

Asset Management System -User Manual

Asset Management System (AMS) is a multi-functional complete web application developed using ASP.NET Core and EF Core code first technology. It enables you to manage all kinds of assets for your company through proper monitoring and a dynamic reporting system.

You can manage assets available in your office, like computers, Printers, Portable Hard-disk, Pen-drive, etc. You can track the history of assets as well.

Core Feature:

Software Inventory Management: Manage all the software of your organizations with the Asset Management System. It provides you with a dashboard to input all the software purchases you have made. Update the license dates from when it was installed, to when the software will expire.

Hardware Inventory Management: Supervise the status of your hardware and the employees' using it with the help of the Asset Management System. Record all the data about the hardware like a purchaser, date of purchase, number of equipment bought, list of all the equipment acquired, et cetera. You can also see the employees who are assigned to use the hardware.

Asset Life Cycle Management: Track an asset life cycle in the organization with Asset Management System. Check for the date it was bought for and see how well the asset was utilized.

Organizations are also enabled to see how many employees have used the asset for what amount of time. It just makes your asset management seamless.

Asset History: It is a very simple feature that allows you to track any change in assets. Ex new assignment, asset status change, etc.

Supplier Management: Asset Management System provides a module to manage all your Suppliers in one place. It also helps you with managing all suppliers.

User Management: Users are provided with admin accounts while registering their company. Admins can create IT Manager Accounts and others custom account by limited page access.

Major Features

- ✓ Software Inventory Management
- ✓ Hardware Inventory Management
- ✓ Asset Life Cycle Management
- ✓ Asset History
- ✓ Asset Allocation
- ✓ Employee Management
- ✓ User Management
- ✓ Dynamic Reporting System
- ✓ Dynamic Dashboard
- ✓ And many more

Technologies:

Microsoft Visual Studio Community 2019 or later, ASP.NET Core 3.1, Identity Core, Entity Framework Code First, MSSQL SERVER 2017, AdminLTE 3.0.5, JavaScript, jQuery data table, bootstrap 4, sweetalert, toastr, Fontawesome etc.

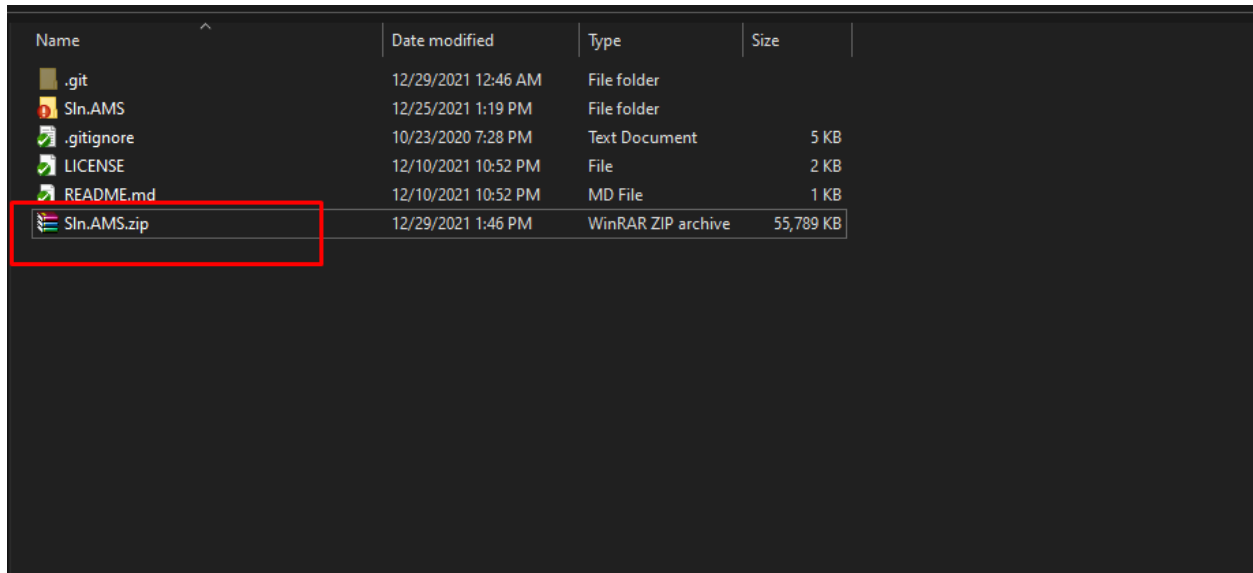
Prerequisite Application:

Microsoft Visual Studio Community 2019 or later, .Net Core 3.1, MS SQL SERVER 2017, IIS, Chrome/Edge Browser.

