



SHEREÉ GREEFF

31083625

19 NOVEMBER 2020

CMPG 323

PROJECT 2

Contents

[*Introduction & Overview* 3](#_Toc56111817)

[*ERD* 4](#_Toc56111818)

[*Use cases* 6](#_Toc56111819)

[*Data flow diagram* 8](#_Toc56111820)

[*How to guide* 10](#_Toc56111821)

**No table of figures entries found.**

# *Introduction & Overview*

ASP.NET Core is a framework for building web applications and web APIs using the Model-View-Controller design pattern. User requests are routed to a Controller, which works with the Model to perform actions or retrieve queries. The controller chooses the View to display to the user and provides it with Model data. The Model represents the state of the application and operations that should be performed by it. Views present the content through the user interface. The Controller is responsible for handling user interaction, working with the model and selecting a view to render.

ASP.NET Core MVC Framework is a lightweight, open source, highly testable presentation framework optimized for the use of ASP.NET Core. It provides a pattern-based way to build dynamic websites that enables a clean separation of concerns. It gives you full control over mark-up, supports friendly development and uses the latest web standards.

ASP.NET Core Web API is designed for RESTful (Representational State Transfer) services, which means it provides interoperability between computer systems on the internet.

ASP.NET Core Web Application is designed to develop web applications that return views and data.

I chose the Web Application route because I am more familiar with working in Visual Studio.

I made use of the following software:

* Visual Studio Code

# *ERD*

Entity relationship diagrams I planned for my project:

1…\*

|  |
| --- |
| RATE |
| RATE\_ID |
| HOURLY\_RATE |
| MONTHLY\_RATE |
| OVERTIME |
| INCOME |

|  |
| --- |
| DEPARTMENT |
| DEPARTMENT\_ID(PK) |
| DEPART\_DESCRIPTION |

1 1

1 1

1 1

1 1

1

|  |
| --- |
| **EMPLOYEE** |
| EMP\_ID (PK) |
| NAME |
| AGE |
| IDENTIFICATION\_NR |
| ADDRESS |
| CONTACT\_NR |
| EMP\_NUMBER |
| RATE\_ID (FK) |
| HISTORY\_ID (FK) |
| EDUCATION\_ID (FK) |
| JOB\_ID(FK) |
| EMP\_DETAILS(FK) |

|  |
| --- |
| **EMP\_DETAIL** |
| GENDER\_ID(PK)(FK) |
| MARITAL\_STATUS\_ID(PK)(FK) |
| DEPARTMENT\_ID (PK)(FK) |

|  |
| --- |
| **GENDER** |
| GENDER\_ID(PK) |
| GENDER\_CODE |

1

1

|  |
| --- |
| **EMP\_HISTORY** |
| HISTORY\_ID(PK) |
| YEARS\_WORKED |
| YEARS\_CURRENT\_ROLE |
| LAST\_PROMOTION |
| YEAR\_LAST\_TRAINING |

|  |
| --- |
| **MARITUAL\_STATUS** |
| MARITAL\_STATUS\_ID(PK) |
| MARITAL\_STATUS\_CODE |

1…\*

|  |
| --- |
| **EMP\_EDUCATION** |
| EDUCATION\_ID(PK) |
| FIELD |

1..\*

|  |
| --- |
| **JOB** |
| JOB\_ID(PK) |
| JOB\_ROLE |
| JOB\_LEVEL |
| JOB\_INVOLVEMENT |
| JOB\_SATISFACTION |

ER for authentication:

|  |
| --- |
| **ASP\_USER\_TOKEN** |
|  |
|  |
|  |

|  |
| --- |
| **ASP\_USER\_CLAIM** |
| ID |
|  |

|  |
| --- |
| **ASP\_USER** |
| ID |
|  |

|  |
| --- |
| **ASP\_USER** |
| ID |
|  |

|  |
| --- |
| **ASP\_USER** |
| ID |
|  |

|  |
| --- |
| **EMPLOYEE** |
|  |
|  |

|  |
| --- |
| **ASP\_USER\_ROLE** |
|  |
|  |

|  |
| --- |
| **ASP\_ROLE** |
| ID |
|  |

|  |
| --- |
| **ASP\_USER** |
| ID |
|  |

# *Use cases*

Authentication use case:

User sends credentials to service’s login page

Authentication

Server

Returns access token to user

.NET Core

Console

Client

User browses to web application

WEB

Application

Web app takes user to authentication service

Web app returns response to user

User provides access token

Login use case:

User

CRUD use case:

Web Application - Admin

User - Administrator

Register Use Case:

Login/register

<<Includes>>

User

# *Data flow diagram*

MVC Data flow diagram:

Database

Repository services

Infrastructure

-Logging

-Security

Business rules

Controller

Model

View

System Data flow diagram:

Login Master

Yes

Yes

No

Verification Data

Data and Control flow diagram for request:

Controller

Set

Database

Model

Get

View

Data flow

Control flow

* Controller:
* Event Handling
* Update Application Data
* Definition of Control Flow
* Model:
* Definition of Application Data
* Connection to Business Functionality
* View:
* Visualization of Data

# *How to guide*