Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 3 з дисципліни

«Сучасні технології розробки WEB-застосувань на платформі Microsoft.NET»

 Виконав студент        ІП-15, Мочалов Дмитро

(шифр, прізвище, ім'я, по батькові)

 Перевірив                  Бардін Владислав

(прізвище, ім'я, по батькові)

 Київ 2023

Завдання:

1. З дотриманням вимог REST-у спроектувати веб-API для

обраної(згідно варіанту) доменної області, використовуючи

методологію C4 для створення діаграми архітектури системи.

2. Створити ER-діаграму для DAL (Data Access Layer), яка

відображатиме структуру бази даних веб-API.

3. Оформити спроєктоване рішення у вигляді звіту до лабораторної

робот

A white paper with black text

Description automatically generated

Виконання:

## C4 Diagram :

Level 1. Context

A diagram of a person

Description automatically generated

В нашій системі буде користувач, який буде використовувати систему для керування своїм бюджетом.

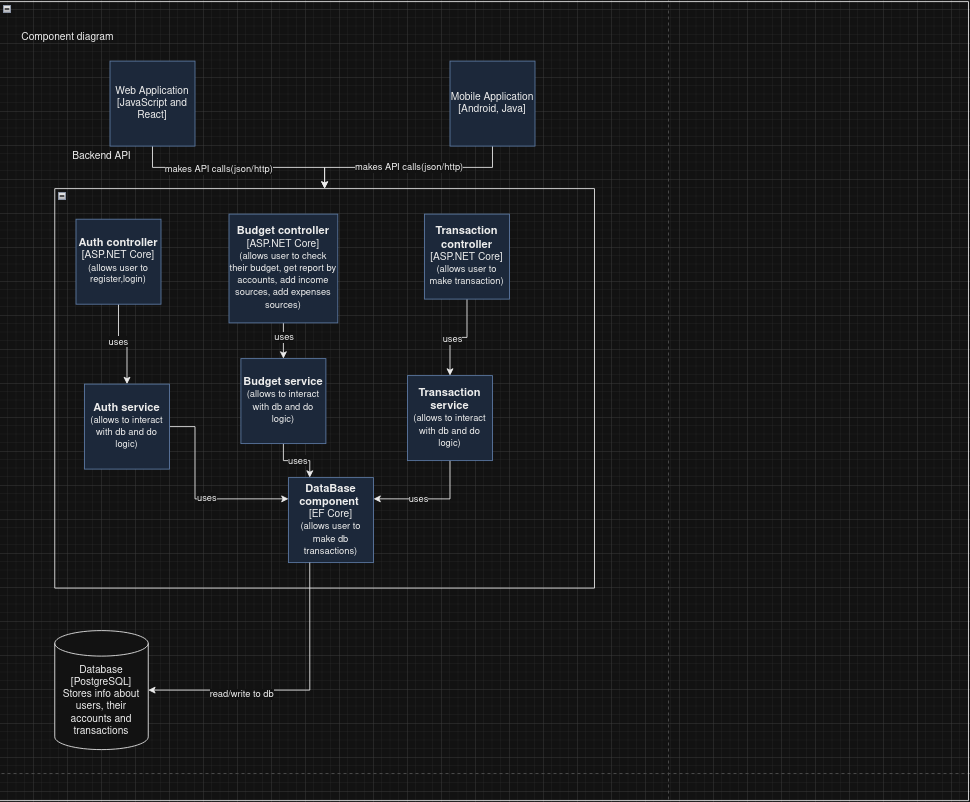
Level 2. Container

A diagram of a user

Description automatically generated

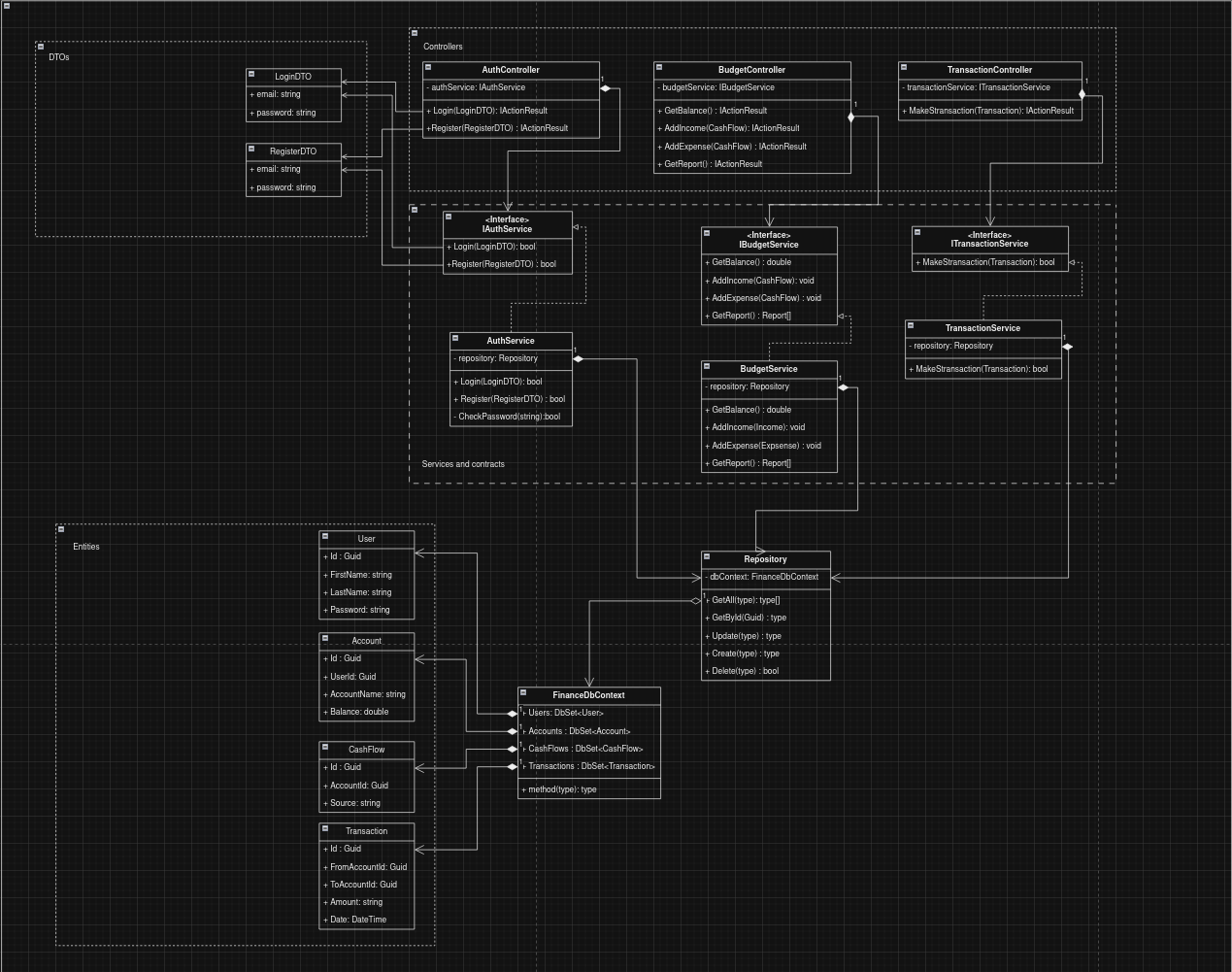
В нашій системі буде 4 контейнери, це веб та мобільний застосунок з яким буде взаємодіяти користувач, ці застосунки будуть звертатись до апі, яке буде повертати йому данні, та виконувати дії на даними в базі, та сама база даних, яка буде зберігати ці всі дані.

Level 3. Component



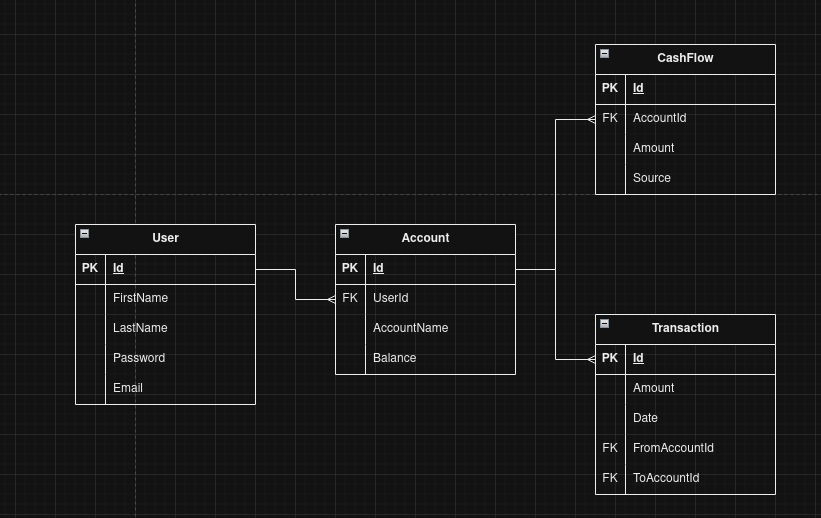
На третьому рівні ми бачимо наші компоненти з технологіями які будуть для них використовуватись.   
В нашому випадку це будуть 3 контролери які будуть відповідати за авторизацію користувача, проведення транзакцій та управління бюджетом користувача.   