How to run this Project:

Step 1:

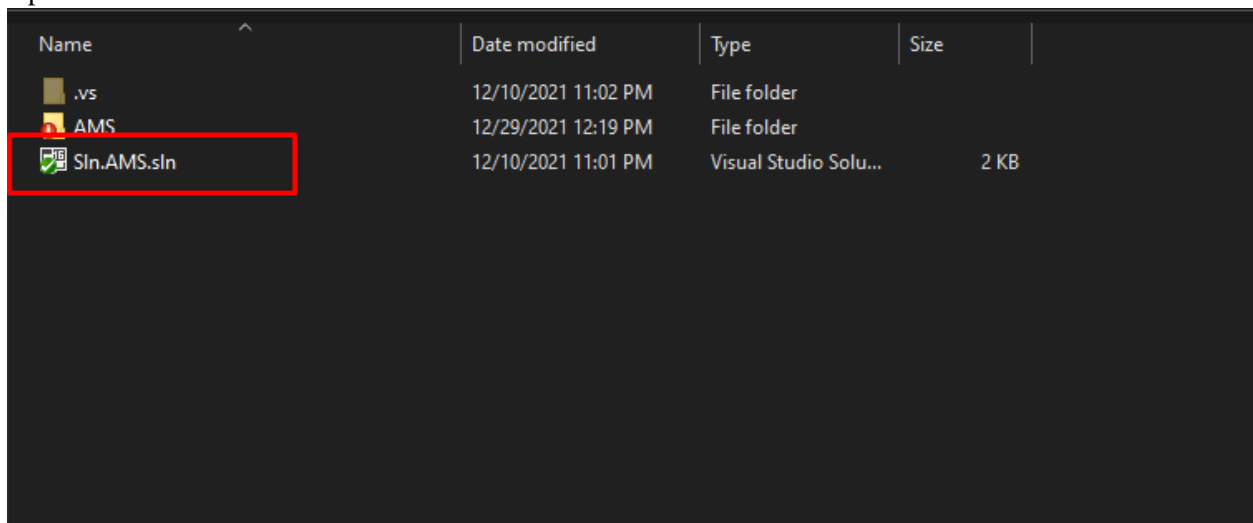
Unzip the downloaded zip folder (Sln.AMS.zip)



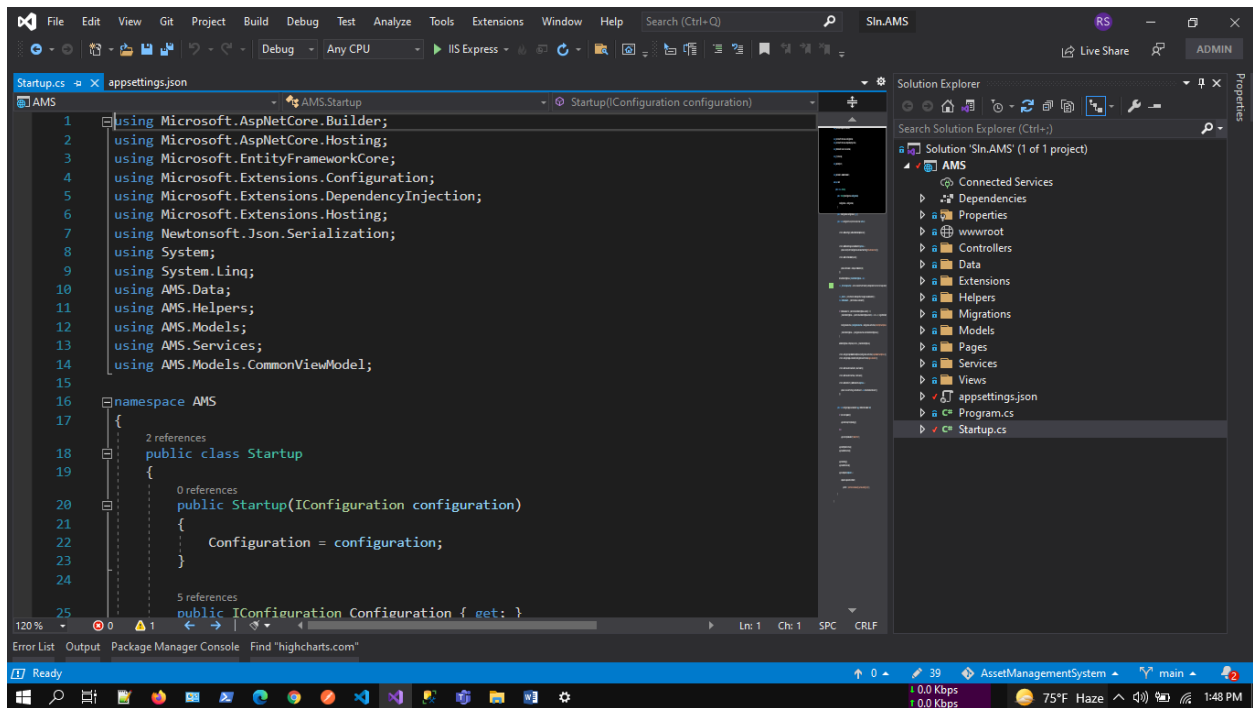
Name	Date modified	Type	Size
.git	12/29/2021 12:46 AM	File folder	
Sln.AMS	12/25/2021 1:19 PM	File folder	
.gitignore	10/23/2020 7:28 PM	Text Document	5 KB
LICENSE	12/10/2021 10:52 PM	File	2 KB
README.md	12/10/2021 10:52 PM	MD File	1 KB
Sln.AMS.zip	12/29/2021 1:46 PM	WinRAR ZIP archive	55,789 KB

Step 2:

Open Sln.AMS.sln file with Visual Studio 2019



Name	Date modified	Type	Size
.vs	12/10/2021 11:02 PM	File folder	
AMS	12/29/2021 12:19 PM	File folder	
Sln.AMS.sln	12/10/2021 11:01 PM	Visual Studio Solu...	2 KB



Step 3:

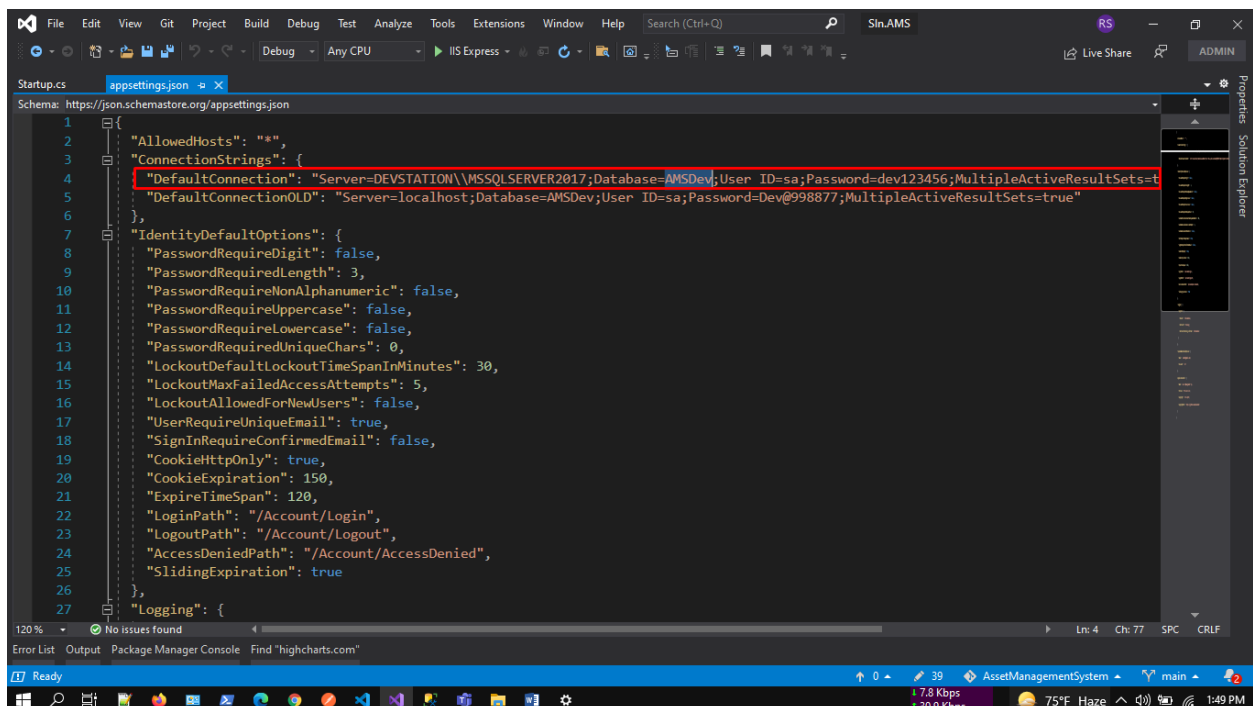
Update database connection: appsettings.json

“DefaultConnection”: “Server=DEVSTATION\\MSSQLSERVER2017;Database=AMSDev;User ID=sa;Password=dev123456”

Server=Your MSSQL Server Name

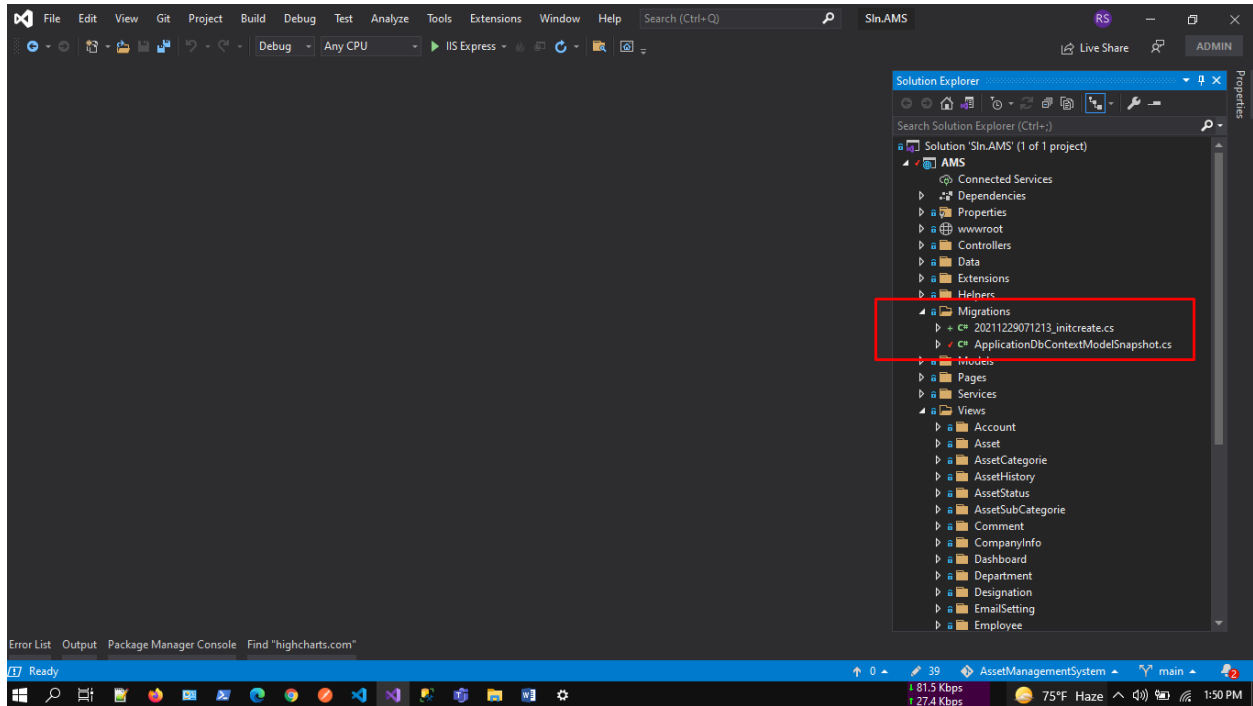
User ID= Your MSSQL Server User Name

Password= Your MSSQL Server User Password



Step 4:

No need to create new migration rule and update database. You just need to build and run the application. Already initial migration rule included with project. Please make sure MS SQL server running properly.



