**Invoice Application**

This application has been built using MVC architecture with basic features.

**App Features:**

It contains two pages, Home and Create Invoice.

Home page will show all the invoice details with status whether it's paid or unpaid. If one would like to mark any of invoice as paid then need to select the invoice record and click on "Paid Invoice" button.

Create Invoice page will show To, From and items details. To & From details would be Name and Address. For Item details, need to click on "Add Item" button to add the items with details like description, quantity, rate per item, cost for quantity. Based on entered details, total cost would be automatically calculated. To mark the invoice as paid, need to check the paid checkbox just before Total Cost tag.

DataBase has been setup in Azure.

**Technologies Used:**

Asp.Net MVC 4, C#, AngularJS 1.7, HTML, Bootstrap, Sql Server 2017 (Azure)

UI Layer: BootStrap & Angular JS

Business Layer: Asp.Net Model, C#

DB Layer: Sql Server 2017 hosted on Azure

**Azure DataBase Setup & Details:**

DataBase has been setup in Azure. It can be connected from your local Sql Server Management Studio Client. Details are given as below:

Server Name: invoiceserver.database.windows.net

DataBase Name: InvoiceDB

User Name: sangeeta

Password: Invoice@1234

**Table Details:**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Table Name** | **Remarks** |
| 1. | InvoiceDetails | It's main table and contains information like InvoiceId (auto generated), Customer Name, Customer address, Company Name, Company Address, Total cost, Paid/Unpaid, Status |
| 2. | InvoiceItem | It contains information like item number (auto generated), description, quantity, rate per unit, cost and reference to main table i.e. InvoiceDetails. |

**Stored Procedures:**

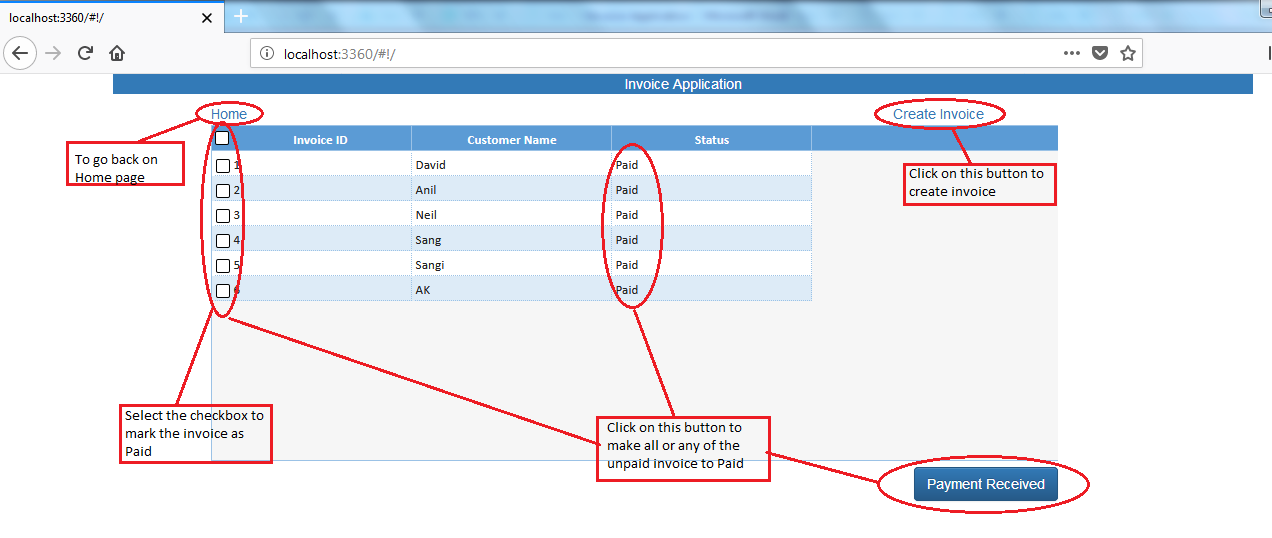
|  |  |  |
| --- | --- | --- |
| **S.No.** | **Stored Procedure Name** | **Remarks** |
| 1. | InvoiceAndItemDetails | This stored procedure is being used to insert invoice details and parts of item details. |
| 2. | InsertItemDetails | This stored procedure is being used to insert item details. |
| 3. | UpdateInvoice | This stored procedure is being used to update invoice status as paid which is selected from UI. |
| 4. | GetInvoiceDetails | This stored procedure is being used to get invoice details. |
| 5. | Split (It's a Function) | This function is being used to split the array of invoice ids to update the status for selected invoices. |

