

Budget Tracker

[Demo video](#)

[Github Repo](#)

1.The list of initial project requirements

Completed

BT-1: As a user, I want to create an account

BT-3: As a student, I want to be able to record all my expenses so that I can save more money to go out with my friends.

BT-4: As a young adult, I want to limit spending money in some areas in order to increase monthly savings.

BT-5: As a mom, I want to add my categories in order to group my expenses and incomes.

BT-6: As a freelancer, I want to monitor my incomes from different sources in order to see the trends in my total income.

BT-7: As a student, I want to have a group in order to easier split bills with my roommates.

BT-8: As a trip lover, I want to go on many trips without spending too much time splitting expenses with my travel buddies.

BT-12: As a user, I want to login into my account so that I can view my transactions

BT-13: As a user, I want to have a mobile UI to register a new account

BT-14: As a mom, I want a mobile UI to manage my personal categories

BT-15 As a student, I want a mobile UI to register my expenses and incomes

BT-10: As a father, I want to see our household's expenses and how are they shared between the members.

Uncompleted

BT-9: As a user, I want to import my expenses from other banking or financial apps.

- Not enough time and some technical issues
- No. We discussed it and decided that we need to have other more important and basic features in our app before adding this.

BT-2: As a user, I want to review my expenses and incomes

- We had some technical issues with charts libraries
- Yes. We will have more time to implement it and deliver it.
- We will have an architectural/technical analysis before starting implementing, this will help us to overcome technical issues.

2. Team description

- Mura Victor (Business Analyst, Front-end developer, Software Architect, QA)
- Talpalariu Iulia (Business Analyst, Backend developer, Product owner, Software Architect, QA)
- Sabau Eduard (UI/UX, Business Analyst, Backend developer, QA)
- