**Municipality Project Description**

**Да се запиша и платя до 15.08, защита на 20.08, фрафик ще е ясен на 17.08 (мейл от** 26.05.2023 16:58**):**

<https://softuni.bg/trainings/4245/retake-session-may-2023>

**Municipality Mobility System** is an online platform that is used to ensure mobility transport solution in sities and towns.

## Technological Requirements and Overview

* The application must be implemented using **ASP.NET Core 6.0** Framework or **ASP.NET Core 5.0** Framework.
  + The application must have at least **15** web pages (views)
  + The application must have at least 6 **entity models**
  + The application must have at least 6 **controllers**
* Use **Visual Studio 2022 / JetBrains Project Rider**.
  + Use the **Razor** template engine for generating the UI
    - Use **sections** and **partial views**.
  + Optionally, you could also use Web API to create a RESTful service and use JavaScript / TypeScript for the **Front-End**
* Use **Microsoft SQL Server** as Database Service
  + Optionally, use multiple storages, e.g. files, other Web services, databases (e.g. MySQL / MongoDB / Cassandra / etc.)
* Use **Entity Framework Core** to access your database
  + If you need additional connectors to other databases, feel free to use them
* Use **MVC Areas** to separate different parts of your application (e.g. area for administration)
* Adapt the default **ASP.NET Core site template** or get another free theme
  + Use responsive design based on **Twitter Bootstrap / Google Material design**
  + Or just design your own
* Use the standard **ASP.NET Identity System** for managing **Users** and **Roles**
  + Your registered users should have at least one of these roles: **User** and **Administrator**
  + If you need, implement your own user management system
* Optionally, use **AJAX** request to asynchronously load and display datasomewhere in your application
* Write **Unit Tests** for your logic, controllers, actions, helpers, etc.
  + You should **cover** at least **65%** of your business logic.
* Implement **error handling** and **data validation** to avoid crashes when invalid data is entered
  + Both **client-side** and **server-side**, even at the database(s)
* Handle correctly the special **HTML characters** and tags like **<br />** and **<script> (escape special characters)**
* **Use Dependency Injection**
  + **The built-in one in ASP.NET Core is perfectly fine**
* **Optionally, use AutoМapping**
* **Prevent from security vulnerabilities like SQL Injection, XSS, CSRF, parameter tampering, etc.**

### VehiclePark

* Has an Id – a **string, Primary Key**
* Has a Name – a string with **min length** **2** and **max length 20** (**required**)
* Has an Email – a string with **min length** **5** and **max length 30** (**required**)
* Has a Phone – a string with **min length** **6** and **max length 20 (before hashed)** (**required**)
* Has an Adress – a string with **min length** **8** and **max length 40 (before hashed)** (**required**)
* Has **ServiceId** – an **integer, foreign key (required)**
* Has **Service**– a **Service** (**required**)
* Has **VehicleWashId** – an **integer, foreign key (required)**
* Has **VehicleWash**– a **VehicleWash** (**required**)
* Has **Vehicles** – a collection of type **Vehicle**

### Vehicle

* Has Id – a unique **integer, Primary Key**
* Has RegistrationNumber – a string with min length **6** and max length **8** (**required**)
* Has Model – a string with min length **10** and max length **50** (**required**)
* Has **Engine**Type – a string with min length **3** and max length **20**
* Has **Rating** – a int with min value **1** and max value **6**
* Has PricePerHour – a decimal with min value **2.00** and max value **200.00** (**required**)
* Has ImageUrl – a string (**required**)
* Has Description – a string with min length **5** and max length **5000** (**required**)
* Has **CategoryId** – an **integer, foreign key (required)**
* Has Category – a Category (**required**)
* Has **VehicleParkId** – an **integer, foreign key (required)**
* Has **VehiclePark**– a **VehiclePark** (**required**)
* RenterId – a nullable string
* Has **ForRepearing** – a bool property (default = false)
* Has **ForCleaning** – a bool property (default = false)
* Has **IsActive** – a bool property (default = true)
* Has **RentedPeriod–** a nullable **TimeSpan**
* Has **RepairingTerm** – a nullable DataTime
* Has **FailureDescription -** a nullable string
* Has **RepairsCount** a int with min value **1** and max value **100**
* Has **RentsCount** a int with min value **1** and max value **10000**
* Has **OrderedParts** – a collection of type **PartsOrder**

### RepairCenter

* Has Id – a unique **integer, Primary Key**
* Has Name – a string with min length **10** and max length **50** (**required**)
* Has ImageUrl – a string (**required**)
* Has **VehicleParkId** – an **integer, foreign key (required)**
* Has **VehiclePark**– a **VehiclePark** (**required**)
* Has **Vehicles** – a collection of type **Vehicle**

### WashingCenter

* Has Id – a unique **integer, Primary Key**
* Has Name – a string with min length **10** and max length **50** (**required**)
* Has ImageUrl – a string (**required**)
* Has **Vehicles** – a collection of type **Vehicle**
* Has **VehicleParkId** – an **integer, foreign key (required)**
* Has **VehiclePark**– a **VehiclePark** (**required**)

### PartsOrder

* Has an Id – a **string, Primary Key**
* Has a Title – a string with **min length** **5** and **max length 20** (**required**)
* Has **VehicleId** – an **integer, foreign key (required)**
* Has **Vehicle**– a **Vehicle** (**required**)
* Has **DeliveryTerm**– a DateTime
* Has IsDelivered– a bool property
* Has TotalPrice – a decimal property with min value **0.00** and max value **10000.00** (**required**)

### Category

* Has Id – a unique **integer, Primary Key**
* Has Name – a string with min length **5** and max length **50**
* Has **Vehicle**s – a **collection** of **Vehicle**

Implement the entities with the **correct datatypes** and their **relations**.

**Areas:**   
**User area** (For All Users , Boss and Administrator)

* **Servisce area** (For **Administrator,** **Boss,** **Manager, Mechanic** and **Cleaner**)

**Roles:**

**Administrator,** **Boss,** **Mechanic, Cleaner** and **Manager** (Administrator **gives the roles**)

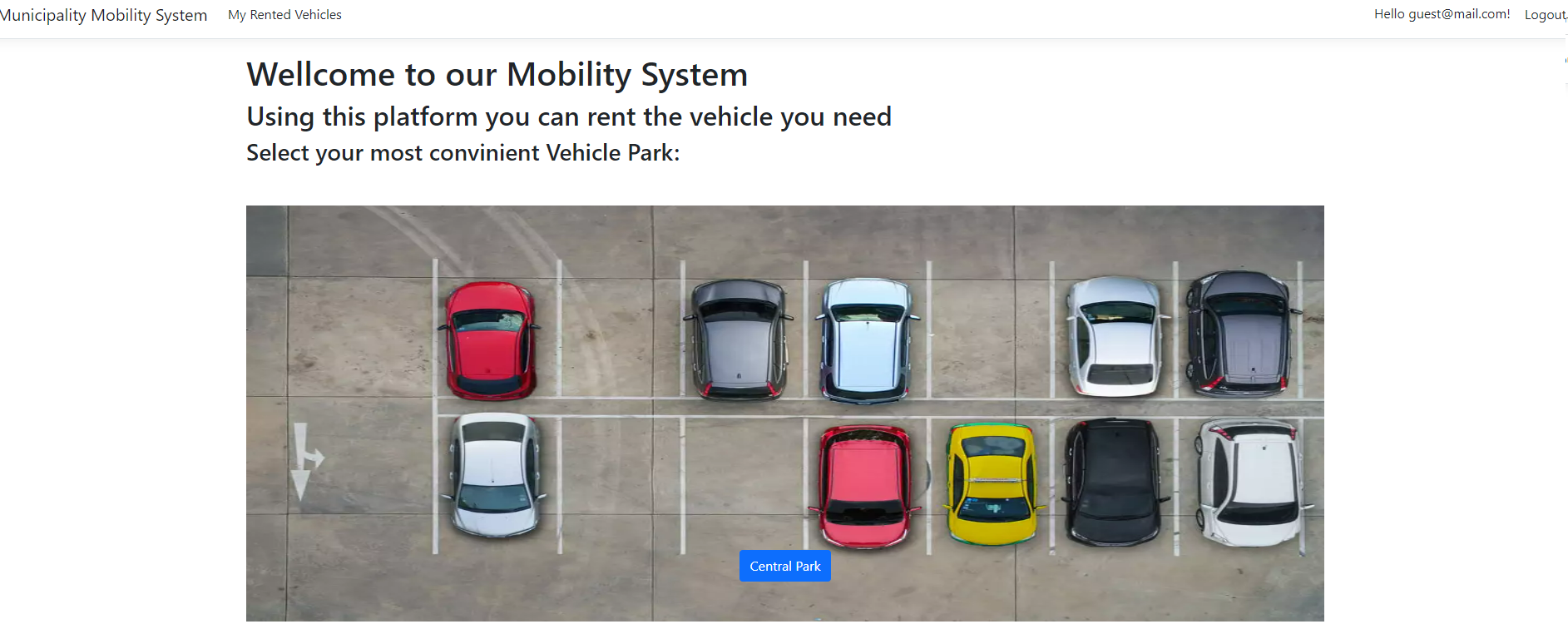
**To Add Web API for a statistics :**

* **Free Vehicles** (For User Area - into Home Index behind a Carosell and Service Area - into Home Index behind thes table with Vehicle Parks):
* **Rented Vehicles** ( For Service Area - into Home Index behind a table with Vehicle Parks):
* **ThreeMostRepairedVehicles** ( For Service Area - into Repair Center behind a table with Vehicles):
* Show Statistics</button> in homePageView Or directly visualized on the views

**Servise and VehicleWash will be added by the Administrator**

## Page Requirements

### Index Page /VehicleParks/ All (logged-out user, User area) – 1

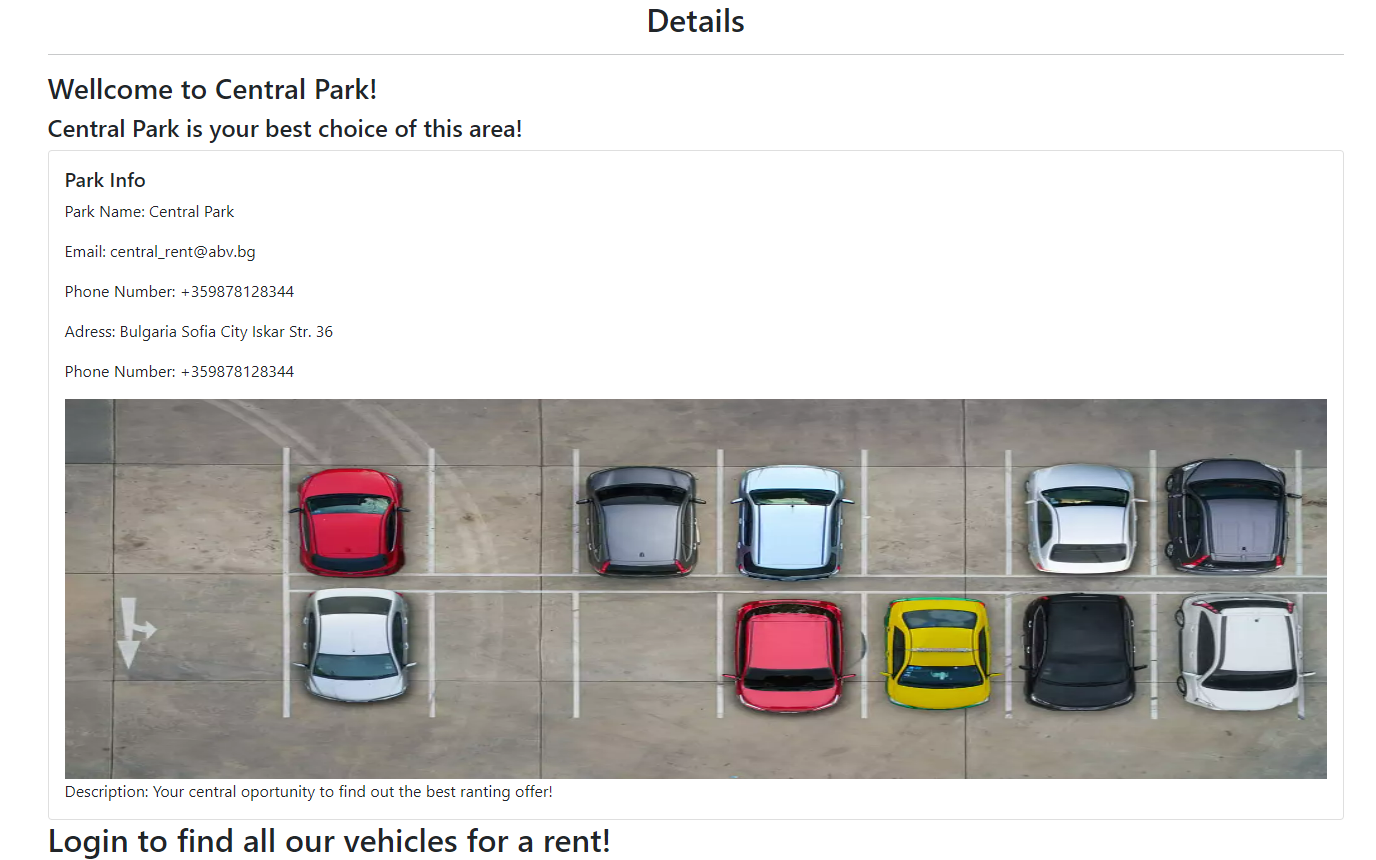


All **Vehicle Parks** in **Carusel**, (behind it dropdown for a selection of nearest **Vehicle Park -** maby, if understand how**)**