**Code Details:**

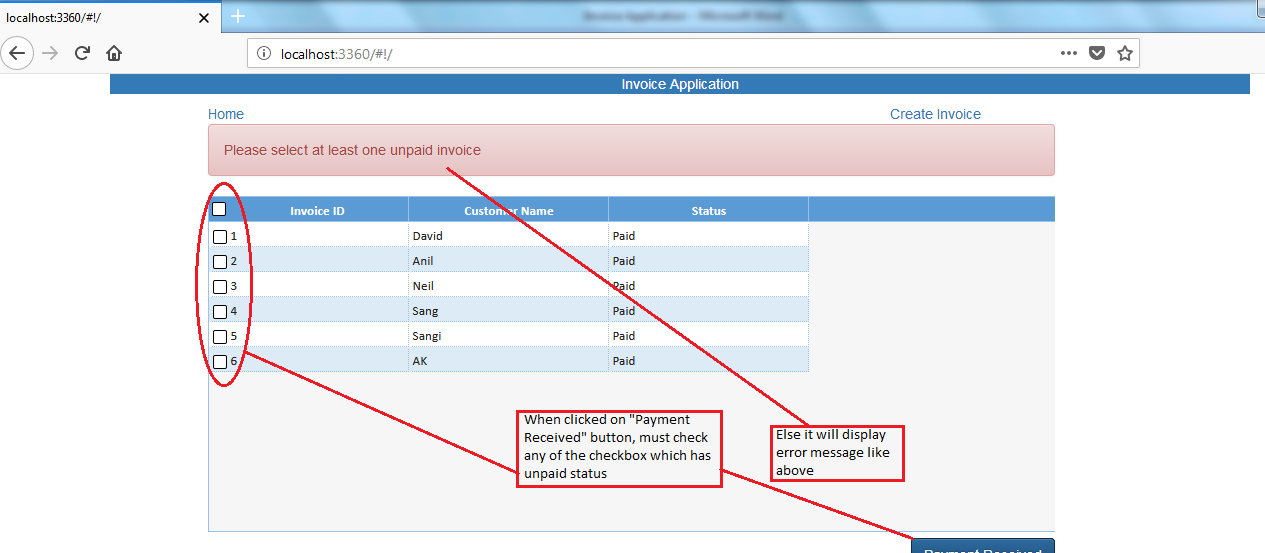
|  |  |  |
| --- | --- | --- |
| **S.No.** | **Folder Name** | **Remarks** |
| 1. | Views | All HTML file(s) to display UI |
| 2. | PartialHTML | All Helper/ partial HTML files |
| 3. | Scripts | All Script files |
| 4. | Models | All Business Logic files |
| 5. | Controllers | All the files which communicate with Model and View |
| 6. | Content | All CSS files |
| 7. | App\_Start | Default created and contains all config mapping files like Route, filter, WebAPI |

