** | **Approved By** | | **Name** | 1. **Balaji Ravindaran**   **(2128548)**   1. **Gokul Raj K (2128799)** 2. **Jayapradaa G (2128402)** 3. **Sanjay Kumar S (2128447)** 4. **Sneka J (2150786)** |  |  | | **Role** | 1. Pod Leader  2. Pod Member  3. Pod Member  4. Pod Member  5. Pod Member |  |  | | **Signature** |  |  |  | | **Date** | 27-03-2023 |  |  | |
|  |

**Table of Contents**

[**1.0**](#_heading=h.1fob9te) **Introduction 3**

[**1.1**](#_heading=h.3znysh7) **Purpose of this document 3**

[**1.2**](#_heading=h.2et92p0) **Project overview 3**

[**2.0**](#_heading=h.tyjcwt) **Solution Summary 3**

[**2.1**](#_heading=h.3dy6vkm) **Scope 3**

[**2.2**](#_heading=h.4d34og8) **Assumptions** 4

[**2.3**](#_heading=h.2s8eyo1) **Dependencies** 4

[**2.4**](#_heading=h.17dp8vu) **Risks** 4

[**3.0**](#_heading=h.3rdcrjn) **Schematic Diagram 5**

**3.1 Administrator Sequence Diagram 5**

**3.2 User Sequence Diagram 6**

**3.3 Artist Sequence Diagram 7**

**3.4 User Case Diagram 8**

[**4.0**](#_heading=h.26in1rg) **System Design** 8

[**4.1**](#_heading=h.lnxbz9) **Proposed design** 8

[**4.2**](#_heading=h.35nkun2) **Component inventory** 9

[**5.0**](#_heading=h.1ksv4uv) **Database Design** 10

[**5.1**](#_heading=h.44sinio) **Data Model** 10

[**5.2**](#_heading=h.z337ya) **Tables Structure** 11

[**6.0**](#_heading=h.3j2qqm3) **Appendices** 13

[**6.1**](#_heading=h.1y810tw) **Glossary** 13

[**7.0**](#_heading=h.2xcytpi) **Terms & Conditions** 13

[**8.0**](#_heading=h.1ci93xb) **Change Log** 13

# Introduction

## Purpose of this document

The purpose of this document is to document the technical design, component details, and Database design. This will also capture the scope, assumptions, risk, and dependencies of this project.

## Project overview

Galleria – E art shows the work of many artists. The online art gallery is where the arts or art crafts of the artists can be displayed online for the visitors view. The art uploaded by the artist are seen by the users who register with the art gallery. So that there will be some scope of improvement of the arts that is displayed and the artist can get some recognition in the art world.

However, these days because of the busy schedule of the people, they don’t have time schedule-wise to visit the art gallery to view the paintings of the famous artist. Yet, imagine a scenario where the artist displays is available on the online indeed, it is conceivable.

The online art gallery is the application that enables the artists to upload their paintings and make it available to the visitors view. The artist after logging into their account can have the special feature of art management. It gives access to a facility through which the admin can monitor the whole system. Along with this, a facility where the users, after logging into their accounts can view the complete art inventory. The admin after logging into their account can have the facility of member management.

Overall, this project of ours has been designed to exhibit the artist’s work available online to the visitors view.

# Solution Summary

## Scope

Online art gallery management system is aimed to manage data of a series of art galleries. This system is used to keep records about artists, paintings and art galleries. The scope of this project consists of the successful creation of a fully developed art gallery management system using ASP.net and MS SQL server in order to assist the artist as well as the user to reduce the efforts that are generally spent on this tedious task of management. The platform manages and displays the art works uploaded by the artist so that the art lovers can view.

The artists can maintain their stock of uploaded painting details. The users and admin can see the details of all paintings by all the artists. This website is a platform to explore the art and craft designed by an artist.

## Assumptions

Assumptions for this project being,

* Creating a member signup page where new users can sign up for Galleria access.
* Creating a member login page where existing users can login.
* Creating a page where Members can manage their profile, and update their details
* Creating a page where we can view the available Paintings.
* Creating a page where admins can login.
* Creating a page where admins can validate the members
* Creating pages like Art management to manage (Add/Update/Delete) paintings.

## Dependencies

Technology used -

* ASP.NET with C# programming
* MS-SQL Database
* Bootstrap

Software Used -

* Visual Studio 2019
* SQL Server Management Studio (SSMS)
* Browser (Chrome, Edge, etc.)

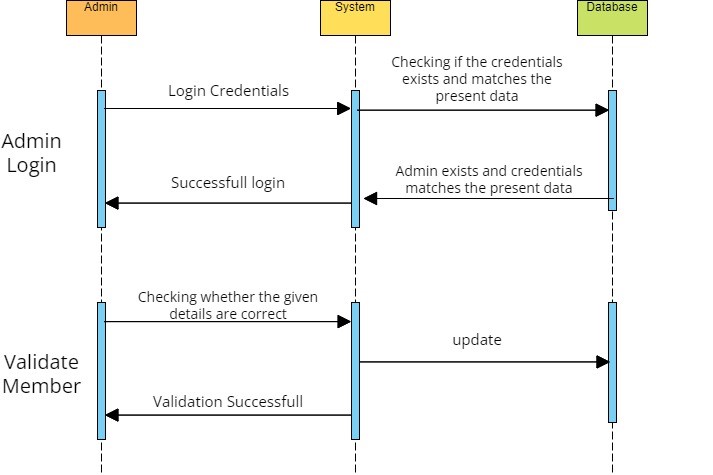
## Risks

Considering real-world scenarios, risks for this project being several factors such as

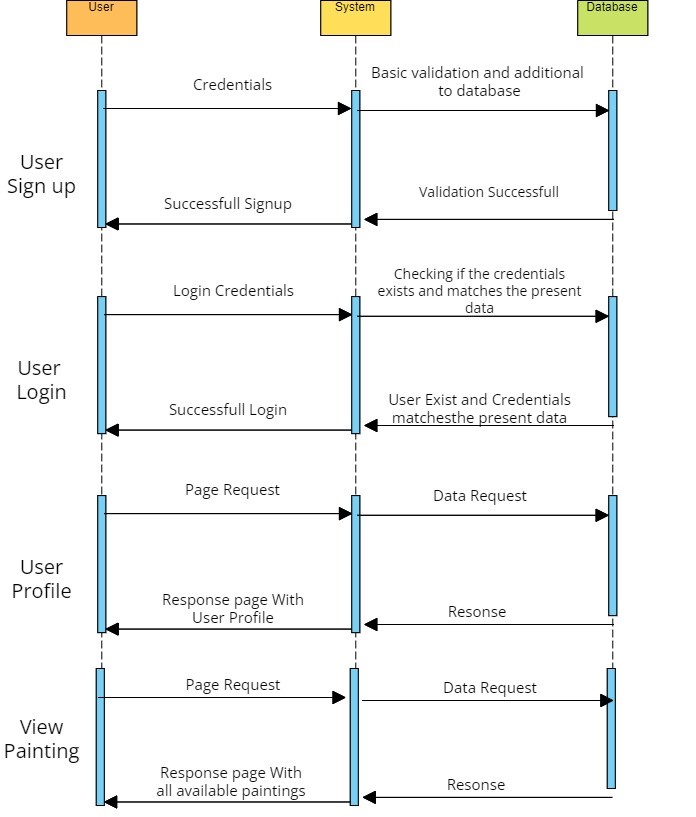
* Broken Authentication and Session Management.
* Security Misconfiguration.
* Unvalidated Redirects and Forwards.
* Connection Failures.
* Vulnerabilities to SQL injections.

# Schematic Diagram

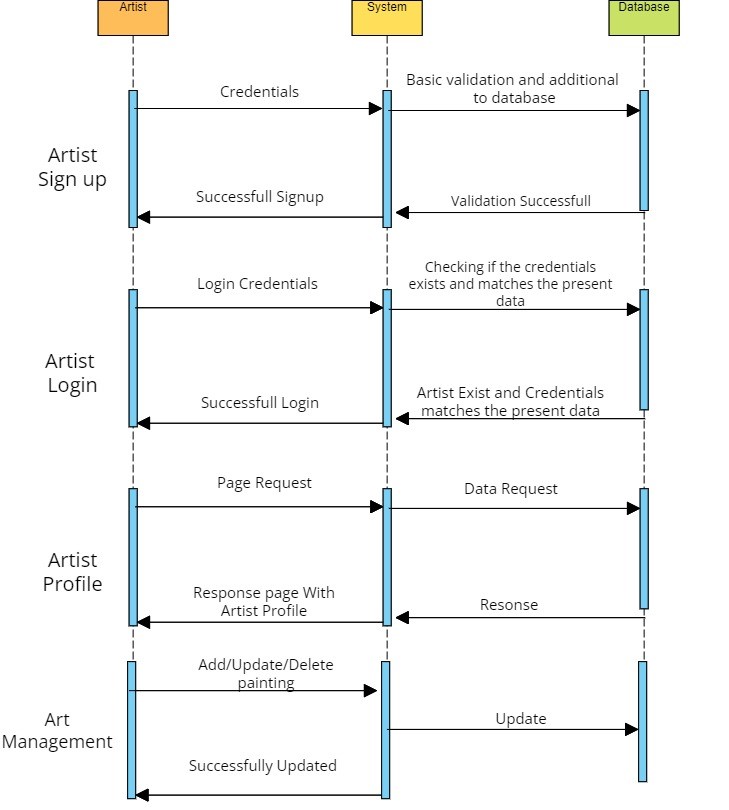
## Administrator Sequence Diagram



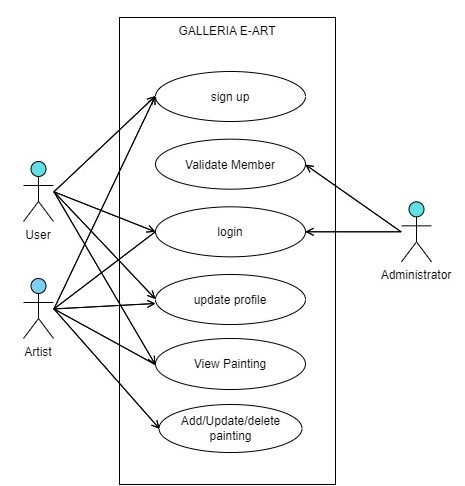
## User Sequence Diagram



## Artist Sequence Diagram



## User Case Diagram



# System Design

## Proposed design

The project is planned to use the ASP.net web application framework as the core working environment for producing dynamic web pages with different functionalities.

Where the data created during the user interactions in the front-end pages such as creating a new account etc, are stored in a back-end SQL Server database. Since there are different functionalities to this project there are to be different tables created under a single database, where each table carries its own data, for example, a dedicated table only contributing to saving the data about user, and another table only contributing to saving the data about Artist, etc.

The User Interface (UI) is proposed to be designed with the help of bootstrap considering the aesthetics and responsiveness of the dynamic pages, One particular page where the available paintings can be viewed in a tabular format is considered to be designed with the help of “datatables” which is a plugin for jQuery dedicated to creating responsive tabular content in web pages.

## Component inventory

Components for this project are

Technologies,

* ASP.net framework for creating dynamic web pages with C#.
  + Static Home page: Home page for the web application.
  + Member registration: For the user to register as a member and use the facilities.
  + Member login: After registering successfully, members can login into the system.
  + Member profile page: Members can manage their profile, and update their details
  + Admin login: Admin login page for the admins to enter the system and use the facilities.
  + Validation Page: Validate page for admins to check the details of the member and allow them.
  + Art Management Page: This page allows the artist to add paintings and perform CRUD operations.
  + View Paintings page: For the users to browse through the available paintings in the inventory.
* Bootstrap for responsiveness and aesthetics of web pages.
* SQL server for storing the data created by different functionalities, under the following tables,
  + Admin\_tbl (For managing different admins).
  + Member\_tbl (For managing different members)
  + Art\_Management\_tbl (For managing different arts).

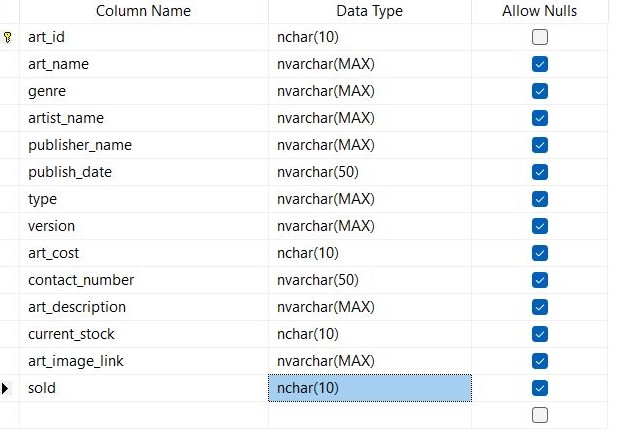
Software’s,

* Visual Studio 2019 for programming ASP.net web pages and hosting them.
* MSSQL Express 2019 for hosting the database on the local machine.
* SQL Server Management Studio (SSMS) for Visualizing the database.
* Browser (Chrome, Edge, etc.) for running/testing the web pages.

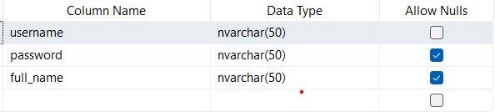
# Database Design

## Data Model

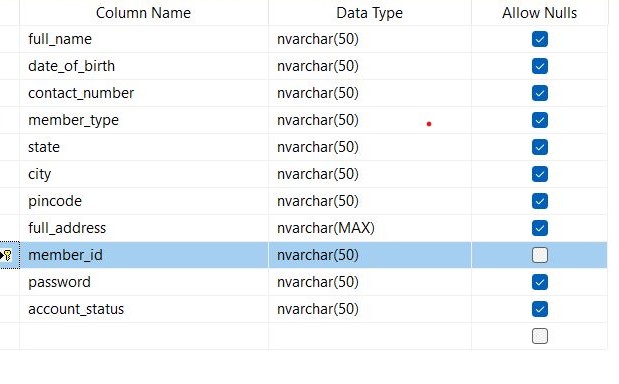
Art\_Management\_tbl



Admin\_tbl



Member\_tbl



## Tables Structure

Table 1: Admin\_tbl

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| username | nvarchar | 50 | NO |
| password | nvarchar | 50 | YES |
| full\_name | nvarchar | 50 | YES |

Table 2: Member\_tbl

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| full\_name | nvarchar | 50 | YES |
| date\_of\_birth | nvarchar | 50 | YES |
| contact\_number | nvarchar | 50 | YES |
| Member\_type | nvarchar | 50 | YES |
| State | nvarchar | 50 | YES |
| City | nvarchar | 50 | YES |
| Pincode | nvarchar | 50 | YES |
| full\_address | nvarchar | MAX | YES |
| member\_id | nvarchar | 50 | NO |
| Password | nvarchar | 50 | YES |
| account\_status | nvarchar | 50 | YES |

Table 3: Art\_Management\_tbl

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Nulls** |
| art\_id | nchar | 10 | NO |
| art\_name | nvarchar | MAX | YES |
| genre | nvarchar | MAX | YES |
| artist\_name | nvarchar | MAX | YES |
| publisher\_name | nvarchar | MAX | YES |
| publish\_date | nvarchar | 50 | YES |
| Type | nvarchar | MAX | YES |
| Version | nvarchar | MAX | YES |
| art\_cost | nchar | 10 | YES |
| contact\_number | nchar | 50 | YES |
| art\_description | nvarchar | MAX | YES |
| current\_stock | nchar | 10 | YES |
| art\_image\_link | nvarchar | MAX | YES |
| sold | nchar | 10 | YES |

# Appendices

## Glossary

|  |  |
| --- | --- |
| **Acronyms** | **Definitions** |
| ASP | Active Server Pages |
| SQL | Structured Query Language |
| MSSQL | Microsoft Structured Query Language |
| UI | User Interface |
| ETC | Etcetera |

# Terms & Conditions

***Disclaimer: Please do not circulate or distribute this document outside of Cognizant Network, We have a Zero Tolerance Policy. Kindly adhere to 100% Compliance at all times.***

# Change Log

*Please note that this table needs to be maintained even if a Configuration Management tool is used.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version Number** | **Changes made** | | | |
| V<n.n> | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes affected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |