



INVENTORY MANAGEMENT SYSTEM

(Advanced Programming Techniques)

Professor

Dr. Andreas Fischer

Team Members

Mehtab Ahmed 00772927

Muhammad Ali 732065

Umang Rajpara 00097170

Contents



	1
INVENTORY MANAGEMENT SYSTEM	1
Team Members	1
Mehtab Ahmed 00772927	1
Muhammad Ali 732065	1
Umang Rajpara 00097170	1
Objective	3
Inventory Management System	3
Implementation Details	4
Why C# and Asp.Net Core (.Net Core)	4
Database	4
Better performance	5
Reduced application cost and complexity	5
Portability	5
Reliability	6
Accessibility	6
Database Design	6
	7
Object Relational Mapper (ORM) and EntityFamrwork	8
How to run the Project	9
Running the Project on Linux	9
Register Microsoft key and feed	9
Install the .NET Core 2.1 Runtime	10
Install the .NET SDK(This is not required in the Production Environment)	10
Test the installation	10
Downloading the Source Code	11
Downloading directly from the repository	11
Downloading with Git	12

Running the Project -Linux	14
Running the Project -Windows	15
How to use the application	17
How to Change Username and Password	17
User Interface of the Application for Different Users	19
Admin UI	19
Admin Tasks	20
Manage Users	21
Manage Branches	22
Manage Tags	23
Employee UI	24
External Account UI	24
Anonymous User	25
Core Tasks	26
User Interface	27
Team Members and Responsibilities	30

Inventory Management System

An inventory management system has several critical components. Inventory systems allow a company to keep track of all its movable assets. The system has an overall summary pages of the items in the inventory like Total Count, Images, Prices, Category and various tags associated with the items. It has proper authentication and authorization for different kind of users of the system.

Implementation Details

The project is developed according to Agile and Scrum principles. Small sprints were developed each week and deployed to GitLab(an open source web-based Devops). ASP Dot Net Core is used as the language for programming in this project. **ASP.NET Core** is a free and open-source web framework developed by Microsoft and the community. It is a modular framework that runs on both the full .NET Framework, on Windows, and the cross-platform .NET Core. SQLite will be Database and MVC(Model View Controller) will the software pattern used in this project.

Some of the technical details are given below

Core of the project: C# .Net Core

Target OS: Windows, Linux and Mac

Technologies for front-end: HTML, CSS, Bootstrap, jQuery and JavaScript.

Database: SQLite Database for Production

MS SQL Server for Development

Technologies for Back-end: ASP.Net CORE

Why C# and Asp.Net Core (.Net Core)

.NET Core is a free and open-source, managed computer software framework for Windows, Linux, and macOS operating systems. It is a cross-platform successor to .NET Framework. The project is primarily developed by Microsoft and released under the MIT License.

Moreover, Its easy learn , High productivity, well-structured, maintained by Microsoft and go up to the level of C++ in performance.

Database

During the Development phase of the application, Microsoft SQL Server was used. MS SQL Server has better GUI tools for dealing with the Database Engine. This improved the productivity. On other side , MS SQL Server is not cross platform (may be cross platform in the near future) and secondly Server is required to fetch and store on Database from application.

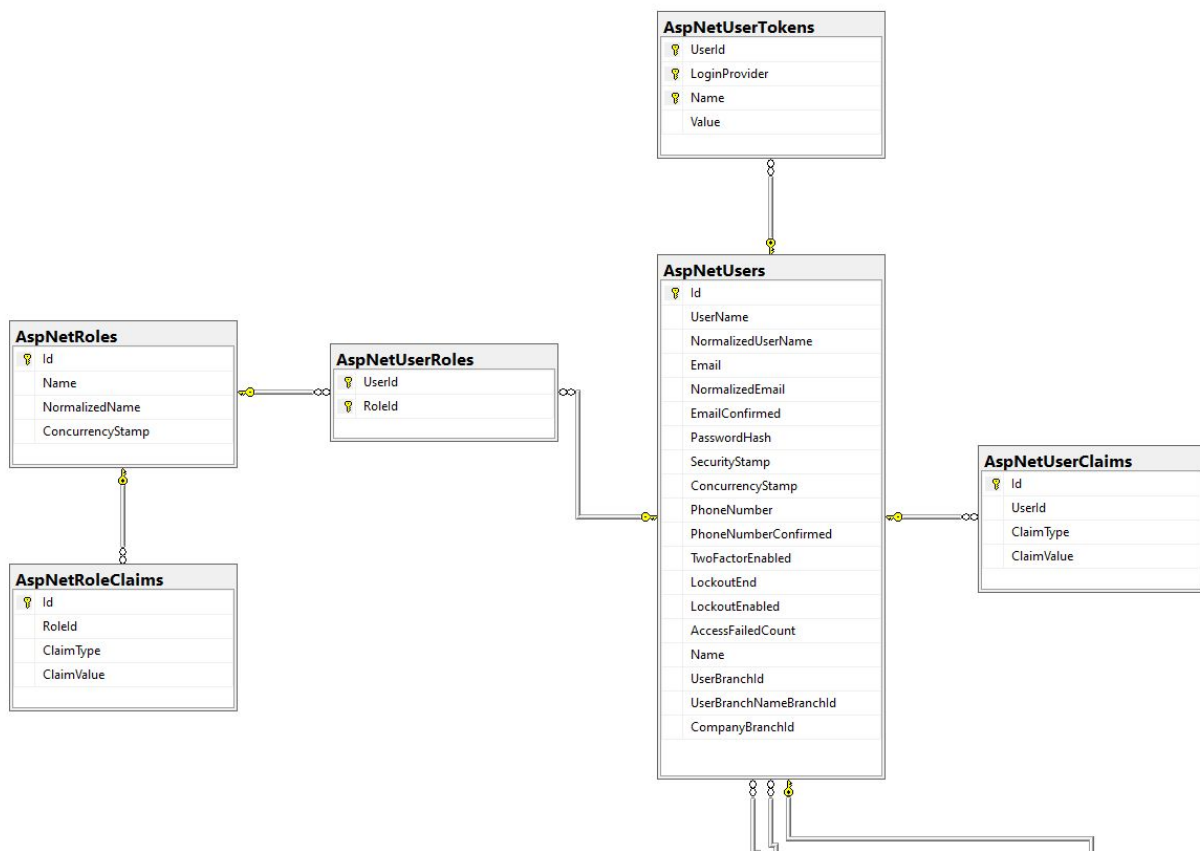
Due to these reasons, we used SQLite for the production environment as it is cross platform as well as Serverless.