Step 5:

5.1: Build application by pressing f6

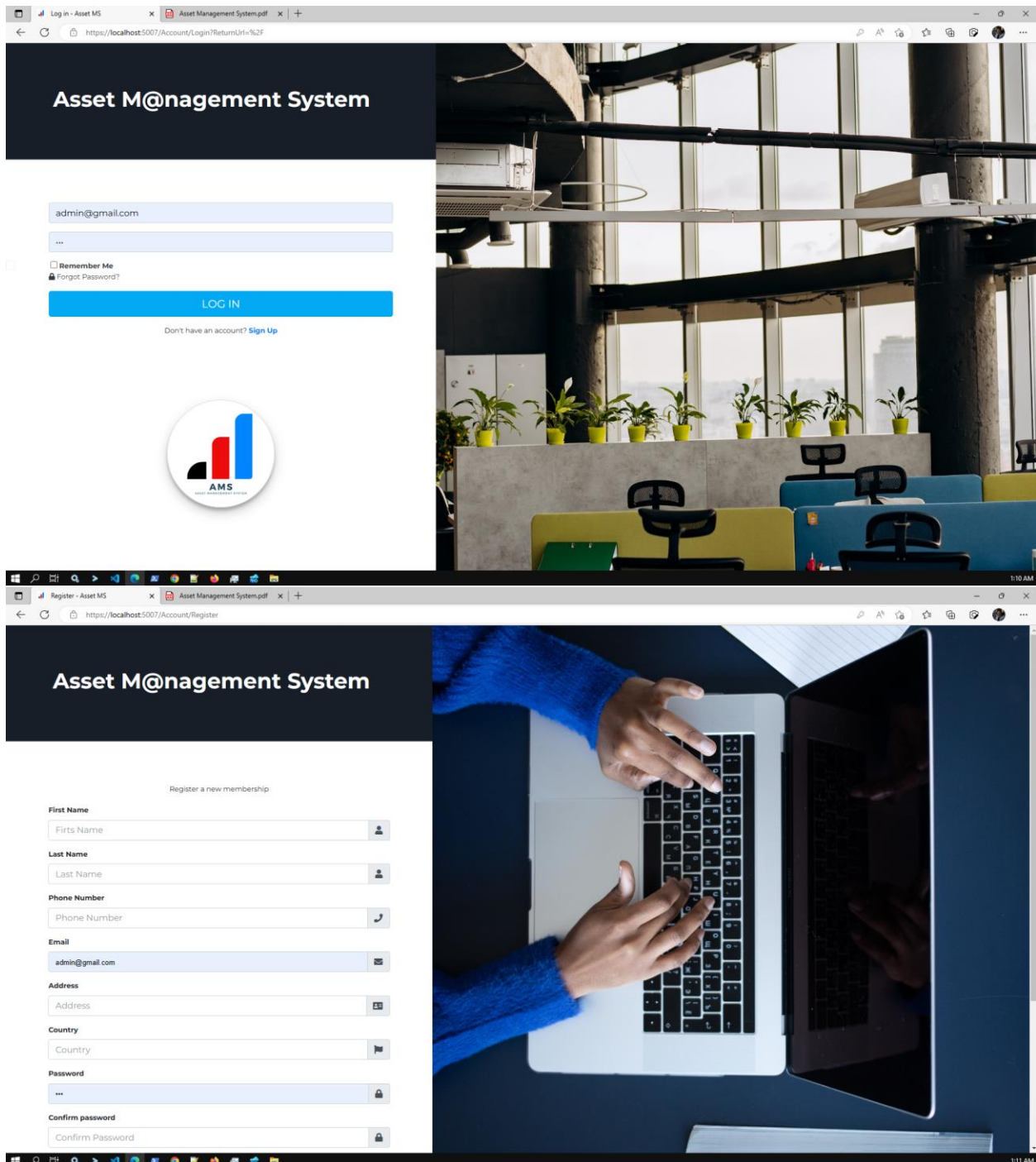
5.2: Run application by pressing f5

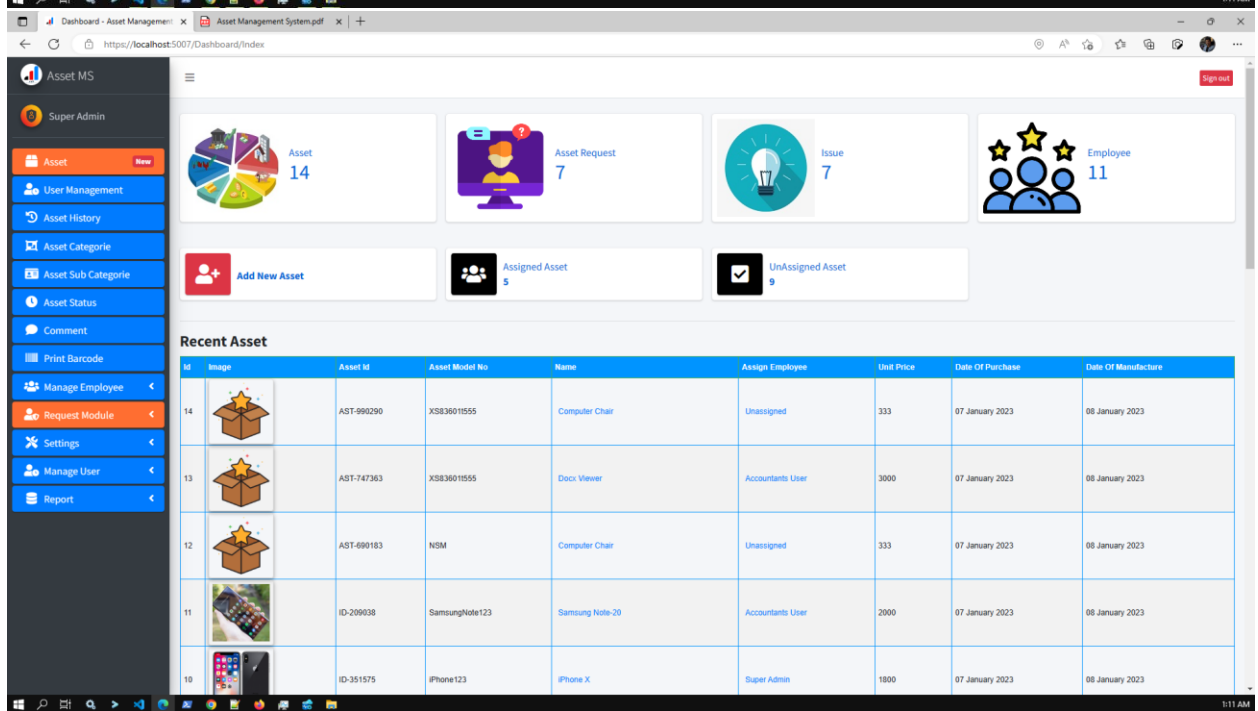
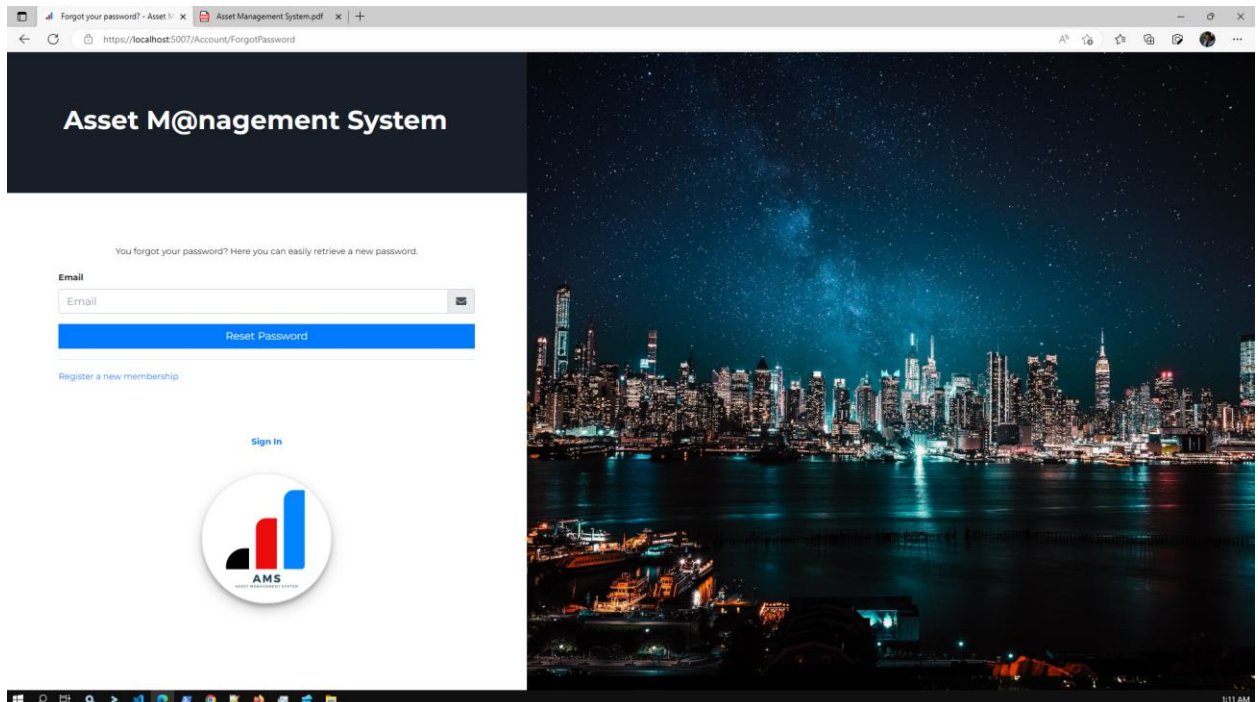
<https://localhost:5001/>

