

TAKE HOME ASSESSMENT

Please make sure you read the entire document before you commence the assessment.

Create a **Web API application** that models a wallet solution.

Your **APIs** should have support for **Authentication and Authorization**.

We are interested in your architecture, code quality and design as well as your ability to fulfill functional requirements.

Requirements

USER

1. As a user I should be able to sign up for an account with my email, first name, last name and password. Required validations should be done on the fields.
 - A user can also upload a profile picture at sign up and be able to add it if it wasn't uploaded at sign-up.
2. As a user when I sign up, a USD, GBP, and NGN wallet should be created for me automatically. Handle scenarios where there is a possibility of failed wallet creation.
3. As a user I can also add more currency type to my wallet.
4. As a user I should be able to fund my wallet with any amount.
5. As a user I should be able to withdraw from my wallet.
6. As a user I should be able to view my wallet balances and transactions.
 - Provide functionality to filter transactions by day, week, month, year.

ADMIN

1. As an admin user I should be able to view all users, activate or deactivate a user.
 - I should be able to view a user, their wallet balance and their transactions.
2. As an admin user I should be able to view all transactions of the platform.
 - Provide functionality to filter by day, week, month, year.
3. As a super admin user; you can seed a default super admin user, I should be able to create new admin users and assign role and permissions to them.
4. As an admin user I should be able to create and update currency type and attach their respective currency logo.
5. As an admin user I should be able to export all transactions of the platform.

There are now over a million records in our wallet database and we've decided to start giving 3.75% per annum interests (Simple Interest) on the wallet balances.

Using the best approach you know, write a service that runs every 12 midnight to calculate the earned interest and apply the resultant amount to the wallet based on the balance of the current amount in the wallets. The admins should also be notified when the process is completed.

Guidelines

Use Git for version control.

Share a link to the Github repository hosting your solution.

Extra points for any form of automated test (i.e unit etc).

For the Files; the profile picture and currency logo, validations must be done on the file size and types, you can also use any file storage service you know i.e. FTP, Cloudinary etc.

For the Notification: it can be in app, real time, push notification, email etc.

Tech stacks to be used :

- *C# / .Net Core >= 3.1*
- *Ms Sql Server*
- *Ef Core or Dapper*