### Name :-

# Event Handler

Web site for book event and manage event by admin and manage admin by super admin

This web site is develop on asp.net, c#

|  |  |
| --- | --- |
| Author |  |
| Title | Event Handler |
| Degree | Computer Science |
| Instructors |  |
| Technologies | Asp .net , c#, SQL server |

### ABSTRACT

This paper describes a new monitoring and event management concept. Event management is the application of the management science of project management to the creation and development of festivals and events. Event Handler Event Management System lets clients and staffs register venues and events in a simple manner.

From registering of venue to informing attendees, Event Handler Event Management System simplifies the reservations process, enabling staff to operate at peak efficiency. And, when everyone is working together with complete, accurate, real-time information, customers receive the best possible service. Generally, the project is about registering venues by vendors for events like meeting, exhibition, convocation etc and for clients to book venue for events. The staff will exist in registering attendees for particular events and send emails days before the event as reminder.

The design of the system was illustrated using several types of diagrams, namely Entity Relationship Diagram (ERD), Data Flow Diagram (DFD), and also Data Dictionary and Data Normalization. This project was written on c# and ASP.NET.

## 1.0 Introduction

Event management is the process by which an event is planned, prepared, and produced. I

I As with any other form of management, it encompasses the assessment, definition, acquisition, allocation, direction, control, and analysis of time, finances, people, products, services, and other resources to achieve objectives. An event manager's job is to oversee and arrange every aspect of an event, including researching, planning, organizing, implementing, controlling, and evaluating an event's design, activities, and production

#### Table 02. Operation Environment for the Design and Sign System

|  |  |
| --- | --- |
| PROCESSOR | Intel Core Processor |
| OPERATING SYSTEM | WINDOWS7 Or Higher |
| MEMORY | 1GB RAM Or More |
| HARD DISK SPACE | Minimum 30 GB for Database Usage for future |
| DATABASE | SQL Server 2019 |
| TECHNICAL SKILLS | JavaScript, CSS, HTML, ASP.NET |
| IDE | Visual Studio 2013 |
| WEB SERVER | IIS 7.0 |

### ****Module Specification:****

* ****Admin****
  + Manage Admin User.
  + Mange Customer.
  + Manage Event
  + Manage Package.
  + Manage Employee.
  + Manage Booking.
  + View Feedback.
  + Generate Various Report.
* ****Customer****
  + Manage Profile.
  + View Event.
  + View Package.
  + Book Package.
  + Cancel Package.
  + Give Feedback.
* ****Guest****
  + View Event.
  + View Package.
  + Give Feedback.

# Development Environment

## Front - end

### Asp.net

ASP.NET is an open source web framework, created by Microsoft, for building modern web apps and services with .NET.

ASP.NET is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web applications for PC, as well as mobile devices.

ASP.NET works on top of the HTTP protocol, and uses the HTTP commands and policies to set a browser-to-server bilateral communication and cooperation.

ASP.NET is a part of Microsoft .Net platform. ASP.NET applications are compiled codes, written using the extensible and reusable components or objects present in .Net framework. These codes can use the entire hierarchy of classes in .Net framework.

The ASP.NET application codes can be written in any of the following languages:

* C#
* Visual Basic.Net
* Jscript
* J#

ASP.NET is used to produce interactive, data-driven web applications over the internet. It consists of a large number of controls such as text boxes, buttons, and labels for assembling, configuring, and manipulating code to create HTML pages.

## ASP.NET Web Forms Model

ASP.NET web forms extend the event-driven model of interaction to the web applications. The browser submits a web form to the web server and the server returns a full markup page or HTML page in response.

All client side user activities are forwarded to the server for stateful processing. The server processes the output of the client actions and triggers the reactions.

Now, HTTP is a stateless protocol. ASP.NET framework helps in storing the information regarding the state of the application, which consists of:

* Page state
* Session state

The page state is the state of the client, i.e., the content of various input fields in the web form. The session state is the collective information obtained from various pages the user visited and worked with, i.e., the overall session state. To clear the concept, let us take an example of a shopping cart.

User adds items to a shopping cart. Items are selected from a page, say the items page, and the total collected items and price are shown on a different page, say the cart page. Only HTTP cannot keep track of all the information coming from various pages. ASP.NET session state and server side infrastructure keeps track of the information collected globally over a session.

The ASP.NET runtime carries the page state to and from the server across page requests while generating ASP.NET runtime codes, and incorporates the state of the server side components in hidden fields.

This way, the server becomes aware of the overall application state and operates in a two-tiered connected way.

## The ASP.NET Component Model

The ASP.NET component model provides various building blocks of ASP.NET pages. Basically it is an object model, which describes:

Server side counterparts of almost all HTML elements or tags, such as <form> and <input>.