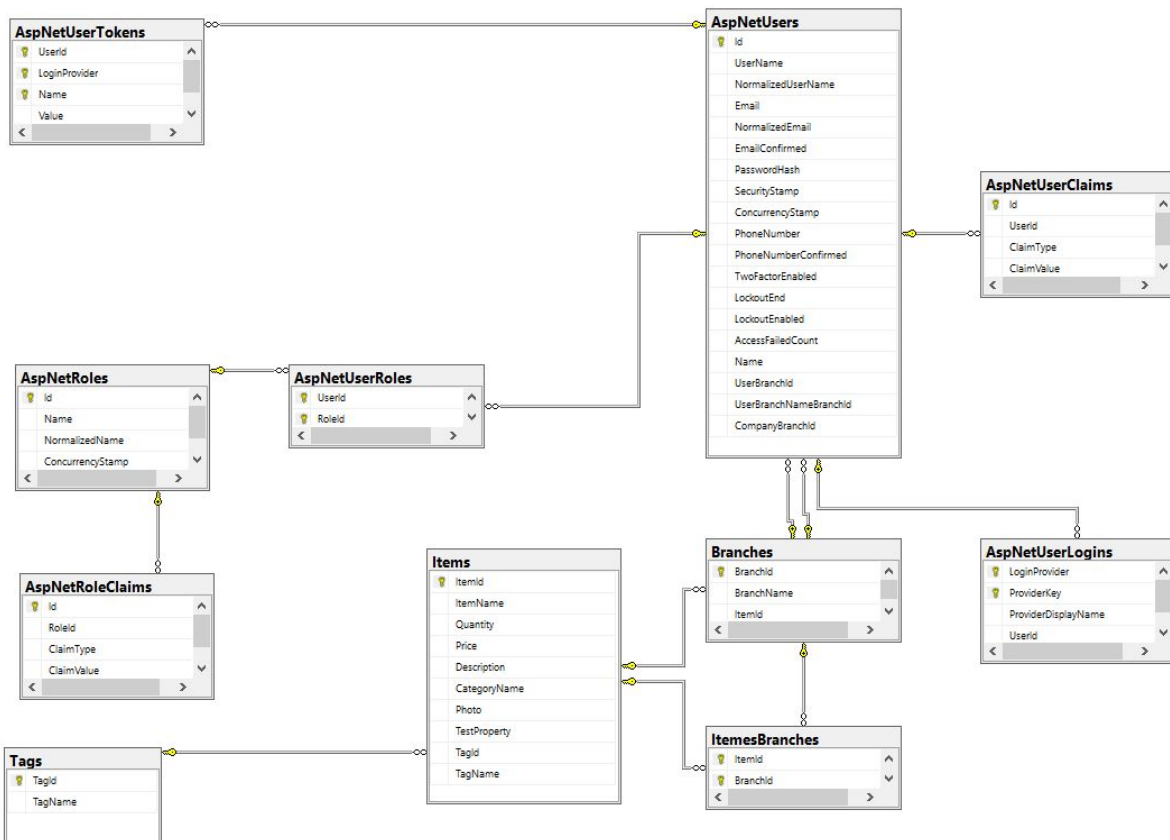
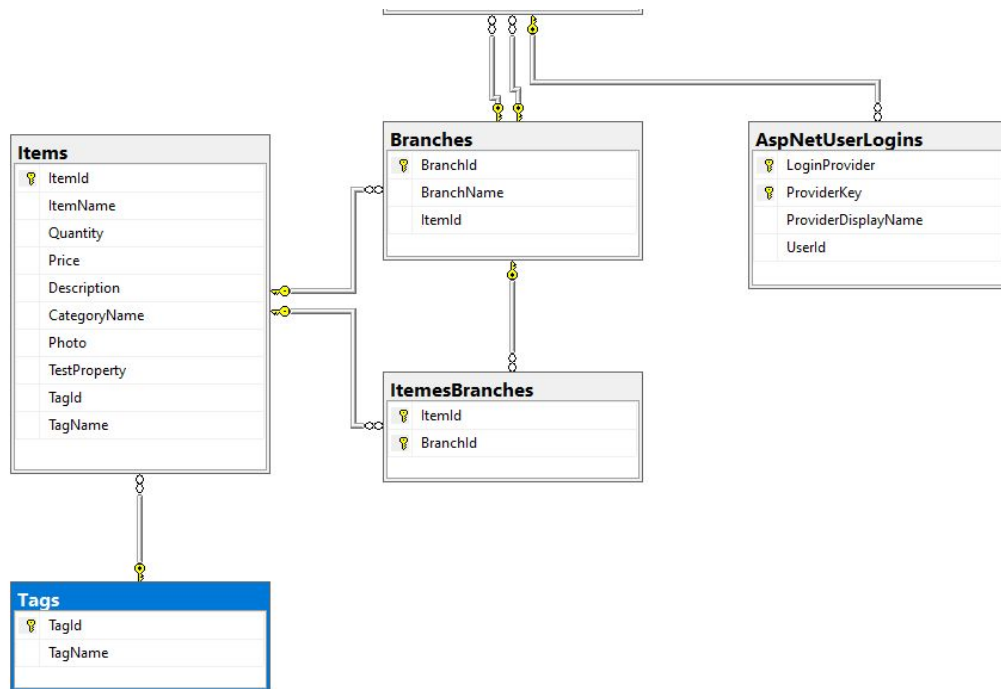
It is a file based Database and always reside with the application therefore deployment is easy .

Some of the other benefits of SQLITE are given below that are taken from the official site of the SQLITE(https://www.sqlite.org/aff_short.html)

Database Design

The Tables in the Database are self-explanatory as shown by the UML diagram below





Object Relational Mapper (ORM) and EntityFramework

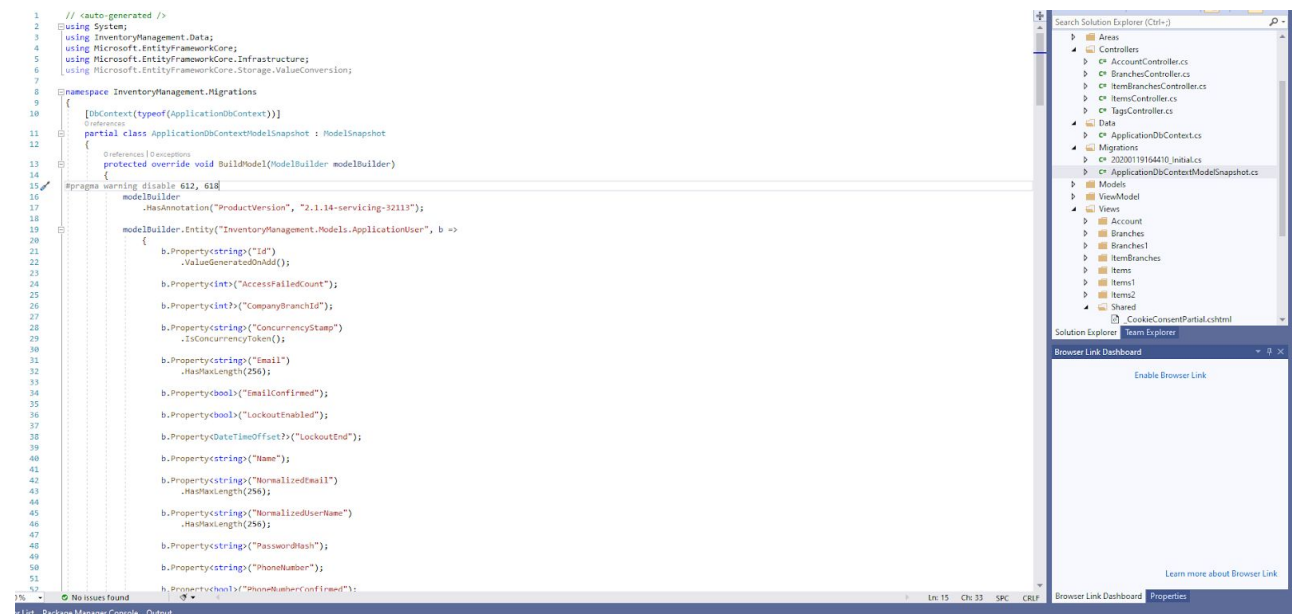
ORMs are the tools that converts Domain Model Classes into Relations in the Database. In very simple words, tables are generated in the Database against Model Class in the application by the ORM.

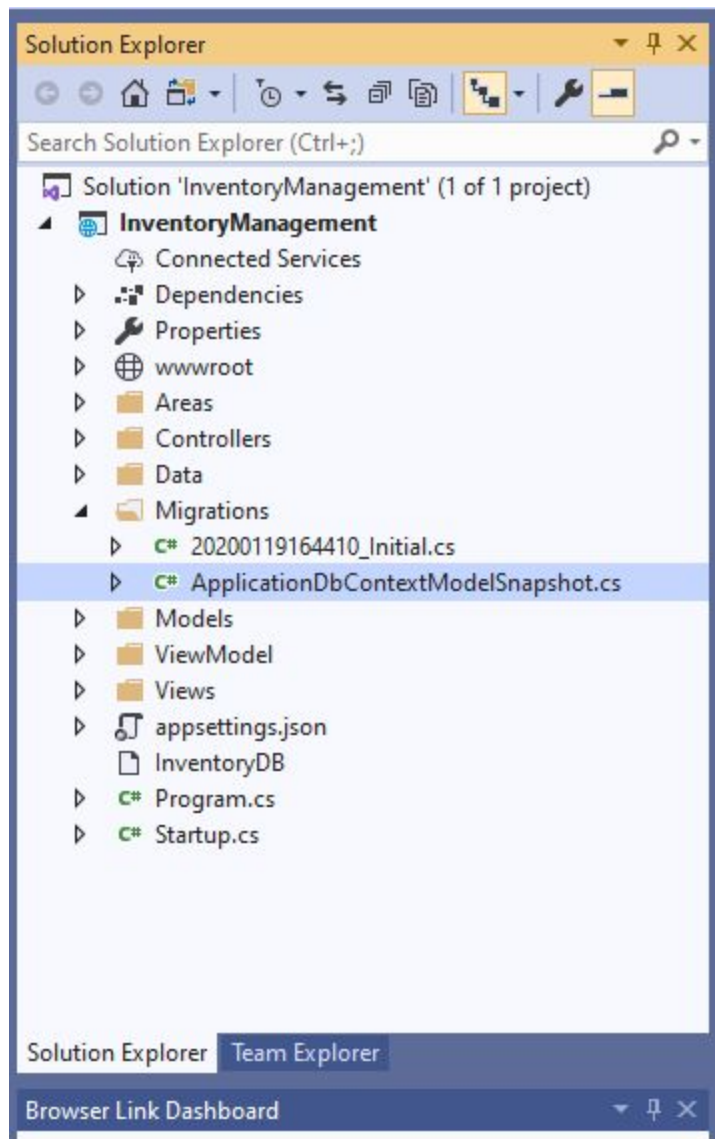
In General, an ORM library is a completely ordinary library that encapsulates the code needed to manipulate the data, so SQL queries are not required anymore(To be send from Application to the database) so a direct interaction is made with object of the language.

.Net Core uses has two famous ORM i-e, NHibernate or Entity Framework.

In our project we used the Entity Framework as it is the most recommended ORM by the Microsoft for the .Net based applications.

Below are some of the code snippets of the code generated by Entity Framework.



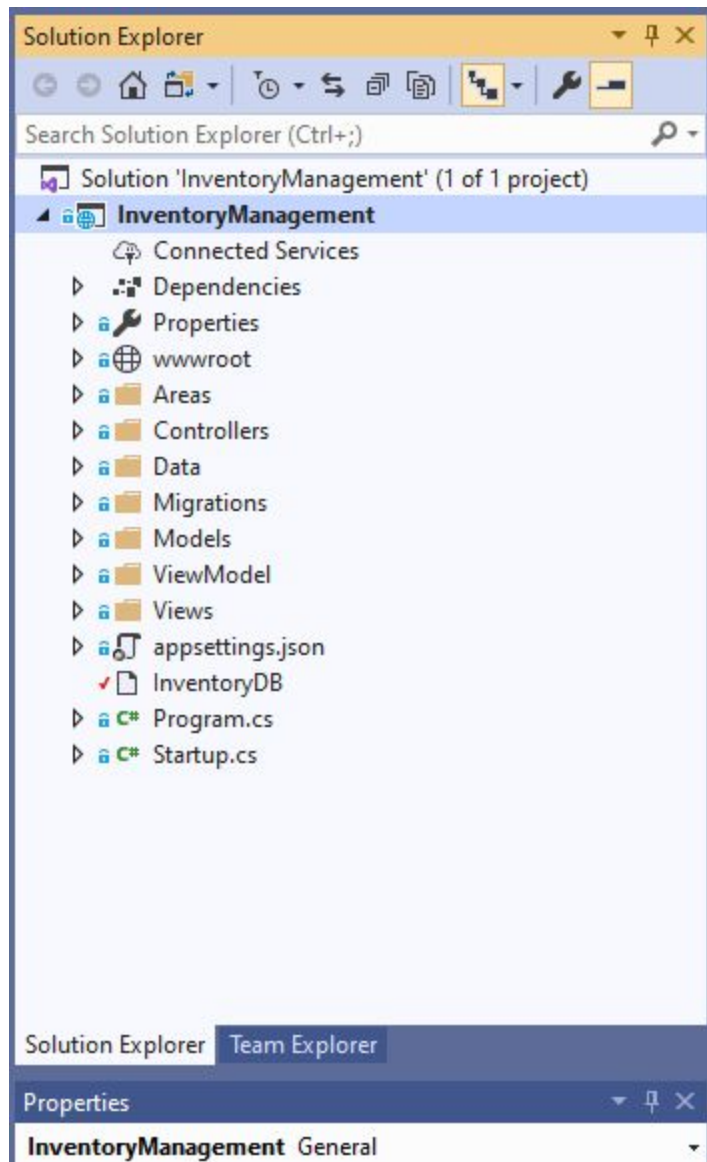


This code is added by Entity Framework(EF) automatically after applying the EF migrations as shown by the above figure. It lies in the folder called Migrations (This folder is also auto generated by EF).

Architecture of the Application

Our Application is based on **Model-View-Controller (MVC)** framework

The **Model-View-Controller (MVC)** framework is an architectural pattern that separates an application into three main logical components Model, View, and Controller. Each architecture component is built to handle specific development aspect of an application. MVC separates the business logic and presentation layer from each other. It was traditionally used for desktop graphical user interfaces (GUIs). Nowadays, MVC architecture has become popular for designing web applications as well as mobile apps.



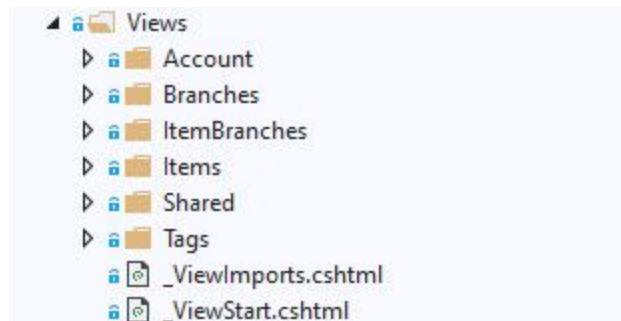
View

A View is that part of the application that represents the presentation of data.

Views are created by the data collected from the model data. A view requests the model to give information so that it resents the output presentation to the user.

The view also represents the data from chats, diagrams, and table.

Our Application consist of many views as per the requirements. Let's look at some of them.



In Asp.Net Core application a view can reside any where in the application. But Its always a good practice to add them in View Folder and make sub folder for each Controller view.

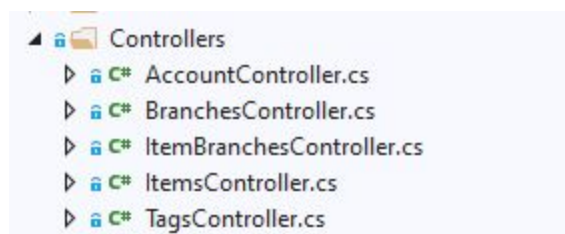
Here for example, the Account sub folder correspond the view required for Account Controller and so on.

Controller

The Controller is that part of the application that handles the user interaction. The controller interprets the mouse and keyboard inputs from the user, informing model and the view to change as appropriate.

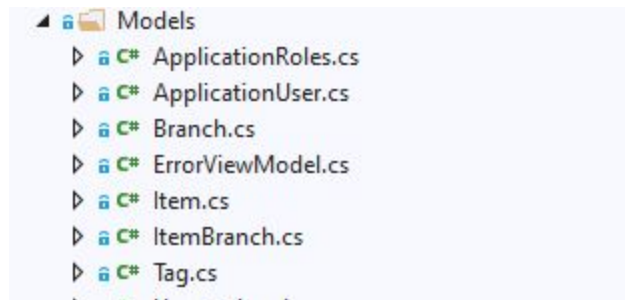
A Controller send's commands to the model to update its state(E.g., Saving a specific document). The controller also sends commands to its associated view to change the view's presentation.

In Asp.Net Core application a Controller will always reside in the Controller Folder of the application and it will always inherit from Base Controller Class.



Model

The model component stores data and its related logic. It represents data that is being transferred between controller components or any other related business logic.



How to run the Project

Here we will discuss running the project in the Linux and Windows OS. To run this project in any of the OS, .Net Core Runtime 2.1 or Above is required. Lets discuss these in detail.

Running the Project on Linux

This documentation is for Debian based Linux like Debian, Ubuntu, Kali Linux etc.

Register Microsoft key and feed

Before installing .NET, you'll need to register the Microsoft key, register the product repository, and install required dependencies. This only needs to be done once per machine.

Open a command prompt and run the following commands:

```
wget -qO- https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor >
microsoft.asc.gpg
sudo mv microsoft.asc.gpg /etc/apt/trusted.gpg.d/
wget -q https://packages.microsoft.com/config/ubuntu/18.04/prod.list
sudo mv prod.list /etc/apt/sources.list.d/microsoft-prod.list
sudo chown root:root /etc/apt/trusted.gpg.d/microsoft.asc.gpg
sudo chown root:root /etc/apt/sources.list.d/microsoft-prod.list
```

Install the .NET Core 2.1 Runtime

This commands will install the .NET Core Hosting Bundle, which includes the .NET Core runtime and the ASP.NET Core runtime. To install just the .NET Core runtime, use the `dotnet-runtime-2.1` package.

In your command prompt, run the following commands:

```
sudo apt-get install apt-transport-https
sudo apt-get update
sudo apt-get install aspnetcore-runtime-2.1
```

Source: <https://www.microsoft.com/net/download/linux-package-manager/ubuntu18-04/runtime-2.1.0>

Install the .NET SDK(This is not required in the Production Environment)

This is an optional command. In your command prompt, run the following commands:

```
sudo apt-get install dotnet-sdk-2.1
```

Notice: It is not necessary to install the .NET SDK on a productive server. You only need to install the SDK in the development environment.