Все це буде викликатись з SPA веб застосунку або андроїд додатку.  
Всі нові та оновлені дані будуть записуватись в базу даних на PostgreSQL.

Level 4. Code



На четвертому рівні наша код діаграма, яка зображена у вигляді UML діаграми класів. Тут показані сутності які будуть в базі, контроллери, сервіси, контракти.

ER diagram:



Таблиця User

* Id – user identifiers
* FirstName – user first name
* LastName – user last name
* Email – user email
* Password – user password

Таблиця Account

* Id – account identifier
* UserId - foreign key to linked user
* AccountName – name/number of account
* Balance – account balance

Таблиця CashFlow

* Id – expense identifier
* Amount – expense/income amount
* Source – expense/income source

Таблиця Transaction

* Id – transaction identifier
* Amount - transaction amount
* Date – transaction date
* FromAccountId – foreign id to account from transaction is sent
* ToAccountId – foreign id to account destination transaction

API enpoints:

AuthController:

* POST /auth/login – authorize user

{

email: string;

password: string;

}

* POST /auth/register – register user

{

email: string;

password: string;

}

TransactionController

* POST /transaction/addTransaction – making transaction between two accounts

{

amount: double;

Date: Date;

FromAccountId : string;

ToAccountId: string;

}

BudgetController

* GET /budget/getBalance – get user balance
* POST /budget/addIncome – add income source

{

accountId: string

amount: double;

source: string;

}

* POST /budget/addExpense – add expense category

{

accountId: string

amount: double;

source: string;

}

* GET /budget/getReport – get report for account