**Application Workflow:**

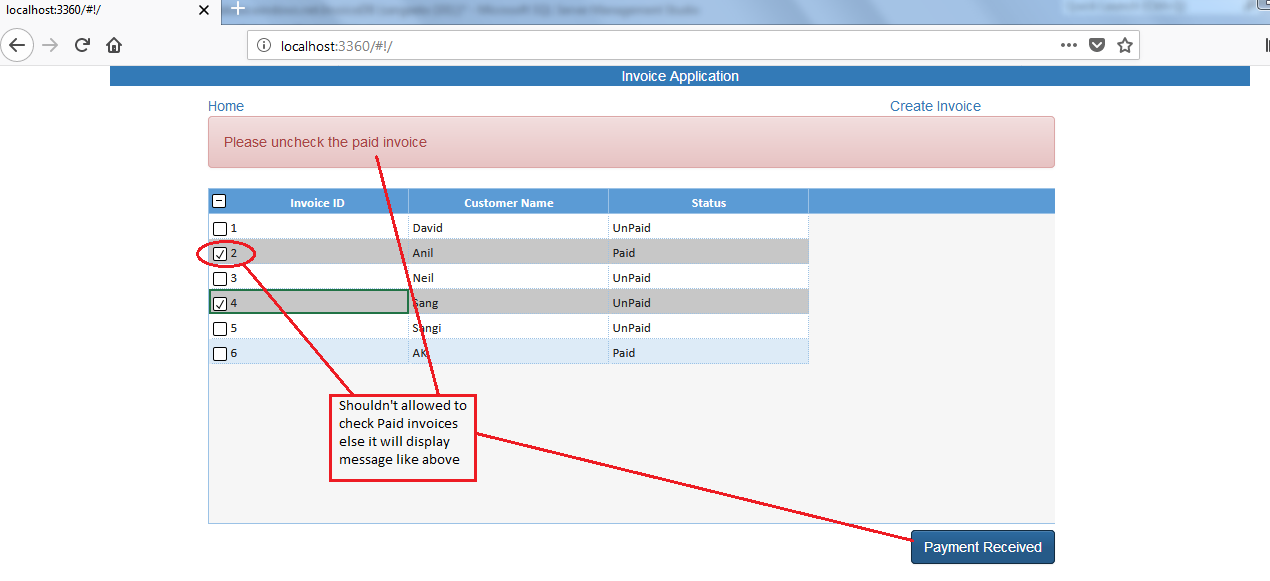
Home page would display all the created invoice (all invoices that available in DataBase).

1. Home Page:

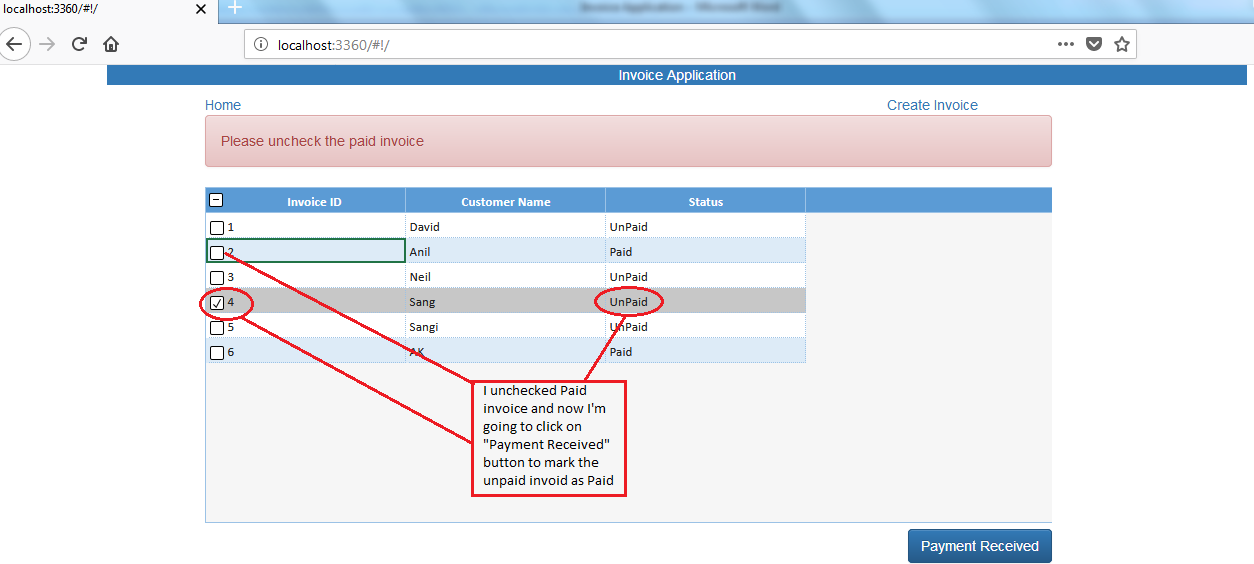
2. If Clicked on "Payment Received" button without selecting the checkbox then it will display error like below.



