

FORCEGET SOFTWARE

Senior Full Stack Developer Case – Version 202512-2

Task:

Develop a full-stack web application that collects and displays quote requests from users.

Objective:

Create a system that supports **Login / Register**, retrieves **Country & City** data from an external API, converts **Currencies to USD**, and saves form results to the database to be displayed in a list.

Tech Stack Requirements

Backend

- **.NET 9 Web API**
- **N-tier Architecture (Domain → Application → Infrastructure → API)**
- **PostgreSQL** as the database
- External API Integration:
- **API Ninjas City API** (Country & City)
- **API Ninjas Currency Conversion API**

Frontend

- **Angular 19**
- **NG Zorro** UI Library
- Reactive Forms
- Interceptor for JWT tokens

Security

- JWT-based authentication
- Password hashing
- Secure API communication (HTTPS)

Pages to Develop

1. Register & Login Pages

Register Page

- Email
- Password
- Confirm Password

Login Page

- Email
- Password

After login, the user receives a JWT token and can access the Quote Form page.

2. Quote Request Form Page

A form where applicants submit shipping details.

Fields

General Fields

- **Mode:** LCL, FCL, Air
- **Movement Type:**
 - Door to Door
 - Port to Door
 - Door to Port
 - Port to Port

- **Incoterms:**

- DDP

- DAT

- **Package Type:**

- Pallets

- Boxes

- Cartons

- **Currency:**

- USD

- CNY

- TRY

Country / City Inputs

Country

- Manually typed input (user types “Turkey”, “USA” etc.)

City

- Must be dynamically fetched from **API Ninjas City API**

- Autocomplete control

- Should show all city names matching the typed country

Example flow:

1. User types "Turkey"

2. Frontend calls

3. Autocomplete shows: Istanbul, Ankara, Izmir, Bursa ...

Currency Conversion

- User enters an amount (e.g., 150 TRY)
- Backend calls

Currency Conversion

- User enters an amount (e.g., 150 TRY)
- Returned USD equivalent should be displayed in the form
- Both values must be stored in the database:
- Original currency amount
- Converted USD amount

3. Quote List Page

A datagrid displaying all submitted quote requests.

Table Columns:

- ID
- User Email
- Country
- City
- Mode
- Movement Type
- Incoterms
- Package Type
- Original Currency & Amount
- Converted USD Amount
- Created Date

Backend Tasks

✓ Implement Authentication

- Register
- Login
- Issue JWT token

✓ API Ninjas Integration Service

ICityService → Fetch cities
ICurrencyService → Convert to USD

✓ Quote CRUD API

- POST /api/quotes
- GET /api/quotes

Frontend Tasks

✓ Login / Register Pages

Store JWT in LocalStorage
Use interceptor to attach token

✓ Quote Form

- Reactive form
- Autocomplete for cities
- Show USD conversion live
- Submit to backend

✓ Quote List Page

- NG Zorro Table
- Fetch & display quotes