User Guide

V1.0

Bikes and Parts Management System (BPMS)

Bhuvesh Gupta

UID: 117326611

bgupta1@umd.edu

Contents

Requirements

1. Windows Machine
2. Microsoft Visual Studio Code 2022
3. Web Browser (Microsoft Edge, Google Chrome, Mozilla Firefox etc.)
4. ASP.NET Core 6.0 LTS
5. Git Bash

Packages Required

1. Microsoft.AspNetCore.Identity.EntityFrameworkCore
2. Microsoft.EntityFrameworkCore.SqlServer
3. Microsoft.EntityFrameworkCore.Tools
4. Microsoft.VisualStudio.Web.CodeGeneration.Design
5. NReco.Logging.File by Vitalii Fedorchenko

Setting Up the Project

1. Start by cloning the repository to the local machine from code.umd.edu using Git Bash at your desired location:

```
git clone git@code.umd.edu:bgupta1/ENPM809WFall2022Project-bgupta1.git
```

2. Once cloned, a folder will be created with the name ENPM809WFall2022Project-bgupta1. Navigate into the directory in Git Bash and then checkout to Phase2 branch using:

```
cd ENPM809WFall2022Project-bgupta1/ && git checkout Phase2
```

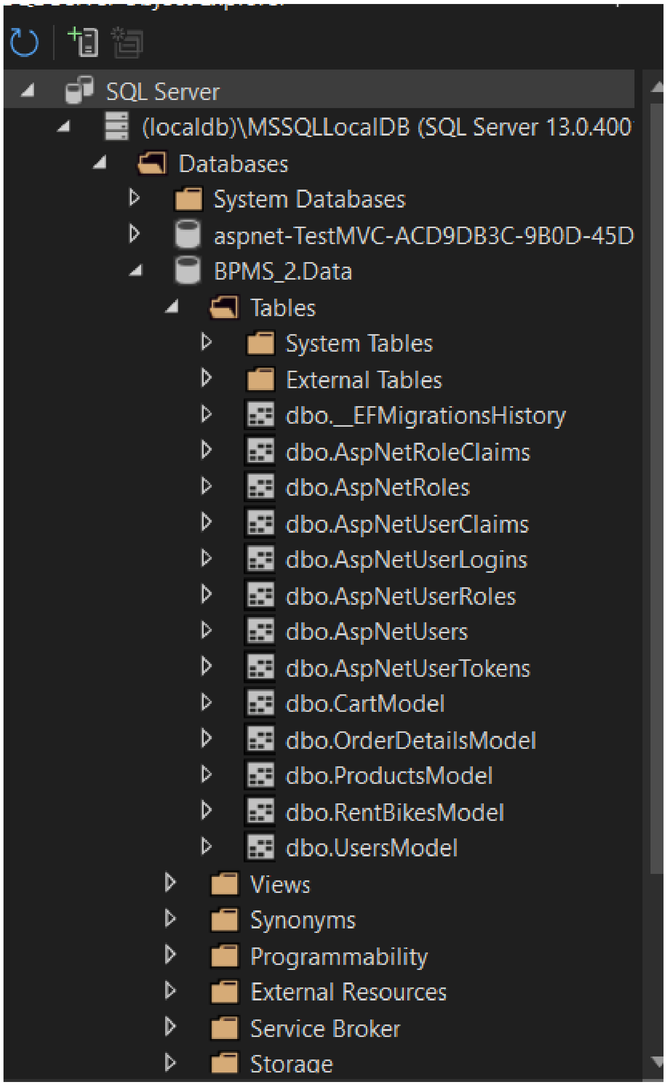
3. In Visual Studio, click on **Open a Project or Solution** and then navigate to the project folder. Then, select the **BPMS-2.sln** file which loads the project into Visual Studio.

4. On initial run, navigate to **Tools** ➔ **NuGet Package Manager** ➔ **Package Manager Console**

Run the following command

```
Update-Database
```

5. To view databases, click on **View** ➔ **SQL Server Object Explorer**. You can find **BPMS_2.Data** database which is used in the web application.

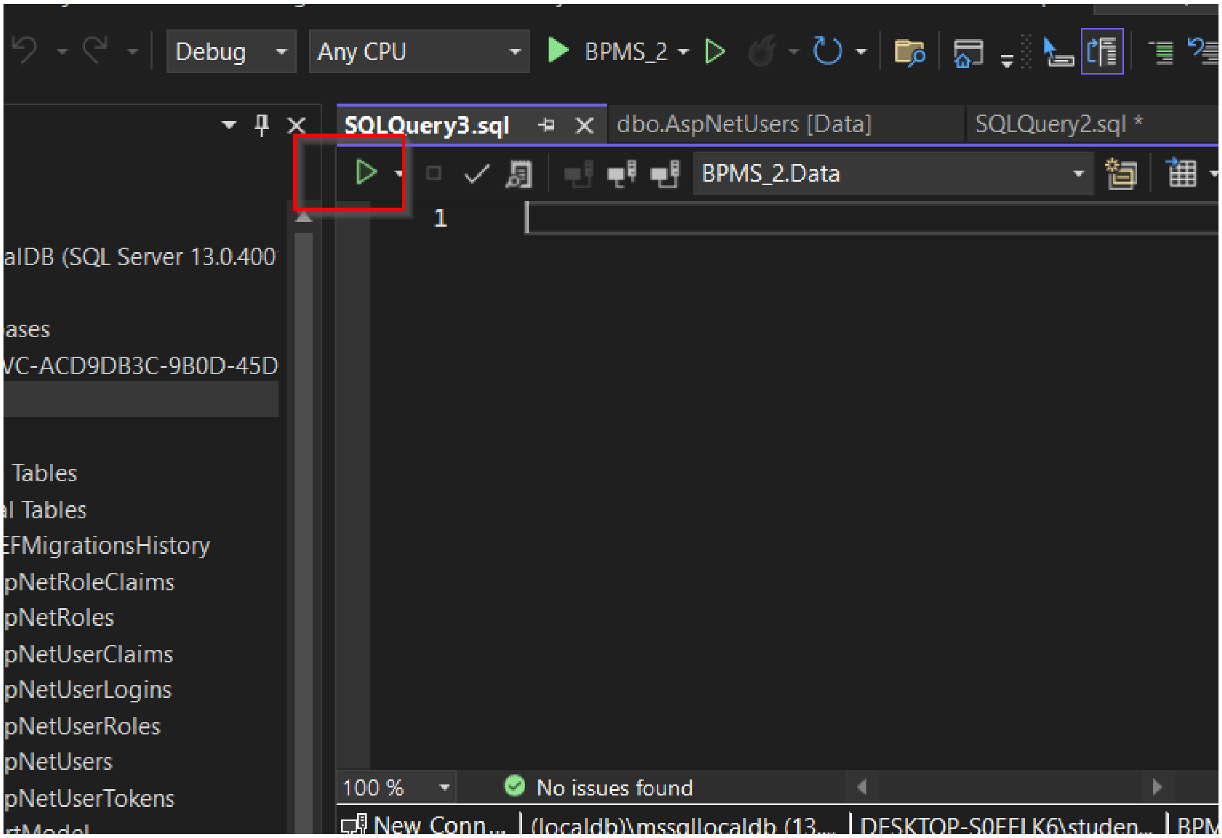


Setting up the Data for the Project Database

6. After running the command right click on **BPMS_2.Data** database and select **New Query** which opens SQL editor which will be used to run a SQL script.

7. Open the **scripts** folder inside the directory **ENPM809WFall2022Project-bgupta1/BPMS-2/BPMS-2/** and open the file named **BPMS-2Data_Script.sql** using Notepad or any Text Editor.

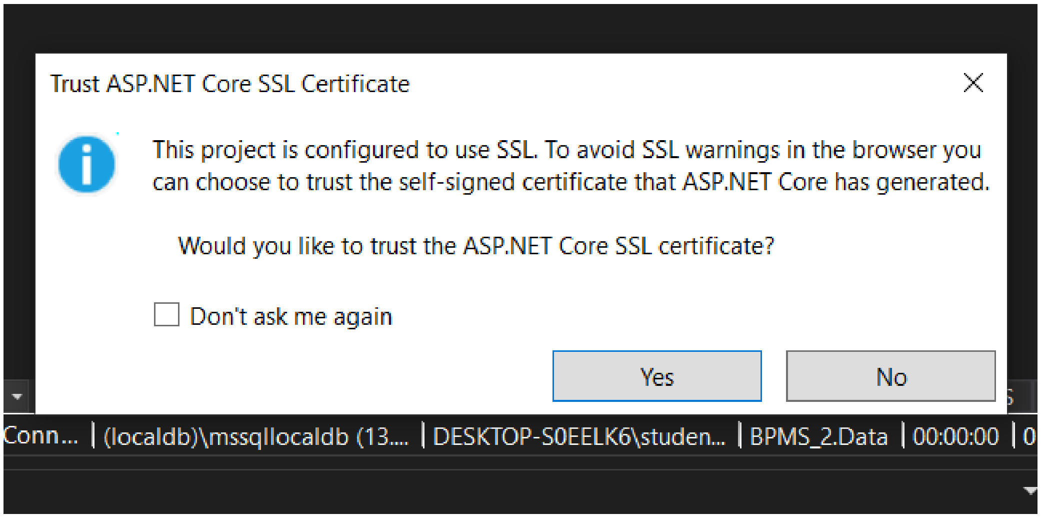
8. Copy the script and paste in the SQL Editor and execute the script using **Ctrl + Shift + E** or the green button at the top left corner.



No issues found

9. Now run the project by clicking on **BPMS_2** button at the center of the toolbar which starts building the project and opens the web application in a web browser.

If the following message pops up, then select **Yes



Trust ASP.NET Core SSL Certificate

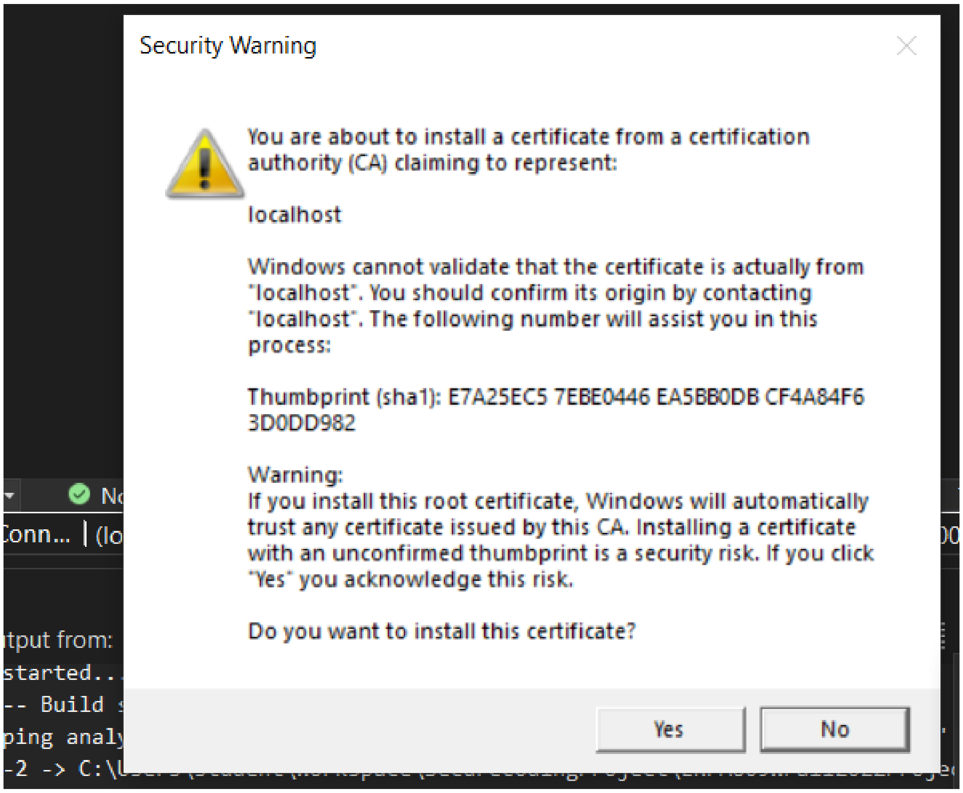
This project is configured to use SSL. To avoid SSL warnings in the browser you can choose to trust the self-signed certificate that ASP.NET Core has generated.

Would you like to trust the ASP.NET Core SSL certificate?

□ Don't ask me again

Yes
No
Conn... | (localdb)\mssqllocaldb (13.... | DESKTOP-S0EELK6\studen... | BPMS_2.Data | 00:00:00 | 0

Select **Yes again



Security Warning

You are about to install a certificate from a certification authority (CA) claiming to represent:

localhost

Windows cannot validate that the certificate is actually from "localhost". You should confirm its origin by contacting "localhost". The following number will assist you in this process:

Thumbprint (sha1): E7A25EC5 7EBE0446 EA5BB0DB CF4A84F6 3D0DD982

Warning:
If you install this root certificate, Windows will automatically trust any certificate issued by this CA. Installing a certificate with an unconfirmed thumbprint is a security risk. If you click "Yes" you acknowledge this risk.

Do you want to install this certificate?

About the Project

The project is about the bikes and parts management system where the users can **buy**, **rent** and **return** the rented bikes. Admins **maintain the inventory** records.

User Roles and Permissions

There are 2 user roles in the application:

Patron – Normal user which refers to all the bike shop customer and they have lesser access.

Admin – Admins with higher level of access.