Server controls, which help in developing complex user-interface. For example, the Calendar control or the Gridview control.

ASP.NET is a technology, which works on the .Net framework that contains all web-related functionalities. The .Net framework is made of an object-oriented hierarchy. An ASP.NET web application is made of pages. When a user requests an ASP.NET page, the IIS delegates the processing of the page to the ASP.NET runtime system.

The ASP.NET runtime transforms the .aspx page into an instance of a class, which inherits from the base class page of the .Net framework. Therefore, each ASP.NET page is an object and all its components i.e., the server-side controls are also objects.

## Components of .Net Framework 3.5

Before going to the next session on Visual Studio.Net, let us go through at the various components of the .Net framework 3.5. The following table describes the components of the .Net framework 3.5 and the job they perform:

|  |
| --- |
| **Components and their Description** |
| **(1) Common Language Runtime or CLR**  It performs memory management, exception handling, debugging, security checking, thread execution, code execution, code safety, verification, and compilation. The code that is directly managed by the CLR is called the managed code. When the managed code is compiled, the compiler converts the source code into a CPU independent intermediate language (IL) code. A Just In Time(JIT) compiler compiles the IL code into native code, which is CPU specific. |
| **(2) .Net Framework Class Library**  It contains a huge library of reusable types. classes, interfaces, structures, and enumerated values, which are collectively called types. |
| **(3) Common Language Specification**  It contains the specifications for the .Net supported languages and implementation of language integration. |
| **(4) Common Type System**  It provides guidelines for declaring, using, and managing types at runtime, and cross-language communication. |
| **(5) Metadata and Assemblies**  Metadata is the binary information describing the program, which is either stored in a portable executable file (PE) or in the memory. Assembly is a logical unit consisting of the assembly manifest, type metadata, IL code, and a set of resources like image files. |
| **(6) Windows Forms**  Windows Forms contain the graphical representation of any window displayed in the application. |
| **(7) ASP.NET and ASP.NET AJAX**  ASP.NET is the web development model and AJAX is an extension of ASP.NET for developing and implementing AJAX functionality. ASP.NET AJAX contains the components that allow the developer to update data on a website without a complete reload of the page. |
| **(8) ADO.NET**  It is the technology used for working with data and databases. It provides access to data sources like SQL server, OLE DB, XML etc. The ADO.NET allows connection to data sources for retrieving, manipulating, and updating data. |
| **(9) Windows Workflow Foundation (WF)**  It helps in building workflow-based applications in Windows. It contains activities, workflow runtime, workflow designer, and a rules engine. |
| **(10) Windows Presentation Foundation**  It provides a separation between the user interface and the business logic. It helps in developing visually stunning interfaces using documents, media, two and three dimensional graphics, animations, and more. |
| **(11) Windows Communication Foundation (WCF)**  It is the technology used for building and executing connected systems. |
| **(12) Windows CardSpace**  It provides safety for accessing resources and sharing personal information on the internet. |
| **(13) LINQ**  It imparts data querying capabilities to .Net languages using a syntax which is similar to the tradition query language SQL. |

## The .NET platform

.NET is a developer platform made up of tools, programming languages, and libraries for building many different types of applications.

The base platform provides components that apply to all different types of apps. Additional frameworks, such as ASP.NET, extend .NET with components for building specific types of apps.

Here are some things included in the .NET platform:

* ****The C#, F#, and Visual Basic programming languages****
* ****Base libraries**** for working with strings, dates, files/IO, and more
* ****Editors and tools**** for Windows, Linux, macOS, and Docker

## ASP.NET extends .NET

ASP.NET extends the .NET platform with tools and libraries specifically for building web apps.

These are some things that ASP.NET adds to the .NET platform:

* ****Base framework for processing web requests in C# or F#****
* ****Web-page templating syntax****, known as Razor, for building dynamic web pages using C#
* ****Libraries for common web patterns****, such as Model View Controller (MVC)
* ****Authentication system**** that includes libraries, a database, and template pages for handling logins, including multi-factor authentication and external authentication with Google, Twitter, and more.
* ****Editor extensions**** to provide syntax highlighting, code completion, and other functionality specifically for developing web pages

## Back-end code

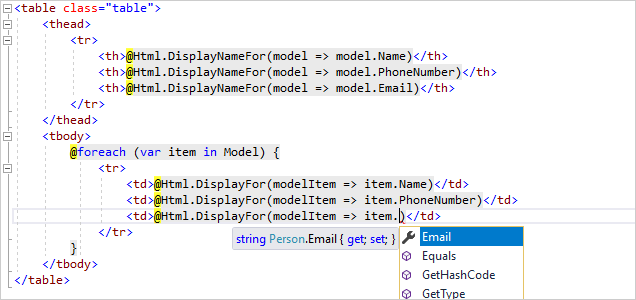
When using ASP.NET your back-end code, such as business logic and data access, is written using C#, F#, or Visual Basic.