Initially in the SQL server, relevant tables and data will be created in the database dynamically as code first approach.

Step 6:

Browse full application.





Asset MS

Super Admin

Asset

User Management

Asset History

Asset Category

Asset Sub Category

Asset Status

Comment

Print Barcode

Manage Employee

Request Module

Settings

Manage User

Report

Add New

Search:

ID	Image	First Name	Last Name	Phone Number	Email	Created Date	Allocate	Action
11		Accountants	User	7882346831	accountants@gmail.com	1/7/2023	+Allocate	
10		HR	User	74478806002	hr@gmail.com	1/7/2023	+Allocate	
9		Finance	User	78925385244	finance@gmail.com	1/7/2023	+Allocate	
8		Technology	User	66382886365	tech@gmail.com	1/7/2023	+Allocate	
7		Regular	User	96559677851	regular@gmail.com	1/7/2023	+Allocate	
6		Employee 1	User	89917578227	Employee1@gmail.com	1/7/2023	+Allocate	
5		Employee 2	User	68721458185	Employee2@gmail.com	1/7/2023	+Allocate	
4		Employee 3	User	10519002444	Employee3@gmail.com	1/7/2023	+Allocate	
3		Employee 4	User	71245073191	Employee4@gmail.com	1/7/2023	+Allocate	
2		Employee 5	User	42130519628	Employee5@gmail.com	1/7/2023	+Allocate	
1		Super	Admin	+8801674411603	admin@gmail.com	1/7/2023	+Allocate	

Showing 1 to 11 of 11 entries

Previous 1 Next

Copyright © 2021-2022 Zero Byte. All rights reserved.

Version 1.0.1

Sample Asset Report:

Print

Total: 1 sheet of paper

Printer

Microsoft Print to PDF

Copies

1

Pages

All

Color

Color

More settings

Print using system dialog (Ctrl+Shift+P)

Print

Cancel

Asset Information

XYZ Company Limited

Dhaka, Bangladesh

Mobile: 132546789

Email:

Website: www.xyz.com

Print Date : 29 December 2021

ID	1
Asset Id	ID-101
Asset Model No	M1 Chip
Name	Macbook Pro m1
Description	
Category	IT
Sub Category	
Quantity	
Unit Price	2500
Supplier	
Location	
Department	
Sub Department	
Warranty in Month	
Depreciation in Month	
Image	Macbook Pro m1
Date Of Purchase	12/29/2021 1:16:13 PM
Date Of Manufacture	6/29/2021 1:16:13 PM
Year Of Valuation	1/1/0001 12:00:00 AM
Assign User Id	0