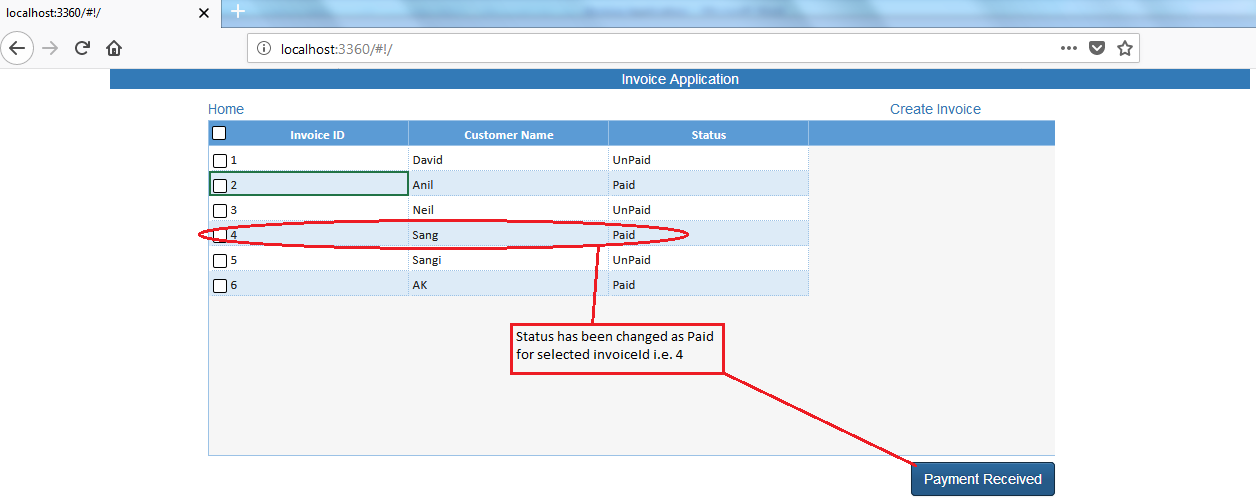
3. If selected the "Paid" invoices then it will display error message like below. Only Unpaid status invoices allowed to mark the status as Paid.



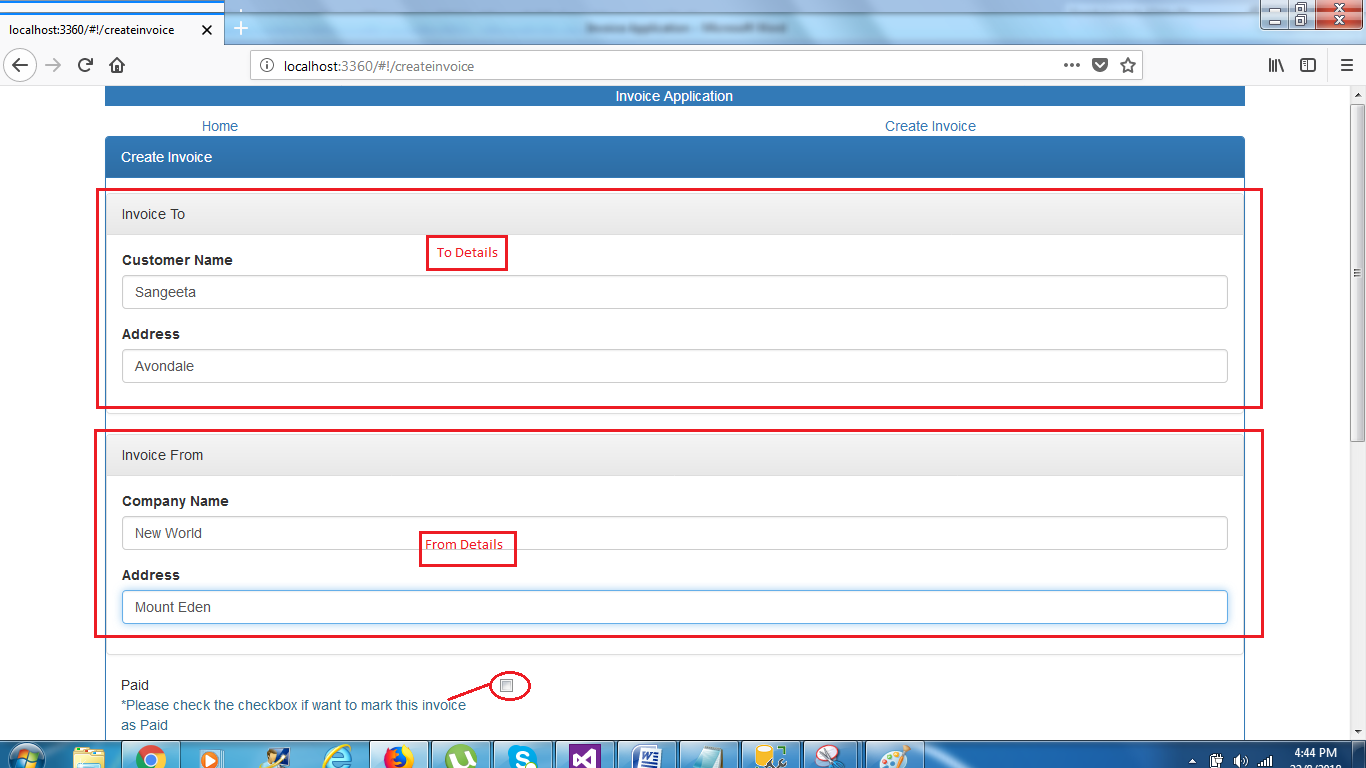
4. Unpaid status invoices only allowed to mark it as Paid.



