Home assignment:

Dear candidate, thank you for your time in taking this home assignment as part of your evaluation.

The purpose of this exam is to assess your coding and design abilities, emphasis on scalability and knowledge in OOD and OOP principles rather than a fully working API.

The assignment should take about 2-3 hours.

**Instructions**

**Server Side**:

In the following exercise you will build a simple web API system.

The purpose of the system is to offer each client a specific loan interests based on the following scenarios:

1. If the client is below the age of 20 the interest of the loan should be 2% + prime interest.
2. If the client is between 20-35, use the following division to calculate the loan:
   1. 5,000 nis and below 2% fixed
   2. 5,000 – 30,000 nis 1.5% + prime
   3. Above 30,000 nis 1% + prime
3. If the client is above 35, use the following division:
   1. 15,000 nis and below 1.5% + prime
   2. 5,000 – 30,000 nis 3% + prime
   3. Above 30,000 nis 1% fixed

The minimum loan period is 12 months, you are also required to add 0.15% for each extra month

above the minimum period.

For example, a client at the age of 21 requests a loan of 4,500 nis for a period of 13 months.

His basic loan interest is 2% = 90 nis and for the extra month 1\*0.15 % = 6.75 nis, so the total amount is 4,596.75 nis.

The system will contain at least one user for each scenario, and an endpoint that receives the client id, amount in nis and period in month. The output should be the total amount at the end of the period. If you like and have the time you can add the amount calculated divided by each calculation depending on the scenario.

**Client side**:

Create a single page project using React\Angular (4+)\Vue.  
The page will contain input fields for user id, loan amount, loan period and a button for loan suggestion.  
Once the user clicked on the button, it will call the endpoint you developed on server side in the previous step and display the loan details in a popup or inside a styled div.  
Field Validation is required.

Guidelines:

1. Please use proper design patterns where they are needed (Repository, Strategy, Factory)
2. You can use Json or xml as your data source (you will not be tested on this)
3. The project must be C# .net Web API preferably .net core latest.
4. Make sure you write clean and scalable code.
5. You don’t have to use async, but it is recommended.
6. Prime interest for this exercise is 1.5%
7. Make sure client-side code is clean and scalable.

Good luck!