Because ASP.NET extends .NET, you can use the large ecosystem of packages and libraries available to all .NET developers. You can also author your own libraries that are shared between any applications written on the .NET platform.

## Dynamic pages using C#, HTML, CSS, and JavaScript

Razor provides a syntax for creating dynamic web pages using HTML and C#. Your C# code is evaluated on the server and the resulting HTML content is sent to the user.

Code that executes client-side is written in JavaScript. ASP.NET integrates with JavaScript frameworks and includes pre-configured templates for single page app (SPA) frameworks like React and Angular.



## What is ASP.NET Core?

If you use ASP.NET, you'll soon come across the term ASP.NET Core.

ASP.NET Core is the open-source and cross-platform version of ASP.NET. You should use ASP.NET Core for all new applications. The tutorials on this site all use ASP.NET Core.

## Back-end

## SQL Server

SQL Server is a database server by Microsoft.

The Microsoft relational database management system is a software product which primarily stores and retrieves data requested by other applications. These applications may run on the same or a different computer.

Going more in-depth, in order to understand what a SQL Server is, you must first understand what SQL is.

SQL is a special-purpose programming language designed to handle data in a relational database management system.

A database server is a computer program that provides database services to other programs or computers, as defined by the client-server model. Therefore, a SQL Server is a database server that implements the Structured Query Language (SQL).

SQL SERVER is a relational database management system (RDBMS) developed by Microsoft. It is primarily designed and developed to compete with MySQL and Oracle database.

SQL Server supports ANSI SQL, which is the standard SQL (Structured Query Language) language. However, SQL Server comes with its own implementation of the SQL language, T-SQ

(Transact-SQL).

T-SQL is a Microsoft propriety Language known as Transact-SQL. It provides further capabilities of declaring variable, exception handling, stored procedure, etc.

SQL Server Management Studio (SSMS) is the main interface tool for SQL Server, and it supports both 32-bit and 64-bit environments.

## **Version History SQL Server**

* Microsoft and Sybase released version 1.0 in 1989.
* However, the partnership between these two ended in the early 1990s.
* Microsoft maintained ownership rights to the name SQL Server.
* Since the 1990s, subsequent versions of SQL Server have been released including SQL Server 2000, 2005, 2008, 2012, 2014, 2016, 2017, and 2019

## **SQL Server Editions**

****SQL Server Enterprise:****It is used in the high end, large scale and mission Critical business. It provides High-end security, Advanced Analytics, Machine Learning, etc.

****SQL Server Standard:****Itis suitable for Mid-Tier Application and Data marts. It includes basic reporting and analytics.

****SQL Server WEB:****It is designed for a low total-cost-of-ownership option for Web hosters. It provides scalability, affordability, and manageability capabilities for small to large scale Web properties.

****SQL Server Developer:****It is similar to an enterprise edition for the non-production environment. It is mainly used for build, test, and demo.

****SQL Server Express:****It is for small scale applications and free to use

## **Key Components and Services of SQL Server**

****Database Engine:****This component handle storage, Rapid transaction Processing, and Securing Data.

****SQL Server:****This service starts, stops, pauses, and continues an instance of Microsoft SQL Server. Executable name is sqlservr.exe.

****SQL Server Agent:****It performs the role of Task Scheduler. It can be triggered by any event or as per demand. Executable name is sqlagent.exe.

****SQL Server Browser:****This listens to the incoming request and connects to the desired SQL server instance. Executable name is sqlbrowser.exe.

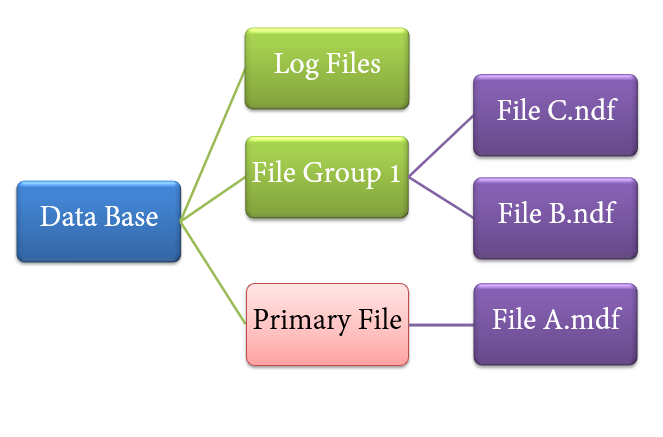
****SQL Server Full-Text Search:****This lets user running full-text queries against Character data in SQL Tables.Executable name is fdlauncher.exe.