Project Functionality Detailed

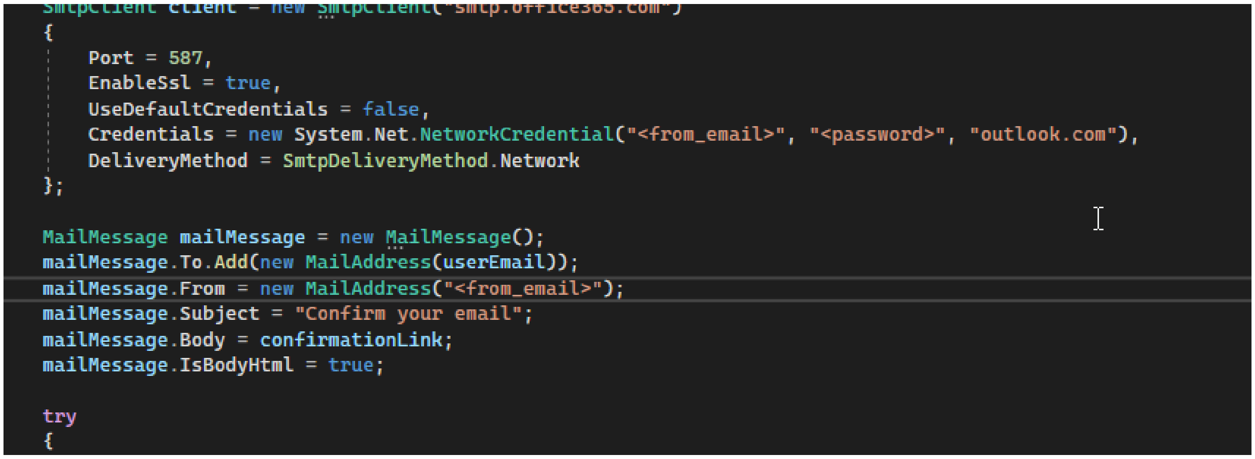
User Functions

➔Register

Every student can register for an account with their umd.edu email address.

Upon clicking Register, an email would be sent to the student to verify and activate their account. They can not login without the verification.

***For the project to send registration email, please add your outlook email address at the placeholder <from_email> and passwords at the placeholder <password> in the file ENPM809WFall2022Project-bgupta1\BPMS-2\BPMS-2\Utils\EmailSender.cs*



```
SmtpClient client = new SmtpClient("smtp.office365.com")
{
    Port = 587,
    EnableSsl = true,
    UseDefaultCredentials = false,
    Credentials = new System.Net.NetworkCredential("<from_email>", "<password>", "outlook.com"),
    DeliveryMethod = SmtpDeliveryMethod.Network
};

MailMessage mailMessage = new MailMessage();
mailMessage.To.Add(new MailAddress(userEmail));
mailMessage.From = new MailAddress("<from_email>");
mailMessage.Subject = "Confirm your email";
mailMessage.Body = confirmationLink;
mailMessage.IsBodyHtml = true;

try
{
```

➔Login

Students can login to their verified account.

➔Home

Takes to the bike shop home page.

➔Rent Bikes

The page shows a list of bikes which can be rented by the student. A bike can be rented for 6 months only and would show the return date to the student once they rent the bike.

Only 1 bike can be rented at a time.

-Keep Shopping

After selecting the bike, the user can click this button to keep browsing more bikes before placing an order. Only 1 bike would be allowed to rent at a time. The user needs to return the previously rented bike to rent a new bike.

-Checkout

Finalizes and executed the order.

-Remove From Cart

Removes the product from the cart and lets user update its product choice.

➔Buy Parts and Bikes

The page shows the various bikes and bike parts for purchase. Any number of bikes or parts can be purchased by student. Every product has its inventory count.

****Quantity-*** It is the field where the user can enter the number of parts or bikes it needs to purchase.

-Keep Shopping

After selecting the bike or part, the user can click this button to keep browsing more bikes and parts before placing an order. The user can buy as many products as it wants but the number should be less than the inventory count.

-Checkout

Finalizes and executed the order.

-Remove From Cart

Removes the product from the cart and lets user update its product choice and quantity.

➔Order History

It shows the user's history of its purchases along with the return date of its rented bike (if any)

➔Return Bike

The page allows the user to return the previously rented bike. If the user exceeds the due date in returning the bike then a fine amount would be calculated for additional payment.

➔ Change Password

It allows the user to change their password.

➔**Logout**

Logs out of the account.

Admin Functions

All the functionalities which a patron has plus some additional and elevated privilege such as:

➔**Manage Products**

The admins can manage all the products by **Create, Edit** and **Delete** feature.

➔**Manage Rental Bikes**

The admins can manage the rental bikes inventory by having feature like **Create, Update** and **Delete.**

➔**Sales**

The admins can view all the orders placed by all the users and look at their total sales for financial records.

***Assumptions:*

The patron completes its rental order and purchases separately and does not mix the two types of purchases.

Test Account Credentials:

Patron User Role

Username: user1

Email: user1@umd.edu

Password: VK#RRJ9GoqgD0rkZ

Username: user2

Email: user2@umd.edu

Password: c65sxsIIN7DR&HhZ

Username: user3

Email: user3@umd.edu

Password: g679SW9QNsd!l#gW

Admin User Role

Username: admin1

Email: admin1@umd.edu

Password: @#oNVbk8azPX3JbJ