MIVeterinary WebApp documentation

In this document you will find all the documentation related with the application: Technologies used, how the folders are structured, and a simple user manual for the use of the application.

## **Technologies used**

To develop this application following technologies has been used:

|  |  |  |
| --- | --- | --- |
| Data storage  Database | Data management  BackEnd | User Interface  FrontEnd |
| MySql (MariaDB 10.4.10) | **.net Core (c#) 3.1** | **Angular 9**  **Bootstrap 4** |

There are also some CDN imported:

* **Google Fonts**. For using the “Roboto” font family.
* **Font-awesome**. For icon usage in the app.

## **Folder structure**

The main .zip file which contains all the necessary files and folders to run the application has following structure:

* **BackEnd**: in this folder you can find all the c# code with the corresponding Visual Studio project.
* **Database**: Contains the .sql files with the creation of the database code and some mock data for the tables.
* **Documentation**: All the docs about the application like this one.
* **FrontEnd**: Complete front-end project as npm package. Easy to run with “npm I & npm start” command.
* .**gitignore**: Git file to ignore files & folders for github.

## **Startup of the application**

Prerequisites:

* You must have the latest node,js installed (with npm manager). Also, Angular-cli is required. This project has been developed with latest Angular 10 version.
* MySql Database is required it can be on local or in a server.
* Visual Studio to start backend.

This application is in develop mode, its not deployed.

To start the application, we need to follow the following steps:

1. **Create the database.**

In the “Database” folder you can find all the sql files needed for this step. There are two options:

* 1. Import the complete dump file (“*00. Complete DUMP database for import*”) inside your mysql server. This file is tested by importing it in the phpMyAdmin panel.
  2. Use the manual 4 scripts in order to create the db, create the tables & mandatory data, add the procedures and use the fourth as optional to add mock data.

1. **Run the Backend.**

This step is quite simple. Just open the backend, update the connection info in the “*appsettings.json*” file and start it.

Once backend is working, we can start with the next and last step.

1. **Run the Frontend.**

Go inside the front-end folder and open this location in the windows command line. Run following command:

|  |
| --- |
| npm install |

Once done, just run this one to start the web app:

|  |
| --- |
| npm start |

This will automatically open the application on your default browser.

## **Features to implement to the app**

This is the first version of the app. But there are some features to implement in the next version. Here is a basic list of the pending task.

* Implement better error handling in the API
* Implement 100% responsiveness of the homepage
* Multiple edit of items in the tables