****SQL Server VSS Writer:****This allows backup and restoration of data files when the SQL server is not running.Executable name is sqlwriter.exe.

****SQL Server Analysis Services (SSAS):****Provide Data analysis, Data mining and Machine Learning capabilities. SQL server is integrated with R and Python language for advanced analytics. Executable name is msmdsrv.exe.

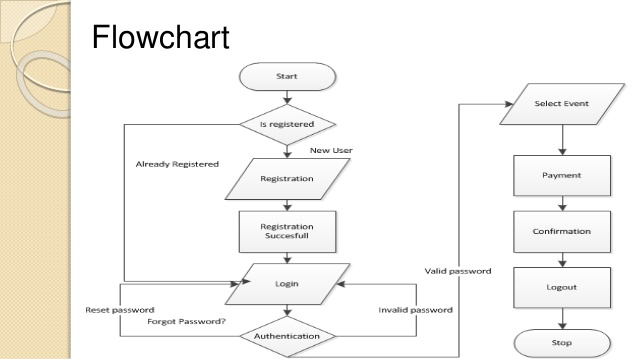
****SQL Server Reporting Services (SSRS):****Provides reporting features and decision-making capabilities. It includes integration with Hadoop. Executable name is ReportingServicesService.exe

****SQL Server Integration Services (SSIS):****Provided Extract-Transform and Load capabilities of the different type of data from one source to another. It can be view as converting raw information into useful information. Executable name is MsDtsSrvr.exe