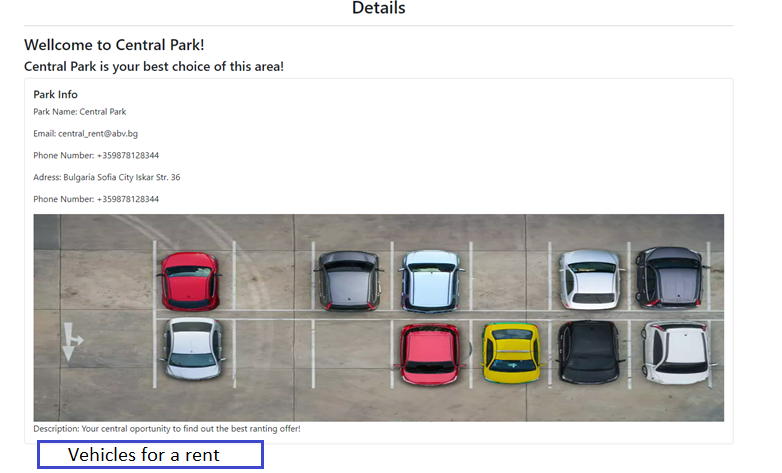


After selecting **Vehicle Park** Redirect to **/Vehicles/Details**

### /VehiclePark/Details (logged- out user, User area) - 2

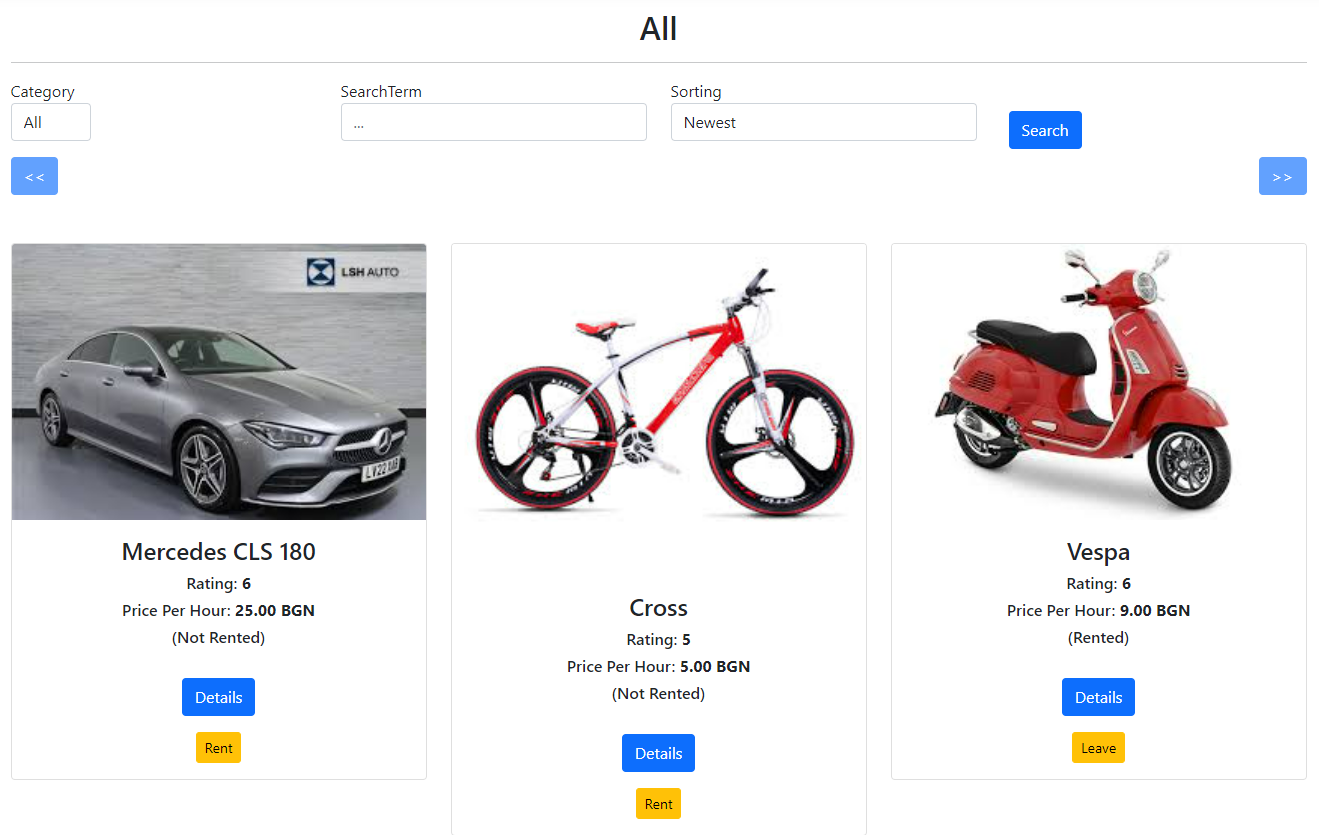


When **User** is **loged in – appear button for finding vehicles:**



The button redirect to VehiclePark/All with Surch and Sort. **The vehicles are all vehicles**, because the Idea is the selected vehicle to be rented from conviniant vehicle park. The Manager make the necesery logistic and communicate with client!

### / VehiclePark/All (logged- in user, User area) – 3



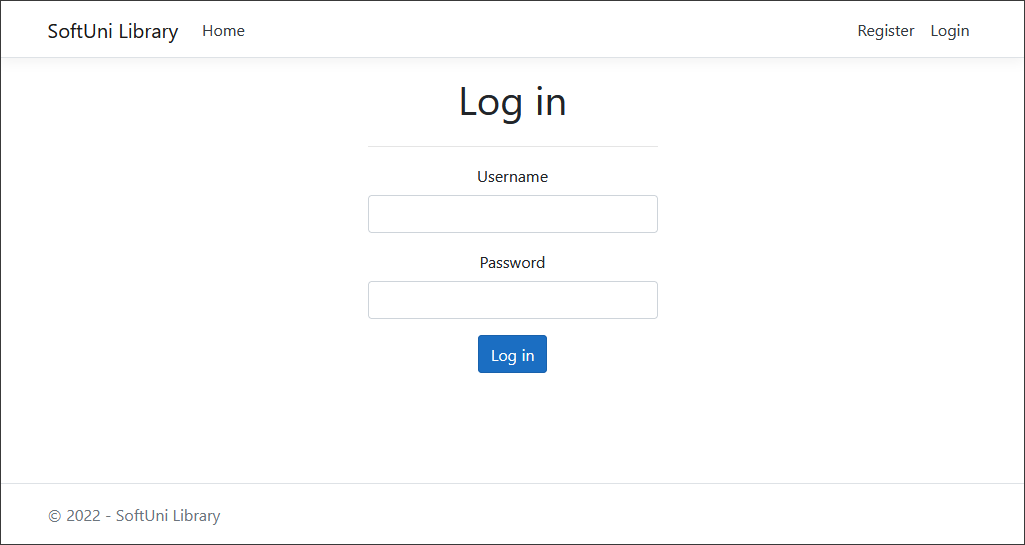
Search redirect to VehiclePark All, ordered on user selected order.

The buttons behind vehicle are for Details, Rent or Leave!

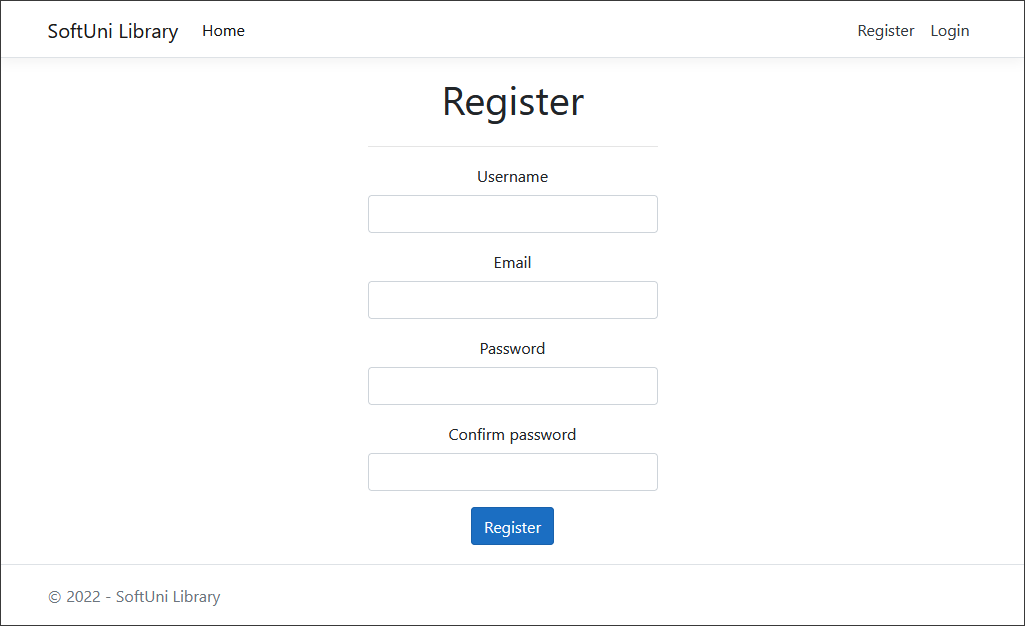
**Index Page (logged- in user,** Service area**)**

* **Free Vehicles** (For **User Area** - into Home Index behind a **table** with **Vehicle Parks**):
* **Rented Vehicles** ( For **Service Area** - into Home Index behind a table with Vehicle Parks):
* **ThreeMostRepairedVehicles** ( For **Service Area** - into Repair Center behind a table with Vehicles):

### Login Page (logged-out user, User area ?Service) - 4



### Register Page (logged-out user, User area ?Service) - 5



**NOTE**: If the user is logged in and tries to go to the **Home page**, the application must redirect them to the **/VehicleParks/All**.

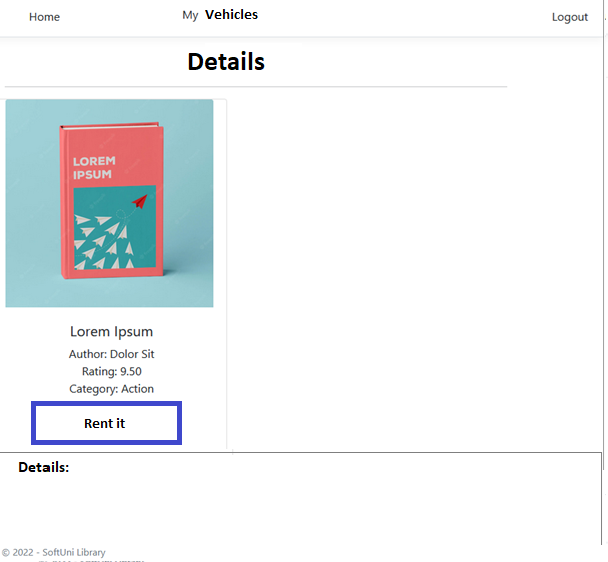
**/Vehicles/All (logged-in user in Role Administrator and Boss,** Service area**) - 6**

All vehicles in table view (free, rented, forRepairing, forCleaning)

**The statistic behind the table**

* **Free Vehicles** (behind a **table** with **Vehicles**):
* **Rented Vehicles** ( behind a **table** with **Vehicles**):
* **ThreeMostRepairedVehicles** ( behind a **table** with **Vehicles**):

### /Vehicles/Details (logged-in user, User area) - 7



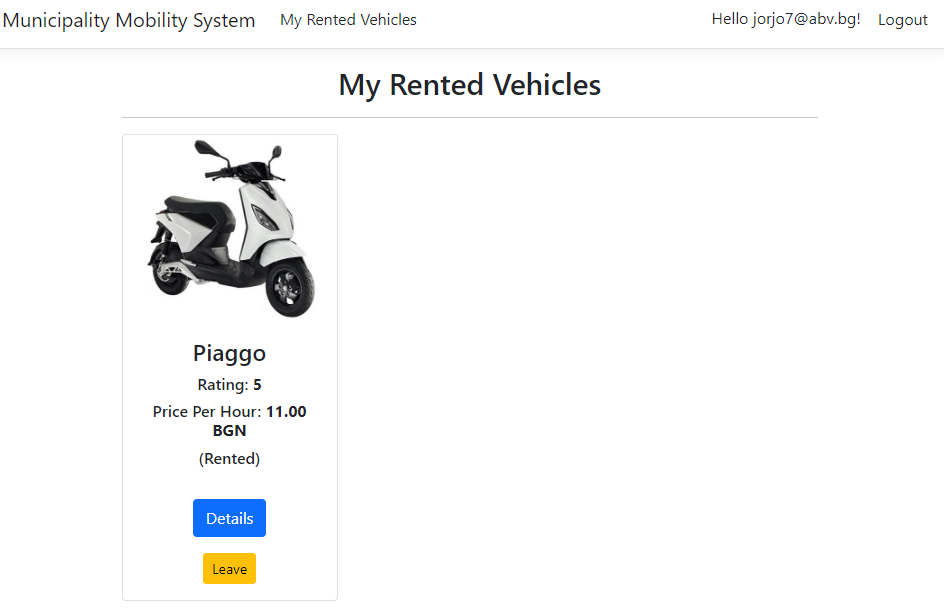
**/Vehicles/** **Details (logged-in user in Role Administrator and Boss,** Service area**) - 8**