Test the installation

To make sure it works after you've set it up, just run:

```
dotnet --info
```

This command will output something like this

```
.NET Core SDK (reflecting any global.json):
Version:   2.1.302
Commit:    9048955601

Runtime Environment:
OS Name:     ubuntu
OS Version:  18.04
OS Platform: Linux
RID:         ubuntu.18.04-x64
Base Path:   /usr/share/dotnet/sdk/2.1.302/

Host (useful for support):
  Version: 2.1.2
  Commit:  811c3ce6c0

.NET Core SDKs installed:
  2.1.302 [/usr/share/dotnet/sdk]

.NET Core runtimes installed:
  Microsoft.AspNetCore.All 2.1.2 [/usr/share/dotnet/shared/Microsoft.AspNetCore.All]
  Microsoft.AspNetCore.App 2.1.2 [/usr/share/dotnet/shared/Microsoft.AspNetCore.App]
```

```
Microsoft.NETCore.App 2.1.2 [/usr/share/dotnet/shared/Microsoft.NETCore.App]
```

To [install](#) additional .NET Core runtimes or SDKs:
<https://aka.ms/dotnet-download>

So, the environment is ready to run Dot Net Core Code.

Downloading the Source Code

There are several ways of doing downloading the source code from the Git repository.

Downloading directly from the repository

This method is almost similar for both windows and Linux.

Visit the link <https://mygit.th-deg.de/ma02065/inventory-management-system>

The screenshot shows the GitLab interface for the 'Inventory Management System' repository. The left sidebar contains navigation links: Project overview, Details, Activity, Releases, Cycle Analytics, Repository, Issues, Merge Requests, CI/CD, Wiki, and Members. The main content area displays the repository details, including the project name, ID, and a description. Below this, there is a commit history table with columns for Name, Last commit, and Last update. A red arrow points to the 'Download source code' button in the bottom right corner of the commit history section.

Name	Last commit	Last update
./vs/InventoryManagement	Adding ItemsViewModel for storing images on We...	3 weeks ago
./vscode	Adding login and Registration Module	1 month ago
./Areas/Identity/Pages	Adding login and Registration Module	1 month ago
./Controllers	Adding the Authorization for Admin_Employee an...	15 hours ago
./Data	Adding the Authorization for Admin_Employee an...	15 hours ago
./Documentation	AddingDokumentation	4 days ago
./Migrations	Adding the Authorization for Admin_Employee an...	15 hours ago
./Models	Adding the Authorization for Admin_Employee an...	15 hours ago
./Properties	Downgrading the Application From Version 3.1 L...	2 weeks ago

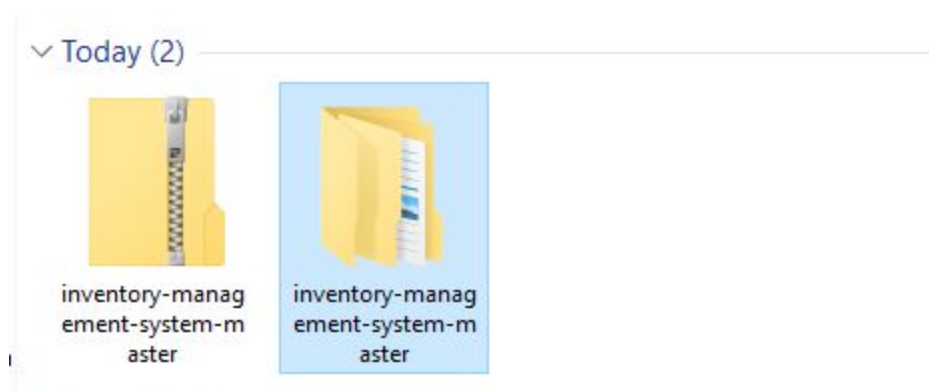
Click on the download button and download the compressed version of your choice.

This screenshot shows the 'Download source code' dropdown menu that appears after clicking the download button. The menu lists four options: zip, tar.gz, tar.bz2, and tar. A red arrow points to the download button in the previous screenshot, indicating the source of this menu.

Download source code

- zip
- tar.gz
- tar.bz2
- tar

Now extract the compressed folder with your favorite unzipper and it will look something like

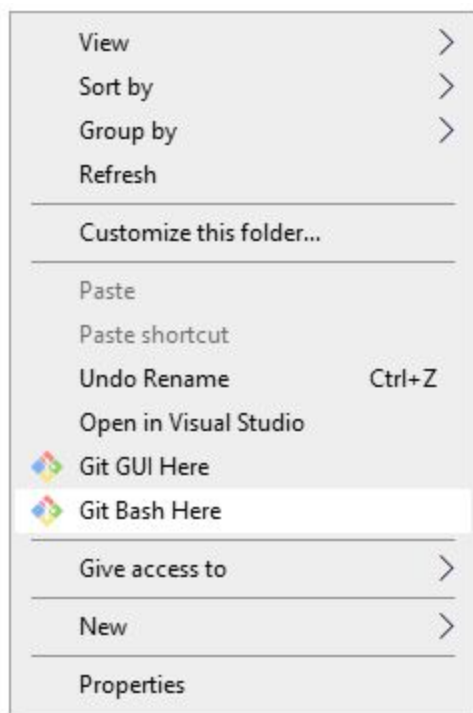


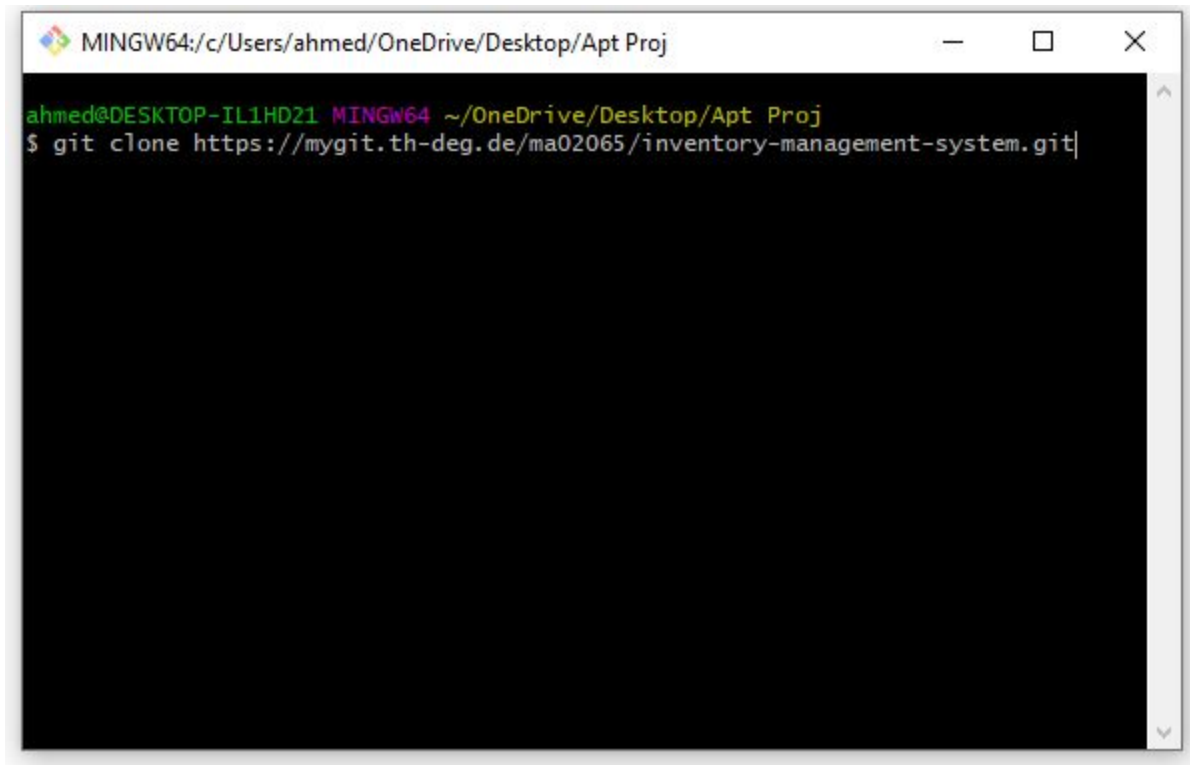
Downloaded version on Windows

Downloading with Git

Here it is supposed that you have already Git installed on your machine. So create new Folder and name it whatever you want.

Open the new folder and right there. Click on the [Git Bash Here](#) as shown in the figure below



A screenshot of a MINGW64 terminal window. The title bar shows the path 'MINGW64:/c:/Users/ahmed/OneDrive/Desktop/Apt Proj'. The terminal content shows the prompt 'ahmed@DESKTOP-IL1HD21 MINGW64 ~/OneDrive/Desktop/Apt Proj' followed by the command '\$ git clone https://mygit.th-deg.de/ma02065/inventory-management-system.git'.

```
MINGW64:/c:/Users/ahmed/OneDrive/Desktop/Apt Proj
ahmed@DESKTOP-IL1HD21 MINGW64 ~/OneDrive/Desktop/Apt Proj
$ git clone https://mygit.th-deg.de/ma02065/inventory-management-system.git
```

A

Bash will pop up and type following command in it

```
Git clone https://mygit.th-deg.de/ma02065/inventory-management-system.git
```

This command will clone the copy of the source code from online repository.

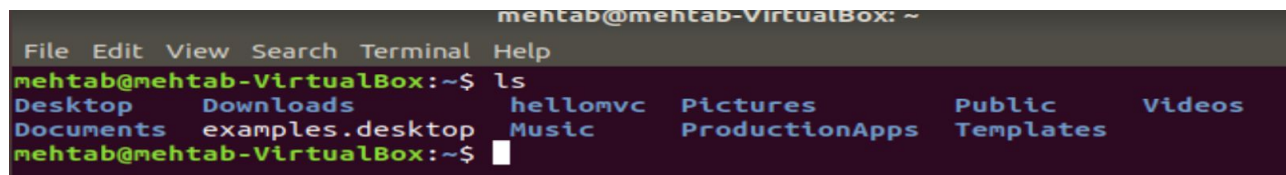


The method for Downloading the source code with Git is same for both Windows and Linux. Here I followed the Windows Method but is same for Linux.

Running the Project -Linux

Open the Terminal and type command

```
ls
```

A screenshot of a terminal window titled 'mentab@mentab-VirtualBox: ~'. The terminal shows the command 'ls' and its output: 'Desktop Downloads hellomvc Pictures Public Videos Documents examples.desktop Music ProductionApps Templates'.

```
mentab@mentab-VirtualBox: ~
File Edit View Search Terminal Help
mentab@mentab-VirtualBox:~$ ls
Desktop Downloads hellomvc Pictures Public Videos
Documents examples.desktop Music ProductionApps Templates
mentab@mentab-VirtualBox:~$
```

In my case, the project is in Downloads folder

```
cd Downloads
```

Type the below command

```
ls
```

```
mehtab@mehtab-VirtualBox:~/Downloads$ ls
inventory  inventory-management-system-master.zip
mehtab@mehtab-VirtualBox:~/Downloads$
```

```
cd Inventory
```

Type the below command

```
ls
```

```
mehtab@mehtab-VirtualBox:~/Downloads/inventory$ ls
appsettings.Development.json  InventoryManagement.csproj      Program.cs
appsettings.json              InventoryManagement.csproj.user  Properties
Areas                          InventoryManagment.csproj        Startup.cs
bin                             InventoryManagement.csproj.user  ViewModel
Controllers                    InventoryManagement.sln          Views
Data                            Migrations                       wwwroot
Documentation                   Models
InventoryDB                     obj
mehtab@mehtab-VirtualBox:~/Downloads/inventory$
```

Type the below command

```
dotnet run
```

Something like below will happen after this command

```
mehtab@mehtab-VirtualBox:~/Downloads/Inventory/Inventory$ dotnet run
Using launch settings from /home/mehtab/Downloads/Inventory/Inventory/Properties/launchSettings.json...
Info: Microsoft.AspNetCore.DataProtection.KeyManagement.XmlKeyManager[0]
      User profile is available. Using '/home/mehtab/.aspnet/DataProtection-Keys' as key repository; keys will not be encrypted at rest.
Hosting environment: Development
Content root path: /home/mehtab/Downloads/Inventory/Inventory
Now listening on: https://localhost:5001
Now listening on: http://localhost:5000
Application started. Press Ctrl+C to shut down.
█
```

Open your favorite browser and type

```
https://localhost:5001/
```

When this window pops up, it means you have successfully run the project

InventoryManagementItemsLogin

Log in

Use a local account to log in.

Email

Password

☐ Remember me?

Log in

[Forgot your password?](#)

[Register as a new user](#)

Use another service to log in.

There are no external authentication services configured. See [this article](#) for details on setting up this ASP.NET application to support logging in via external services.