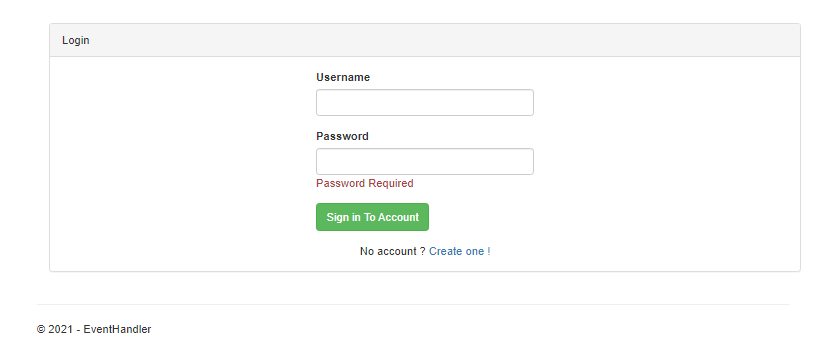
## Event Handler Architecture

1. Flow Chart

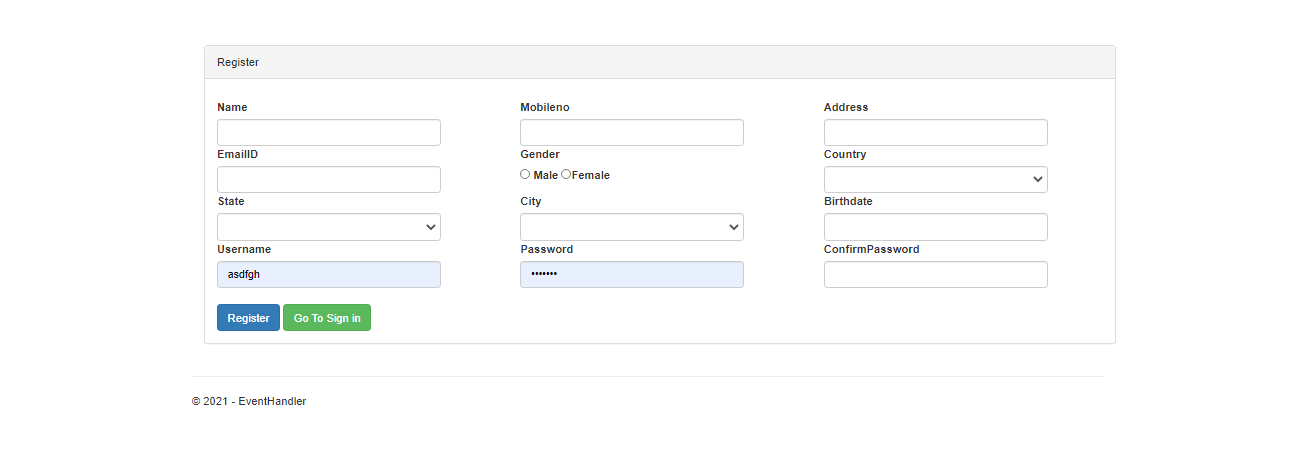


Event Handler Front-end :-

1. Log in Page

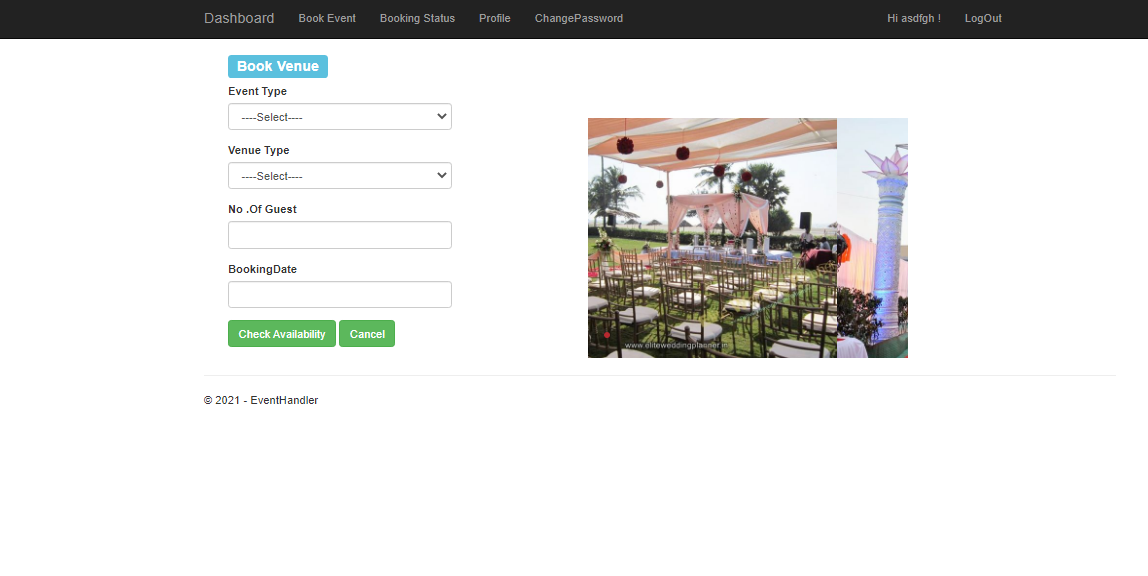


1. Register Page

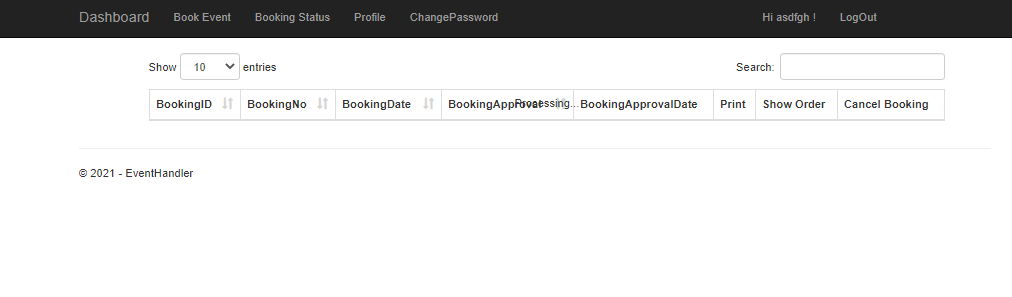


#### End user Interface

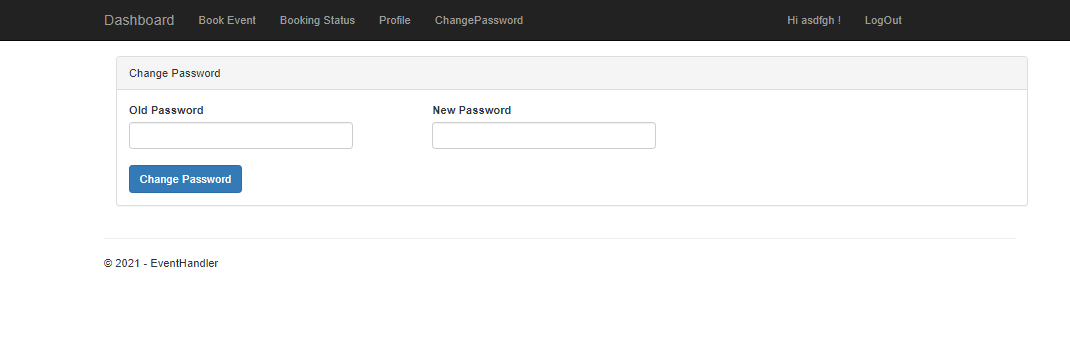
1. Event Booking Page



1. Booking status page

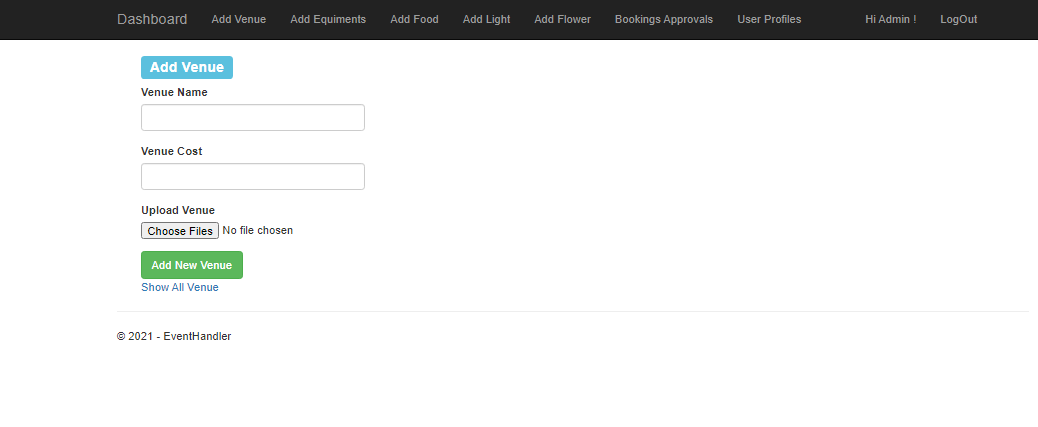