All vehicle in table view with properties: free, rented, forRepairing, forCleaning

**The statistic behind the table:**

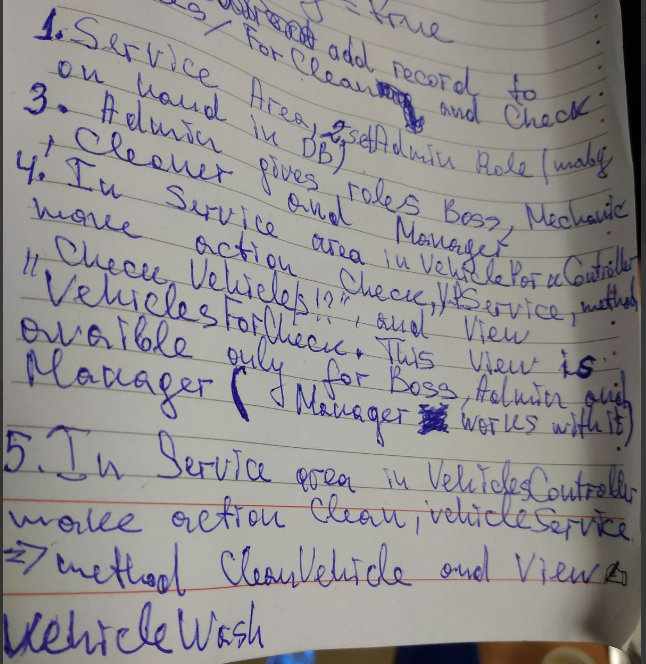
* RentCount:
* RepairsCount:
* TotalRentsIncome:
* TotalForRepairs:

### / Vehicles /Mine (logged-in user, User area) - 7



When **Leave**, Set RenterId=null, set ForClean = true card disappear from **Mine, Redirect** to **Mine and set record** to **VehiclesForCheck.** In **All** the **left vehicle do not appear,** because of **ForClean = true**!!! When **ForClean = false**, the **left vehicle appear to All** !!!

**(in Service area)**



**Последно наети по някакъв начин да стоят в май виакълс.**

Съобщение при наемане в колко часа и къде може да се получи ключа.

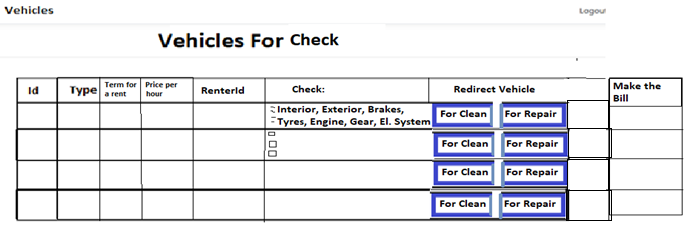
**Extras on second stage:**

Гугъл мапс да интегрирам, евентуало чат м/у потребители и мениджър, както и вътрешна комуникация. Система за плащане.

Логин с външен доставчик.

### /Vehicles /CleanAndCheck

**(logged-in user in role Manager,** Servisce area**) – 14**



Button For Clean adds the vehicle in **Vehicles for Clean** table in **Vehicles Wash Page**!

Button For Repair adds the vehicle in **Vehicles for Repair** table in **Vehicles Service Page**!

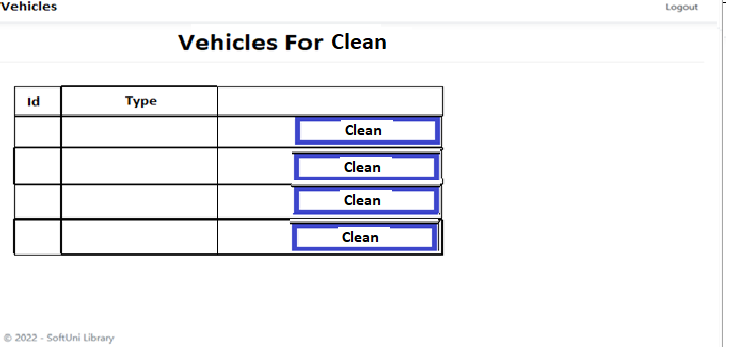
When vehicle is for **Repair** after **repairing** thevehicle is set for **Cleaning**

**ForCleaning** = false, and on this way Add it to All (Free) Vehicles collection

**Redirect to ForCleaning**

### /Vehicles /RorClean

**(logged-in user in role Cleaner and Boss,** Servisce area**) – 12**

****

With Clean button sets **ForCleaning = false**, and Vehicle appear to **All** !

### /Vehicles /Service

**(logged-in user in role Mechanic and Boss,** Servisce area**) – 12**

****

If parts do not needed:

Set ForRepair = false, and on this way Add it to All (Free) Vehicles collection

**Redirect to RorRepair**

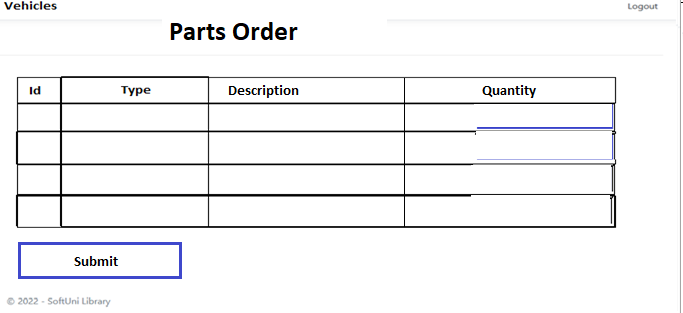
**else:**

Using **Description** make **Parts Order** and **after Delivering** can **repair** and **set** ForRepair = false, and on this way Add it to All (Free) Vehicles collection

**Redirect to RorRepair**

### /Vehicles /PartsOrder

**(logged-in user in role Mechanic and Boss,** Servisce area**) –**

****

**До тук!!!**

If checkboxes are not checked – Delete from My Vehicles, add it to **All** and **Redirect** to **All**,

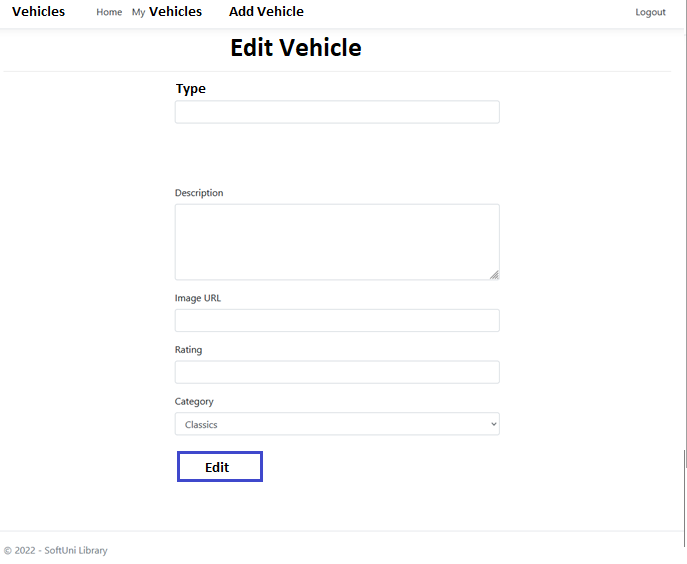
else – Delete from My Vehicles, add it to **Service** or **Vehicle Wash** and **Redirect** to **All. If “it need to be repaired”** is checked, the **text Area** for Description have to appear.

### /Vehicles/Add (logged-in user in role Administrator and Boss, Servisce area) - 9

### / Vehicles /AddToCollection?vehicleId={vehicleId} (logged-in user in role Administrator, Servisce area)

Adds the selected vehicle to the collection of **All vehicles**. If the vehicle is already in **All vehicles**, it shouldn't be added. If everything is successful, the Administrator must be redirected to the home "**/Vehicles/All**" page.

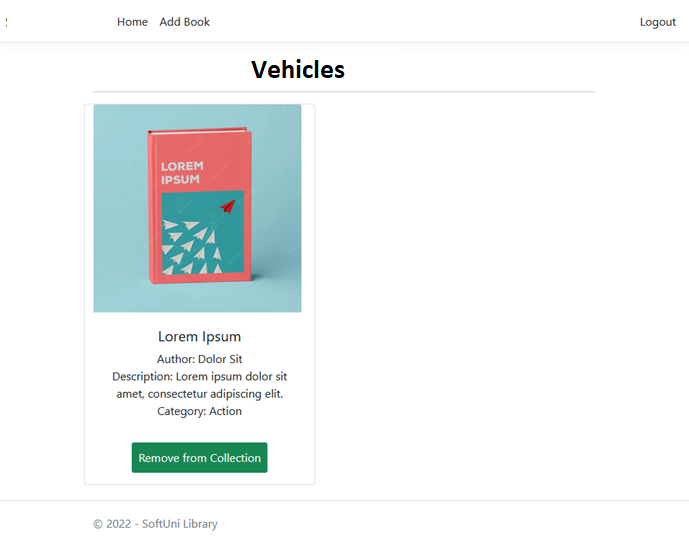
**/Vehicles/Edit (logged-in user in role Administrator and Boss,** Servisce area**) - 10**



If everything is successful, the Administrator must be redirected to the home "**/Vehicles/All**" page.

### /Vehicles /RemoveFromCollection? vehicleId={vehicleId}

**(logged-in user in role Administrator and Boss,** Servisce area**) - 11**

****

Removes the selected vehicle from **All vehicles**. If everything is successful, the **Administrator** must be redirected to collection "**/ Vehicles / All** " page.

**NOTE**: The templates should look **EXACTLY** as shown above.

Usersin role **Manager** or **Mechanic** are registered from **Administrator**

If parts do not needed:

## Functionality

The functionality of the **Municipality Mobile System** Platform is simple.

### Users

Guests can Register, Login and view the Index Page.

**Logged Users** can **rent** and **release from rent**

Administrator can Add (in /Vehicles /All) **, Edit** and **Delate** Vehicles From the Home Page (/Vehicles /All),they can also view statistic Info about each one of those Vehicles and about all **Vehicle Parks**. Also **Administrator** gives roles to **Boss**, **Mechanic** and **Manager.**

**Boss** can Add (in /Vehicles /All) **, Edit** and **Delate** Vehicles From the Home Page (/Vehicles /All),they can also view statistic Info about each one of those Vehicles and about all **Vehicle Parks**.

**Mechanic** can make **Parts Order** and after delivered parts can repair Vehicles.

**Manager** can check, redirect for clean or service and make a bill. Bill have to appear in My Bills page. My Bills page have to visualize the bills in table view each sorted by DateTime and with status “paid “ or “for paying”. The bills with status “for paying” must have button “Pay”, that redirect the user to Page for Online Paying.

### Redirections

* Upon successful Registration of a User, you should be redirected to the Login Page.
* Upon successful Login of a User, you should be redirected to the / Vehicles /All.
* Upon successful Creation and Adding a Vehicle, **Administrator** should be redirected to the / Vehicles /All.
* Upon successful Removal of a Vehicle from the All collection, **Administrator** should be redirected to the / Vehicles /All.
* If a **Administrator** tries to **add** an **already added** Vehicle to All **collection**, he should be redirected to /Books/All (or just a page refresh).
* Upon successful Logout of a User, you should be redirected to the Index Page.
* If any of the **validations** in the POST forms **don't pass**, **redirect** to the **same page** (**reload/refresh** it).

## Security

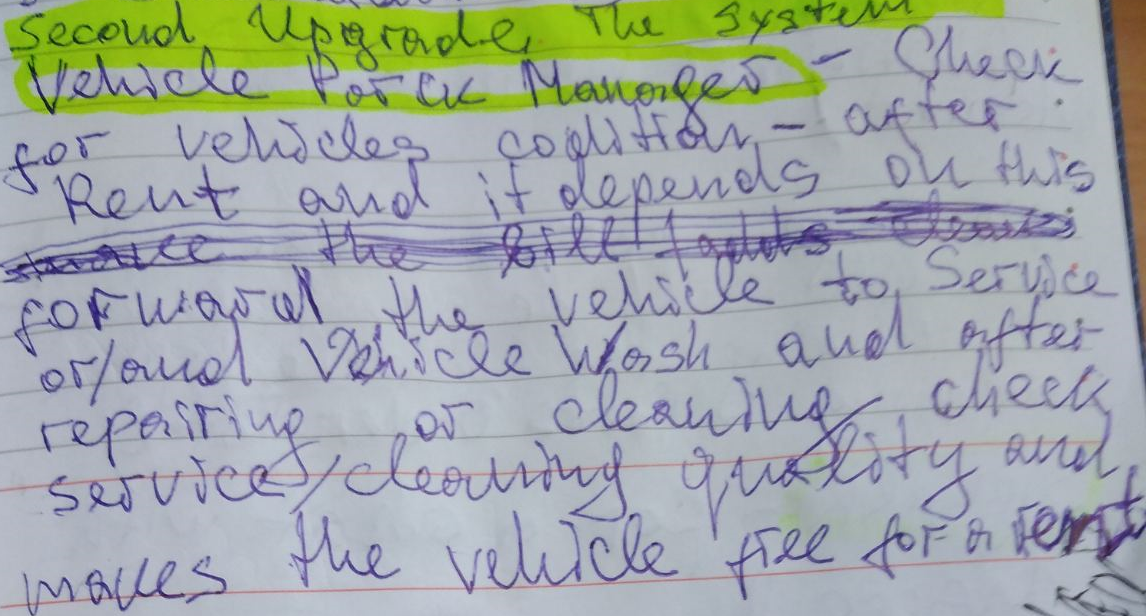
The Security section mainly describes access requirements. Configurations about which users can access specific functionalities and pages:

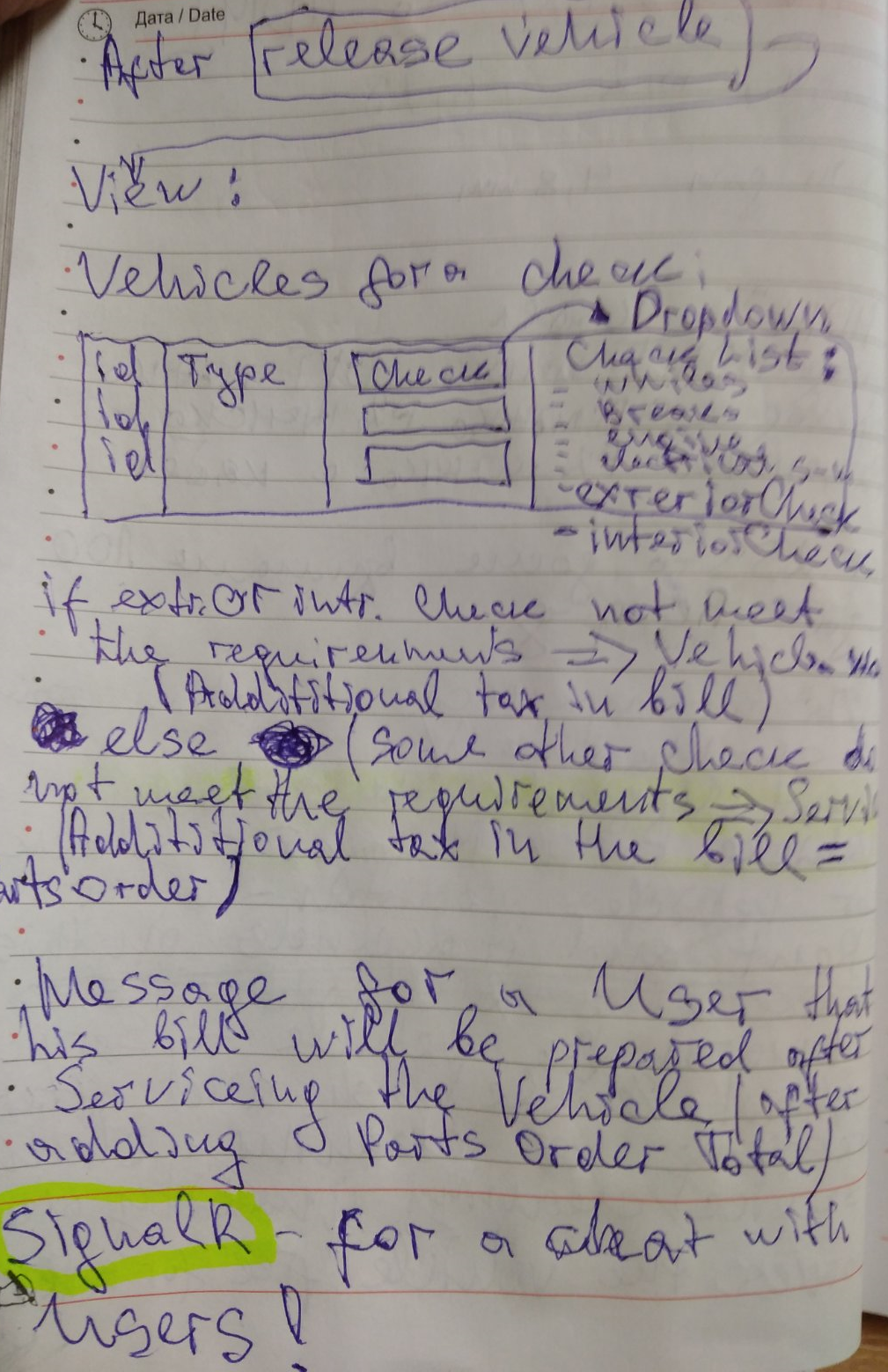
* Guest (not logged in) users can access the Index page.
* Guest (not logged in) users can access the Login page.
* Guest (not logged in) users can access the Register page.
* Guests (not logged in) cannot access Users-only pages.
* Users (logged in) cannot access Guest pages.
* Users (logged in) can not access the Vehicles /Add page and functionality.
* Users (logged in) can access the Vehicles /All page.
* Users (logged in) can access the **My** Vehicles page.
* Users (logged in) can access Logout functionality.

## Code Quality

Make sure you provide the best architecture possible. Structure your code into different classes, follow the principles of high-quality code (**SOLID**). You will be scored for the Code Quality and Architecture of your project.

## Second Upgrade of the System:





**Data for Seed:**

**Name**

**Email**

**Phone**

**Adress**

**Image URL**

**Description**

Your best northen possibility for renting a vehicle!

**Name**

**Email**

**Phone**

**Adress**

**Image URL**

**Description**

**Model**

**RegistrationNumber** = "C000004",

**Engine Type**

**Description** 

**Image URL**

**Price per hour**

**VehicleParkId** 2

**Category**                                                                   

**Model**

**RegistrationNumber =** "C000005",

**Engine Type**

**Description** 

**Image URL**

**Price per hour**

**VehicleParkId**

**Category**                                                                   

Central repair center

https://media.istockphoto.com/id/628096148/vector/auto-repair.jpg?s=612x612&w=0&k=20&c=GCVOh9kmmlYAoRvT\_g86559RDmGA06w5m4nEYqsfE9w=

**Bulgaria Sofia City Iskar Str. 36**

VehicleParkNumber 2

Eastern repair center

https://mybayutcdn.bayut.com/mybayut/wp-content/uploads/Cars-lifted-in-a-service-centre.jpg

**Bulgaria Sofia City Mladost 4**

VehicleParkNumber 1

Western repair center

https://conceptualminds.com/wp-content/uploads/2022/10/auto-repair-shop-bays.jpg

##### Bulgaria Sofia City Lulin 2

VehicleParkNumber 3

Central washing center

<https://tommys-express.com/wp-content/uploads/2022/11/hero.jpg>

**Bulgaria Sofia City Iskar Str. 36**

VehicleParkNumber 1

Eastern washing center

<http://ultrasonicexpress.com/wp-content/uploads/2020/12/DSC_4331-scaled.jpg>

**Bulgaria Sofia City Mladost 4**

VehicleParkNumber 2

Western washing center

<https://imocarwash.com/media/5688/wetherill-park.png?anchor=center&mode=crop&width=851&height=381&rnd=133173853890000000>

##### Bulgaria Sofia City Lulin 2

VehicleParkNumber 3

(Eastern – 2, Western - 3)

ToDo: To build buttons functionality!

Users:

[cleaner@mail.com](mailto:cleaner@mail.bg) - Cleaner123

[admin@mail.com](mailto:admin@mail.bg%20) - Admin123

[mechanic@mail.com](mailto:mechanic@mail.bg) - Mechanic123

[manager@mail.com](mailto:manager@mail.com) - Manager123

[boss@mail.bg.com](mailto:boss@mail.bg.com) - Boss123

Failure Description:

“The problems are with the Engine starter. You also need to check the electricity system!”

OrderParts btn -> OrderPartsViewModel -> AddPartsOrder(Action) in RepairCenterController -> AddPartsOrderToVehicle(int vehicleId) (Service) - View with Save btn– With this functionality **all repaired vehicles** will have **information for expenses** , **which will be useful for statistic information. The statistic information needs the Boss! DONE!**

Make the bill -> TheBillViewModel -> MakeTheBill(action) in OfficeController-> MakeTheBillAndSendIt( (int renterd) (Service)– With View with send button. With this functionality the renter will receive the bill like card with “Pay” button in his page “My Bills” **asp-controller**="Office" **asp-action**="MyBills" , view"MyBills" , MyBillsViewModel. In Layout MyBills ancer (The “Pay functionality will build on the Second Stage with “Third Part Autentication”). **DONE!**

**ForgetUser**!!!?

**I need to create:**

It’s good idea to Seed:

users,

vehicles,

vehicle centers,

wasing centers and repair centers

**ЗА СЕГА: DONE!**

Съобщение при наемане в колко часа и къде може да се получи ключа.

**The statistic behind the table: DONE!**

* **Free Vehicles** (behind a **table** with **Vehicles**):
* **Rented Vehicles** ( behind a **table** with **Vehicles**):
* **ThreeMostRepairedVehicles** ( behind a **table** with **Vehicles**):

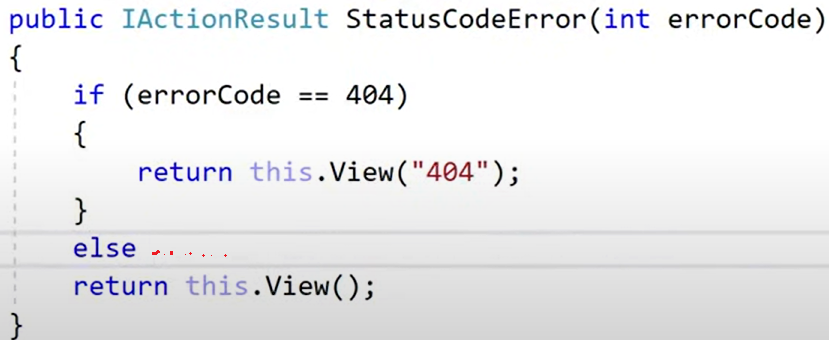
**Последно наети по някакъв начин да стоят в май виакълс!??? НЕ**

**For a second stage (for a Softuni defense) I will add:**

* **Third part authentication**
* **More Statistics with Web API**
* **Забравяне на потребител (GDPR)**
* евентуало чат м/у потребители и мениджър, както и вътрешна комуникация.
* Система за плащане (Google Pay, E-Pay!?).
* Rating will be determine on the base of users vote – With AJAX (Ники ще покаже) – I’ve made it without !?
* Втори език – английски ?
* В случай че го използвам за защита на WebApplications, ще покрия 75% от сървисите с тестове Nunit.
* Telegram -удобно за съобщения!

**Да си оправя качеството на кода:**

* **Да махна от всякъде ViewBag or ViewData**
* **Да сложа коментари и направя Рийдми**
* **Да сложа сайд бар с Топ Рентед Виакълс към Лейаута с помощта на ViewComponent.**
* **Брой регистрирани потребители мога да показвам с ViewComponent!**
* **Momet.js да вкарам**
* И взависимост от **errorCode** да вадим различна страница с различна картинка на потребителя (различни Вюта вадим). Трябва да си направим Екшън, който да обработи всички тези заявки! Така ще стане редирект, който браузъра ще види и обработи. **– Да вкарам в проекта!**



* **За комуникация с потребителя може да добавя SendGrid -** В **SendGrid** сайта трябва да влезнем и да се регистрираме и да си вземем собствен Api key, в Темплейта на Ники няма, защото ако имаше щеше много бързо да е блокиран от SendGrid. В PressCenters.com има интегриран **SendGrid, да разгледаме.**

From YouTube I can **copy** directly **video tag** if I want to use it!

Chart JS за визуализации на графики!?

Да проверя ползвам ли го:

Добра практика е да си слагаме на контролерите **[Authorize],** a където не е нужно **[AlowAnonimus**

**Record your screen**

1. Go to the game or app you want to record.
2. Press the Windows logo key+Alt+R to start recording a clip. ...
3. To toggle the microphone on and off during the recording, press the Windows logo key+Alt+M. To display the Game Bar controls, press the Windows logo key+G. You hear: "Xbox Game Bar window."

**Highes tech option (chat bot) Да проуча има ли безплатен вариант и евентуално да интегрирам:**

### ChatGPT in Preview on Azure OpenAI Service

[**https://visualstudiomagazine.com/articles/2023/03/09/chatgpt-azure-openai.aspx?oly\_enc\_id=2071F1153545J9G**](https://visualstudiomagazine.com/articles/2023/03/09/chatgpt-azure-openai.aspx?oly_enc_id=2071F1153545J9G)

**CodeGPT 3** is extention for **Visual Studio Code** add when install it we have to **link wth API from our profile in** **Playground** (using API keys maby ….10 безплатни) препоръчва да работим с **Playground** профил ползвайки **CodeGPT – по добре е! Платен е реално, когато си създадем акаунт ни дават да ползваме 18 долара в началото, но това не не никак малко за да го разцъкаме и си поиграем с него за да го разучим (1 долар изхарчен за около 400 промпта (запитвания)). Но ползвайки Auto GPT да си знаем, че прави множество заявки – хиляди и бързо ще ни свърши „безплатния“ акаунт.**

**Можем да накараме ChatGPT да ни разкаже за настрийките на моделите в Playground. Да ползваме API -то, да си свържем приложението**

**Auto GPT – best GitHub project**