EPF Income and Expense Project

# Overview

The project is a simple web application that provides a form for the user to enter a customer’s name, last name, date of birth and an excel sheet containing their income and expenses for the last 12 months. After submitting the form, the application will then take the information and render a temporal graph based on the customers income and expenses from the graph.

# Tech stacks

## Programming languages

* Java: The main programming language for the domain model.
* HTML: The structure of the website
* CSS: Styling of the HTML sites

## Frameworks

* Javalin: Framework used to build the Web API.
* Thymeleaf: Used for HTML manipulation.

## Explained

Java is the main programming language used. This project is a Java project. The are 2 HTML files, namely index.html – which contains the form – and table.html which will draw the temporal graph based on the details of the form from the previous file. There’s a styles.css that is used to style both the index.html and the table.html

Javalin is the framework that is used to build the Web API, it has an embedded servlet container which simplifies running our API. It works with the java code to create endpoints and routes for requests coming from the HTML forms. Thymeleaf is the template engine used to implement the BudgetServer app. The engine is used to manipulate table.html to have the data coming from index.html. This is used to make table.html a dynamic website.

## Architecture

This project uses a layered architecture containing the domain layer, the web API layer and the presentation layer. The domain layer is mainly responsible for the Record object. The web API layer is responsible for handling form data, endpoints, routes and requests from the client. The presentation layer is responsible for the visual design and layout of the user interface of the program.

## Demo

In the directory of the project, there is a demo of the application named demo.mp4. This demo uses the linux bash terminal to execute the program using maven and then google chrome as the browser used for the user interface.