1. For changing Password

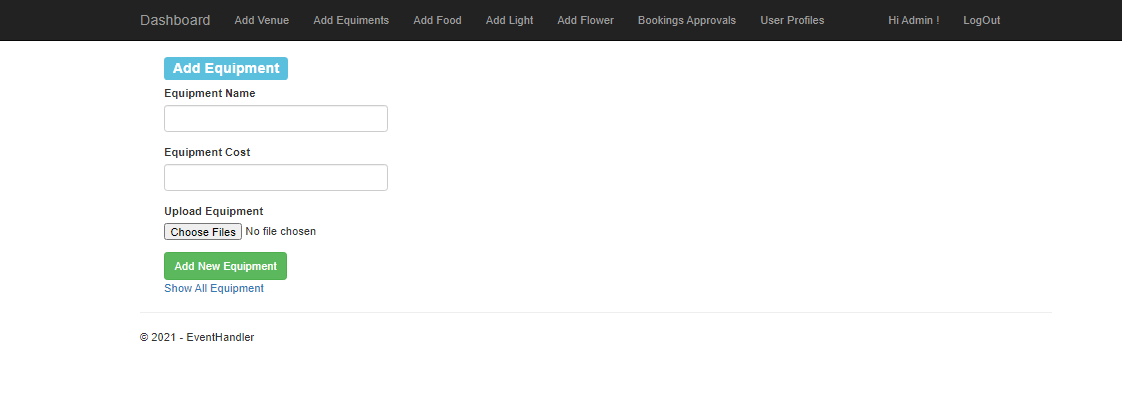


* Admin User Interface

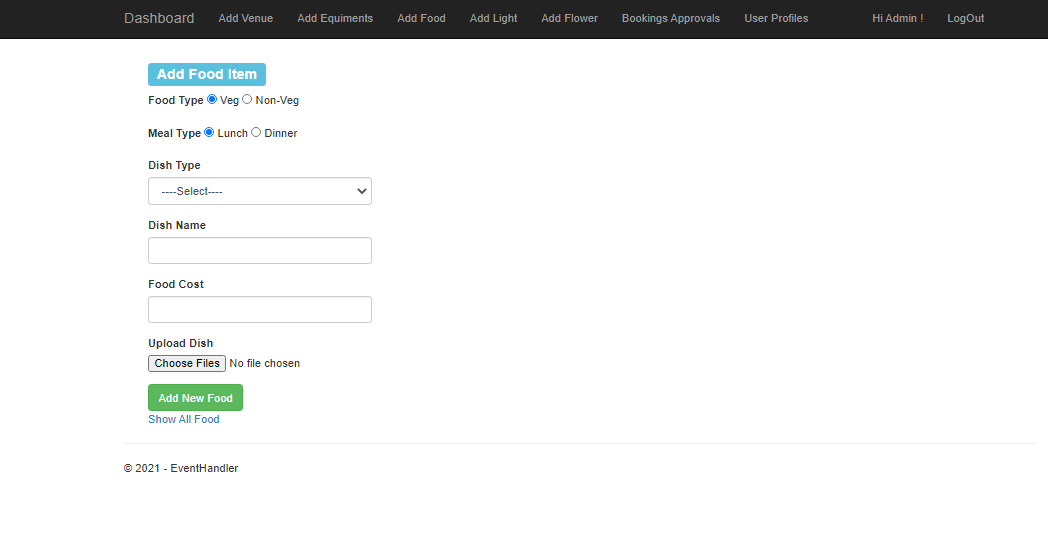
1. For adding new venue



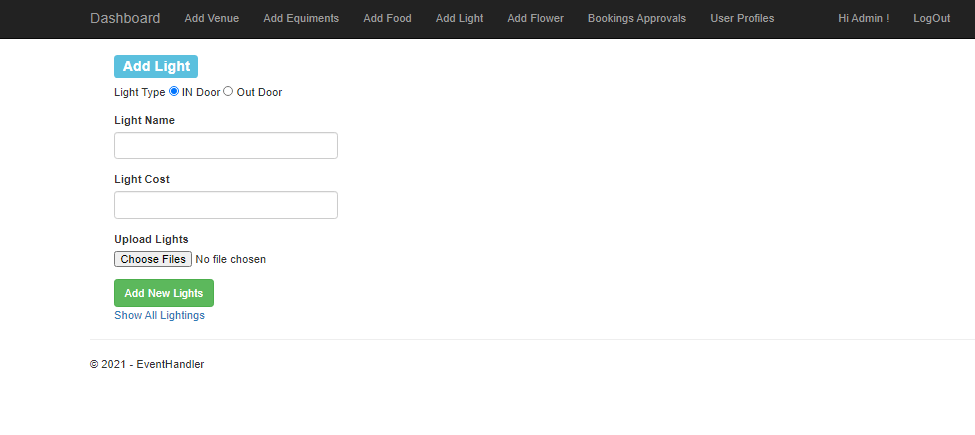
1. For adding Equipment



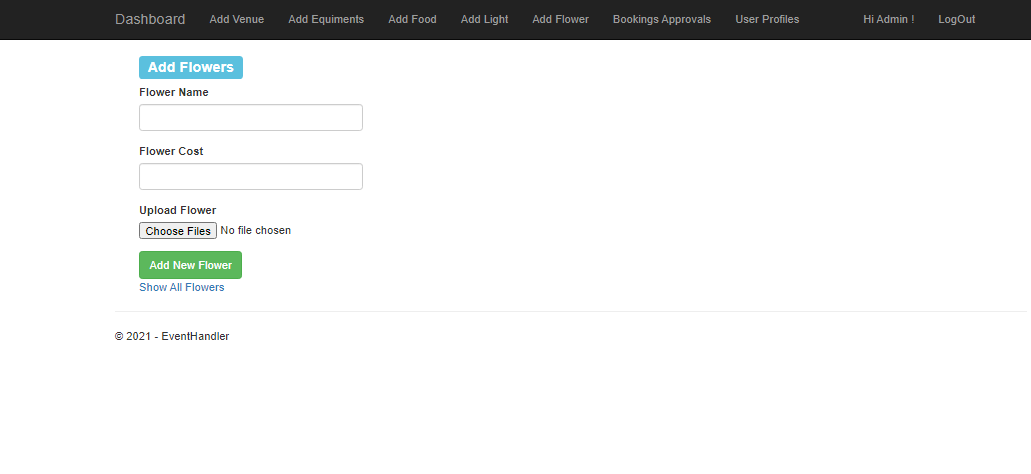
1. For adding Food Item



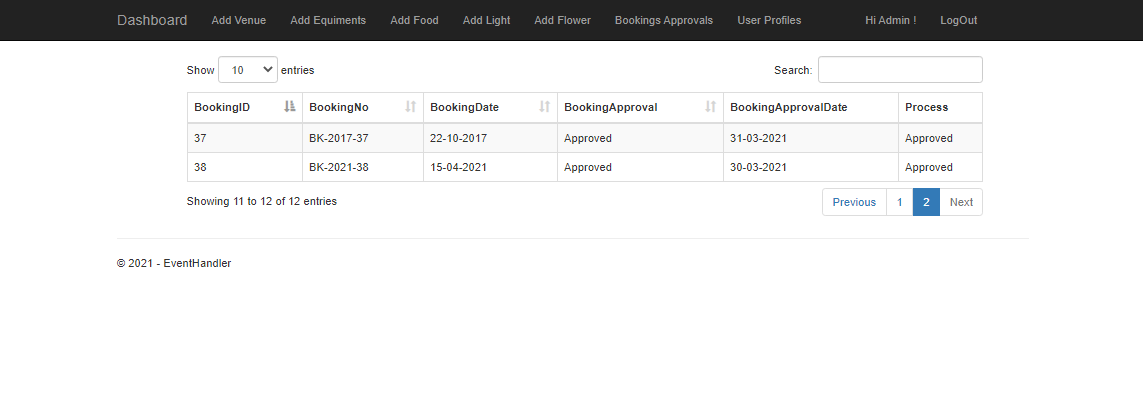
1. For adding new light



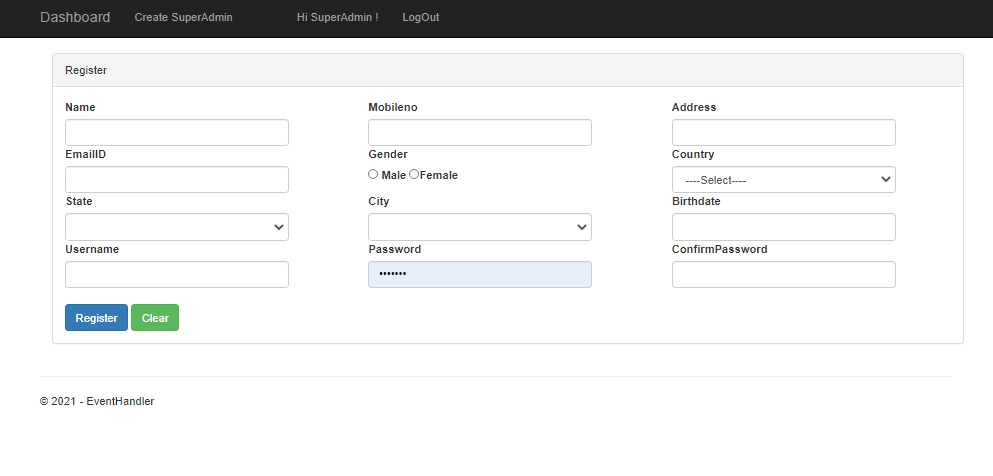
1. For adding new Flowers