Sample Asset Allocation Report:

The screenshot shows a web application interface for 'Asset M@nagement Sys'. The browser address bar displays 'https://localhost:5001/Report/AssetAllocationReport'. The interface includes a sidebar with navigation options: 'Manage Asset', 'Manage Employee', 'Settings', 'Manage User', 'Report', 'Asset Status', and 'Asset Allocation'. The main content area is titled 'Asset Allocation Report' and features a 'Start Date' and 'End Date' filter with 'mm/dd/yyyy' placeholders, a 'Submit' button, and a 'Reset' button. Below the filter, it states 'Showing Data on Date Range: 12/29/2021 12:00:00 AM to 12/29/2021 12:00:00 AM'. A table displays the asset allocation data:

SL	Assigned User	Total Assigned
1	Ms Merry	1
2	Mr Tom	1
3	Unassigned	5
		Σ: 7

Below the table, a 'Summary' section provides a breakdown:

Total Asset:	7
Total Assigned Asset:	2
Total Unassigned Asset:	5

The footer of the application shows 'Copyright © 2021-2022 Zero Byte. All rights reserved.' and 'Version 1.0.1'. The Windows taskbar at the bottom indicates the system time as 1:52 PM on 12/29/2021.

Demo Video Preview

<https://youtu.be/4J2HOIe5638>

App Screenshot: <https://photos.app.goo.gl/wFYWnwm4WMPxcRG26>

DB Migration:

Tools--> Nuget Package Manager--> Package Manager Console

```
PM> add-migration initcreate
```

```
PM> update-database
```

Check .Net Version:

Press Windows + R.

Type cmd.

On the command prompt, type dotnet --version

FAQ:

Question: How to install Database/How to run migration/Database table not creating?

Answer: Please follow these steps,

1. Make sure the right database connection is in the appsettings.json file
2. Check your server name, by running this SQL: **SELECT @@SERVERNAME AS 'Server Name'**
3. Delete the database from your SQL Management Studio (if already created)
4. Just build the project (As we have already included Migrations in the project)
5. Run the project
6. Now check your DB. Hopefully, DB and all tables you will find now.

Please cross-check, Program.cs and appsettings.json files properly.

appsettings.json

```
"DefaultConnectionMSSQL":
```

```
"Server=DEVSTATION\\MSSQLSERVER2017;Database=WebStarterKit;User  
ID=sa;Password=dev123456;MultipleActiveResultSets=true",
```

Note that,

Server = your local server

Database = any name

User ID=your local MSSQL Server user name

Password=your local MSSQL Server user password

Required Application:

- Visual Studio 2022
- MSSQ 2017/later
- Dotnet SDK 6.0

Our Product:

SL	Product
1	Advance POS System with Stock Manager ASP.NET Core EF Core .NET Core 6.0
2	Asset Management System with Barcode ASP.NET Core EF Core .NET Core 6.0
3	Hospital and Pharmacy Management System ASP.NET Core EF Core
4	ASP.NET Core full CRUD with .NET 5 MSSQL/MySQL EF code first
5	Inventory and User Management System
6	Invoice Gen(Invoice Generation and Management) using .Net Core and EF
7	Complaint Issue Helpdesk Ticket My Ticket HelpDesk Support System ASP.NET Core EF Core
8	Advanced Files & Users Management ASP.NET Core EF Core .NET Core MSSQL MySQL
9	User Management
10	Business ERP Solution/Product/POS/Company Management
11	School/College/University Management ERP
12	ASP.NET Core CRUD Operation Using Dot Net 6.0 MVC MSSQL MySQL EF Core Code First jQuery

Tutorials Video:

[How to deploy/host ASP.NET Core 6.0 MVC App to IIS Using VS Code | Visual Studio](#)

[How to Install Microsoft SQL Server 2019 and SSMS on Windows 10 | Windows 11| Step By Step](#)

[Our YouTube Channel](#)

Supports:

If you want to modify or add new features, or any technical issues you have faced during installation you can always contact us at, shahedbddev@gmail.com

WhatsApp: 8801674411603

Telegram: shahed71bd

Skype: shahedatomap

Email: shahedbddev@gmail.com