**People’ Bank**

**Project Overview:**

This application is an online banking platform that will be developed using C# with ASP.NET Core MVC and Entity Framework Core for data access. It will have a backend logic including user authentication, registration, and dashboard functionalities. Additionally, we'll use SQLite as the database.

A regional bank called People’s bank has opened a branch in Louisville, KY and its management has contracted Capstone LLC to develop an online application to cater to the needs of their customers.

Capstone LLC offered to develop an online application called People’s online bank. People’s online bank will include the following major features:

* Login
* Registration
* View Balances
* Transfer Funds
* Make deposits
* Link Accounts
* View Financial Statements

**Requirements**

Requirement Elicitation

* How will Capstone LLC elicit the requirements?
* Stakeholder Meetings
* Informal Interviews
* Requirement Surveys
* What does Capstone LLC do with the elicited requirements?
* Brainstorm
* Storyboard
* Backlog
* Begin the processes of designing and developing the product

**Technical Insight**

**Tools:** Visual Studio 2022, SQLite, Entity Framework Core.

In order to develop the People’s online bank application, Capstone LLC chose the MVC-Model from among other software models. Also, employed are ASP.Net Core, Entity Framework core, Identity UI – User Registration and Login, ORM and DbContext.

Here are the steps to build this application:

Setting up the development environment: This involves installing Visual Studio 2022 and setting up SQLite for our server-side.

**Step 1: Create ASP.NET Core MVC Project**

Open Visual Studio and create a new ASP.NET Core Web Application project:

* Select **ASP.NET Core Web App (Model-View-Controller)** template.
* Choose a project name and location.
* Select **ASP.NET Core 6.0** or the latest version available.

**Step 2: Set up Entity Framework Core**

Entity Framework Core (EF Core) will be used to interact with the SQL Server database (SQLite).

1. Install the necessary EF Core packages:
   * Microsoft.EntityFrameworkCore.SqLite
   * Microsoft.EntityFrameworkCore.Tools

Install these packages via the NuGet Package Manager in Visual Studio or using the Package Manager Console:

1. Define your database context and entity classes:
   * **Data/AppDbContext.cs** (Database context)

**Step 3: Implement Authentication and Registration**

**Step 4: Implement Dashboard**

**Step 5: Configure Startup and DbContext**

**Step 6: Configure Connection String -** Update appsettings.json with your SQLite connection string:

**Step 7:** Tests (unit and integration) and debug application thoroughly, especially when dealing with sensitive data like financial information We will do this using a combination of manual and automated tests.