© 2020 - InventoryManagement

Running the Project -Windows

Open the command prompt and type

```
dir
```

```
Command Prompt
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\ahmed>dir
Volume in drive C has no label.
Volume Serial Number is BE61-E527

Directory of C:\Users\ahmed

24/01/2020  15:48  <DIR>      .
24/01/2020  15:48  <DIR>      ..
24/01/2020  15:47           5.865 .bash_history
20/12/2019  17:09  <DIR>      .config
17/12/2019  20:01  <DIR>      .dotnet
25/12/2019  11:28           65 .gitconfig
17/12/2019  19:14  <DIR>      .nuget
16/12/2019  23:40  <DIR>      .ssh
16/12/2019  22:22  <DIR>      .templateengine
24/01/2020  13:06  <DIR>      .VirtualBox
16/12/2019  21:23  <DIR>      .vscode
22/12/2019  12:56  <DIR>      3D Objects
24/01/2020  15:20  <DIR>      Apt Proj
29/12/2019  14:43      8.388.608 aspnet-loginTesting-5F5D0F99-1EB8-418C-AEBD-20994C2F886D.mdf
29/12/2019  14:43      8.388.608 aspnet-loginTesting-5F5D0F99-1EB8-418C-AEBD-20994C2F886D_log.ldf
22/12/2019  12:56  <DIR>      Contacts
16/12/2019  22:48  <DIR>      Documents
24/01/2020  15:37  <DIR>      Downloads
22/12/2019  12:56  <DIR>      Favorites
24/01/2020  15:22  <DIR>      Inventory
```

Here I suppose the source code exist in C:\User\YourUsername

```
cd Inventory
```

Inventory is the folder name that contains source code. Type

```
Dotnet run
```

Expected Output

```
C:\Users\ahmed\m2>dotnet run
info: Microsoft.AspNetCore.DataProtection.KeyManagement.XmlKeyManager[0]
      User profile is available. Using 'C:\Users\ahmed\AppData\Local\ASP.NET\DataProtection-Keys' as key repository and
Windows DPAPI to encrypt keys at rest.
Hosting environment: Development
Content root path: C:\Users\ahmed\m2
Now listening on: https://localhost:5001
Now listening on: http://localhost:5000
Application started. Press Ctrl+C to shut down.
```



Please press Ctrl+C to stop the running project.

How to use the application

When the application starts at

<https://localhost:5001/>

A login in Page pops up as shown below

InventoryManagement

Items

Login

Log in

Use a local account to log in.

Use another service to log in.

Email

Password

☐ Remember me?

Log in

[Forgot your password?](#)

[Register as a new user](#)

There are no external authentication services configured. See this article for details on setting up this ASP.NET application to support logging in via external services.

© 2020 - InventoryManagement

Here is the default username and password of the application that could be change later on

Username

admin@gmail.com

Password

HelloWorld@176

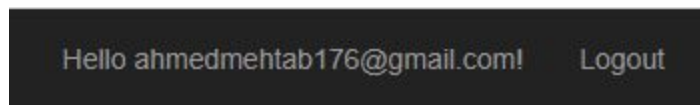


Please change this username and password after the first login. This is for sake of ease during the deployment process. Please follow the next method to change the username and password.

How to Change Username and Password

When login successfully, the login link in the extreme right will disappear and other links appears

The username and Logout as by the below screenshot



Kindly click on the username and new window pops up.

InventoryManagement

Items

Manage Users

Manage Branches

Item and Branches

Manage Tags

Hello ahmedmehtab176@gmail.com

Logout

Manage your account

Change your account settings

Profile

Password

Two-factor authentication

Personal data

Your profile has been updated

Username

ahmedmehtab176@gmail.com

Email

ahmedmehtab176@gmail.com

Send verification email

Phone number

01567654334

Save

© 2020 - InventoryManagement

Click on the Profile and change username and email here. Click on Save button after changing.

Changing the Password

Below Profile, click on the Password Tab, new windows Pops Up

Manage your account

Change your account settings

Profile

Password

Two-factor authentication

Personal data

Change password

Current password

New password

Confirm new password

Update password

© 2020 - InventoryManagement

Enter the current Password and enter new Password. Click on Update Password.

This facility (Manage their Profile setting) is available to all the users except anonymous of the application.

User Interface of the Application for Different Users

The application distinguishes among four different kinds of users. It doesn't contain separate Admin panel and User UI but instead it uses the Built In Api of Dot Net Core called as Identify that provides the authorization to the different parts of the application.

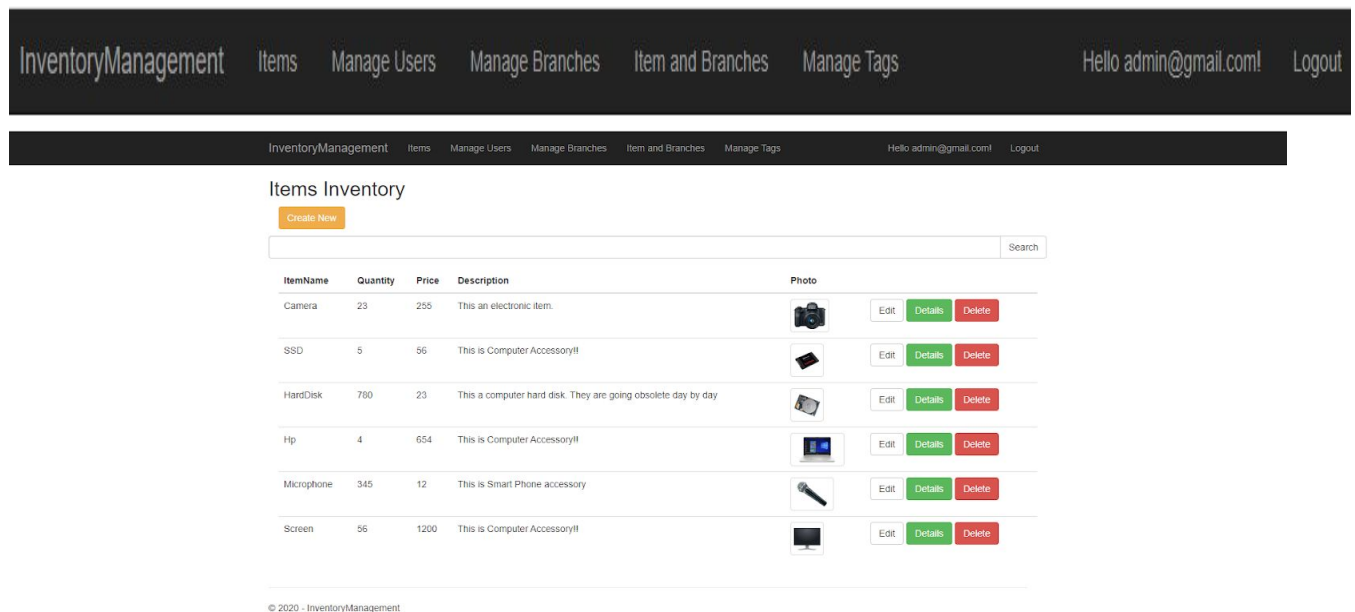
Lets discuss them in details

Admin UI







Admin is super user of the system that can-do CRUD operation on Items, Branches, Tags, Accounts.