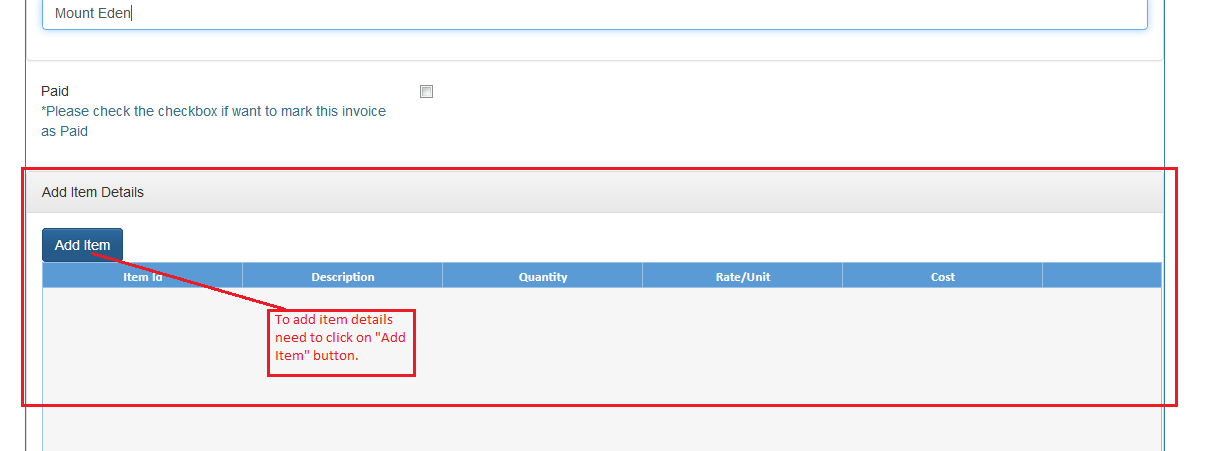
5. Status will be updated as Paid for unpaid invoices.



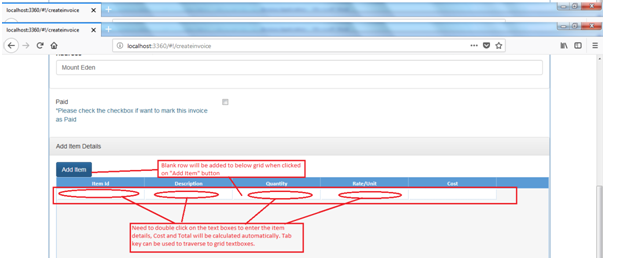
6. Create Invoice Page:



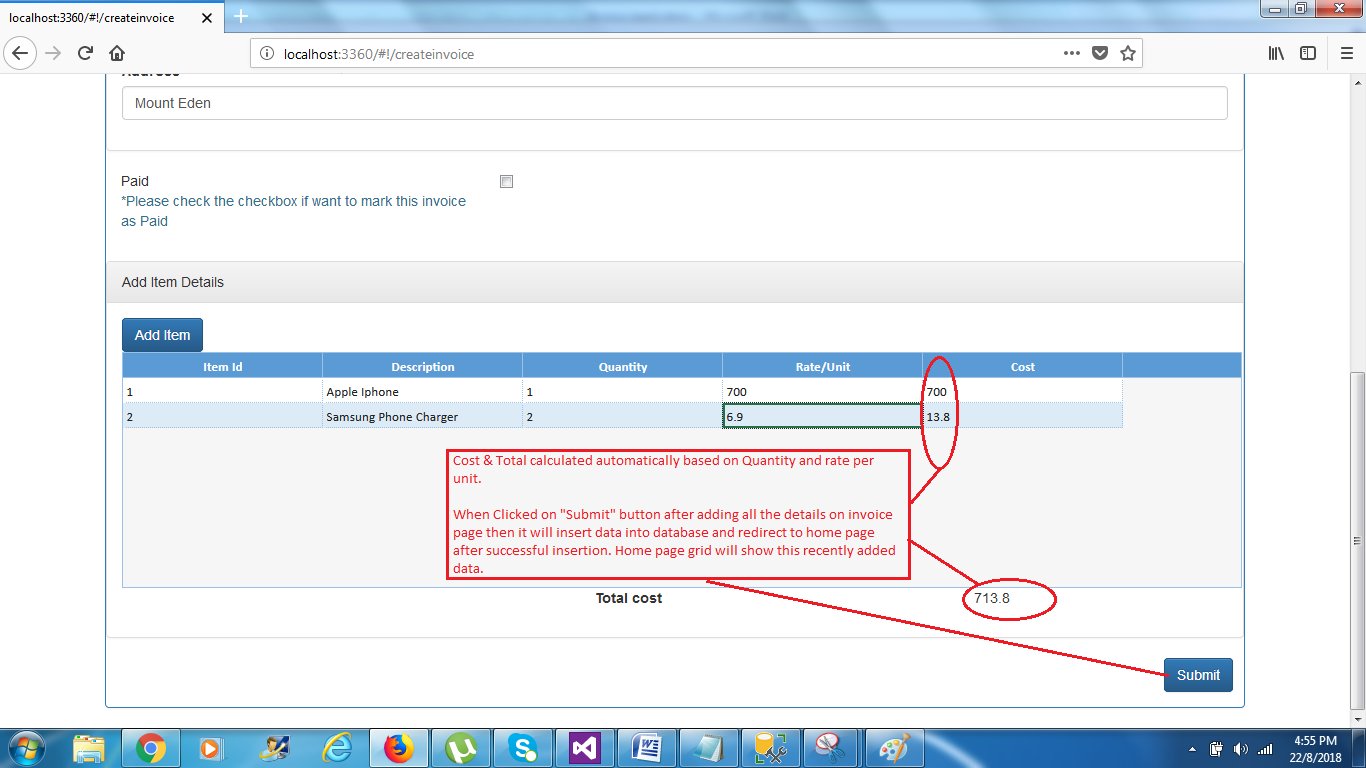
7. Add Item details on Create Invoice page. Need to click on "Add Item" button to add item details.



8. A blank row will be added into grid to add item details. Need to double click on grid textboxes to enter the item details. Tab can be used to traverse to textboxes. Cost and Total will be calculated automatically. **Assumption:** Item Id to be integer (although Validation is not applied for integer).



9. Cost & Total calculated automatically based on Quantity and Rate per unit. After adding all the details can click "submit" button to insert (create invoice) this data into data base. After successfully insertion of invoice record in data base then it will redirect to home page and home page will display this recently added data into grid.



10. Data has been inserted successfully and redirected to Home page.

**Assumptions:**

1. Item Id on Create Invoice page should be integer (validation yet to be applied).

2. Quantity, Rate per Unit and Cost must be integer or float (validation yet to be applied).

3. DataBase details are kept in web.config file which is not a good practice for security reason. It should either be encrypted or should use IIS integrated security (we can use the IIS service account to access Data base then no need to define user name and password in web.config file ).

I'm not able to encrypt the web.config connection string as I don't have "**Aspnet\_regiis.exe**" on my laptop.

This can be achieved by using below:

aspnet\_regiis -pe "connectionStrings" -app "/SampleApplication"

**PLEASE NOTE:** Insertion/Updation/fetching records to/from database can be achieved by using entity framework as well. I have added Entity Framework model into the project but it's not fully implemented (as require bit more time). So I have achieved it by using Sql server objects into business layer. So currently two connection strings are there in web.config file (please don't get confused). One is for entity model object and other is for sql object.