1. Pending approval

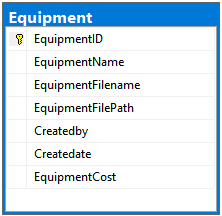


* Super Admin user interface

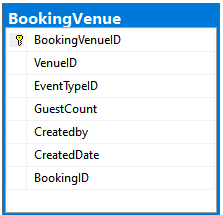


## Back- end Design

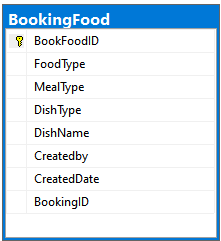
1. Equipment table



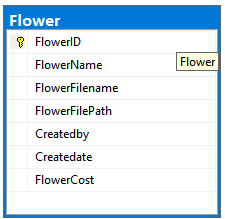
1. Booking Venue



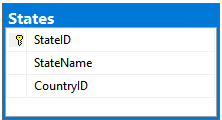
1. Booking Food



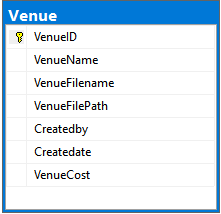
1. Flower



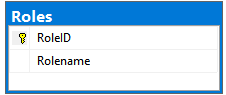
1. States



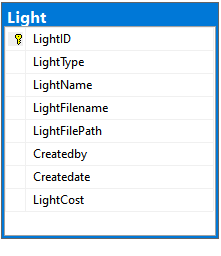
1. Venue



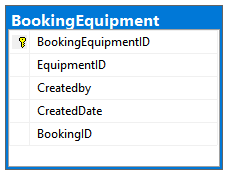
1. Roles



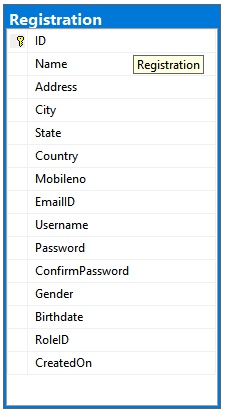
1. Light



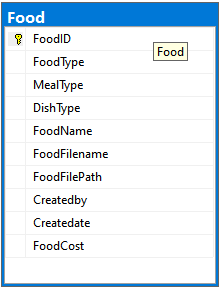
1. Booking Equipment



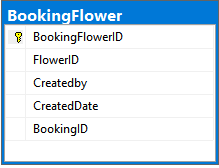
1. Registration



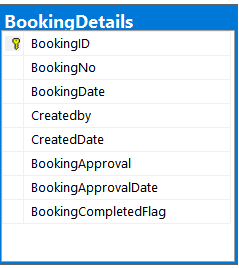
1. Food



1. Booking Flower



1. Booking Details



1. Booking Light