Homepage

Application contains the Inventory Items as a Homepage after successful login.



The screenshot displays the 'InventoryManagement' application interface. At the top, a dark navigation bar contains links: 'InventoryManagement', 'Items', 'Manage Users', 'Manage Branches', 'Item and Branches', 'Manage Tags', and a user greeting 'Hello admin@gmail.com!' with a 'Logout' link. Below this is a secondary dark bar with the same navigation links and user information. The main content area is titled 'Items Inventory' and features a 'Create New' button and a search bar. A table lists inventory items with columns for ItemName, Quantity, Price, Description, and Photo. Each item row includes 'Edit', 'Details', and 'Delete' buttons. The footer shows '© 2020 - InventoryManagement'.

ItemName	Quantity	Price	Description	Photo
Camera	23	255	This an electronic item.	 Edit Details Delete
SSD	5	56	This is Computer Accessory!!	 Edit Details Delete
HardDisk	780	23	This a computer hard disk. They are going obsolete day by day	 Edit Details Delete
Hp	4	654	This is Computer Accessory!!	 Edit Details Delete
Microphone	345	12	This is Smart Phone accessory	 Edit Details Delete
Screen	56	1200	This is Computer Accessory!!	 Edit Details Delete







Admin Tasks

Managing Items

Can Create, Edit and Delete Items

Items Inventory

Create New

<input type="text"/>					Search
ItemName	Quantity	Price	Description	Photo	
Camera	23	255	This an electronic item.		Edit Details Delete
SSD	5	56	This is Computer Accessory!!		Edit Details Delete
HardDisk	780	23	This a computer hard disk. They are going obsolete day by day		Edit Details Delete
Hp	4	654	This is Computer Accessory!!		Edit Details Delete
Microphone	345	12	This is Smart Phone accessory		Edit Details Delete
Screen	56	1200	This is Computer Accessory!!		Edit Details Delete



The items page is similar for both admin and employees

Manage Users

Can Create, Edit, Delete and change Role of the User

Click on the Manage User in the Navigation Bar

Manage Users

All Users

Create New

<input type="text"/>					Search
Name	UserName	Email	Role	Branch Name	
Umung	admin@gmail.com	admin@gmail.com	1		Edit Delete
Ali	ali@gmail.com	ali@gmail.com	2		Edit Delete

There is a search functionality available where admin can easily search for a user in huge number of users

Create User

Name

Email

Password

Confirm password

Assign a Role

BranchId

* Please Enter the Required Branch Id

1 for Sale Department

2 for Manufacturing Department

3 for R&D Department

4 for Marketing Department

5 for HR Department

Register

Branch Id is the Id of the Company Branches.

Manage Branches

Can Create, Edit and Delete Branches

Click on the Manage Branches in the Navigation Bar

Manage Branches

Index

Create New

BranchName

Sale

Edit

Delete

Manufacturing

Edit

Delete

R&D

Edit

Delete

Marketing

Edit

Delete

HR

Edit

Delete

Create button is used to add new Branch as shown below.

Create

Branch

BranchName

Create

Back

Manage Tags

Can Create, Edit and Delete Tags

Click on the Manage Tags in the Navigation Bar

Manage Tags

Index

Create New

TagName

Sport

Edit

Details

Delete

Hunting

Edit

Details

Delete

Computers

Edit

Details

Delete

Electronics

Edit

Details

Delete

Education

Edit

Details

Delete

Health

Edit

Details

Delete

Create button is used to add new Tags as shown below.

Create

Tag

TagName

Create

Back to List

Employee UI

Employee can make Crud Operation on Items so only Items Tab is visible to the Employee (Tags also Visible) on the Navigation Bar

InventoryManagement

Items







Manage Tags

Hello ahmedmehtab176@gmail.com! Logout

Items Inventory

Create New

Search







ItemName	Quantity	Price	Description	Photo	
Camera	23	255	This an electronic item.		<div>EditDetailsDelete</div>
SSD	5	56	This is Computer Accessory!!		<div>EditDetailsDelete</div>
HardDisk	780	23	This a computer hard disk. They are going obsolete day by day		<div>EditDetailsDelete</div>
Hp	4	654	This is Computer Accessory!!		<div>EditDetailsDelete</div>
Microphone	345	12	This is Smart Phone accessory		<div>EditDetailsDelete</div>
Screen	56	1200	This is Computer Accessory!!		<div>EditDetailsDelete</div>

External Account UI

External Accountants only view Items summary so they can only view the summary of the Items with no CRUD functionality available to them.

Items Inventory

[Create New](#)

<input type="text"/>					Search
ItemName	Quantity	Price	Description	Photo	
Camera	23	255	This an electronic item.		
SSD	5	56	This is Computer Accessory!!		
HardDisk	780	23	This a computer hard disk. They are going obsolete day by day		
Hp	4	654	This is Computer Accessory!!		
Microphone	345	12	This is Smart Phone accessory		
Screen	56	1200	This is Computer Accessory!!		

External Accountants can only view Items and cannot create, edit, update or delete items.



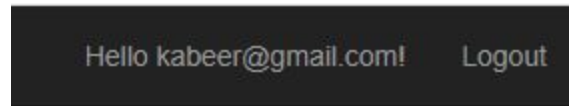
The BranchId for External Accountant is Zero.

Anonymous User

A user without log-in will be considered as an anonymous user, anonymous user is always directed to the login page of the application. Unlike Social media or other sites, a user cannot Register or sign up. The admin of the application creates the username and password for the anonymous user and then can logged into the application.

How a User Can Delete his Account

Login into the system and click on the



The following Menu will Pop up, click on the Personal Data. There are two options ,

Download- Its is used to download all the information the server

Delete – It will permanently delete the user account.

Manage your account

Change your account settings

Profile

Password

Two-factor authentication

Personal data

Personal Data

Your account contains personal data that you have given us. This page allows you to download or delete that data.

Deleting this data will permanently remove your account, and this cannot be recovered.

Download

Delete

Team Members and Responsibilities

Mehtab Ahmed

Back-end: Develop Functional Requirements of our Application using C# and ASP.Net Core.

Muhammad Ali

Front-end and Back-end: Develop basic web page content and control the visual layout. Also providing support in the implementation of backend functionality.

Umang Rajpara

Testing, improving functionality and Documentation: Develop an appropriate testing suite to test functionality developed as well as Documentation.

References

https://www.sqlite.org/aff_short.html :Sqlite

<https://odan.github.io/2018/07/17/aspnet-core-2-ubuntu-setup.html>

<https://www.guru99.com/mvc-tutorial.html> MVC