

Wex Senior Software Engineer Assessment

By: Ricardo de Araujo Goes de Almeida

Date: December, 12, 2025

Objectives

Build an application capable of creating, editing and deleting purchase transactions, and retrieving them converted to currencies supported by the Treasury Reporting Rates of Exchange API, using the exchange rate active on the transaction date.

Architectural and Implementing Decisions

- The application is decoupled into a frontend (user interface) and a backend (storage and management services).
- The backend is developed in C#, using DOTNET Core 9 and Entity Framework 9.
- The database used is MariaDB (MySQL-compatible).
- The frontend is developed in Typescript, using the Angular 20 framework
- The web server and API Gateway used is NGINX
- The solution runs entirely within three Docker containers, using a private network, requiring no external database or web server installation, satisfying the requirement for a self-contained environment.
- Best practices were followed in the application design, especially regarding code organization and security.
- The application requires authentication (login/password). To simplify this assessment, a single static login/password was used to demonstrate functionality.
- The development of more complete functional automated tests was planned; however, during implementation, I prioritized completing all core functional requirements and the supporting execution infrastructure (containers, database, and API gateway). Given the limited time available, I chose to deliver a solid and well-structured foundation—ensuring proper layer separation, isolated services, and clearly defined test points—while leaving the full implementation of automated tests for a subsequent stage. I have included some basic functional tests in the frontend (Karma/Jasmine) and on the backend (xUnit + FluentAssertions + test doubles via Moq) just to demonstrate my knowledge of this area.

Distribution

Source code for the entire project is available on GitHub (<https://github.com/ricardag/wex-assessment>).

Deployment

Requirements

1. .NET 9 SDK to build the backend
2. Latest Angular CLI (20.1.6 or later) to build the frontend

Proceeding

A bash script (build-and-deploy.sh) is provided at the project root. It performs the entire deployment process: builds the application, prepares the Docker environment, creates the containers, initializes the database, and configures the NGINX gateway.

Before deployment, create a '.env' file at the project root. This file contains secrets and other sensitive parameters that are not committed to Git. Use the '.env.example' file as a template.

After configuring the '.env' file, run the "build-and-deploy.sh" script. It will create and start the containers, initialize the database, and make the application available..

You can access the application with your browser at <http://localhost:8100>. "Use the credentials specified in AUTH_USERNAME and AUTH_PASSWORD in your .env file.

Contact me if you encounter any issues during deployment or execution.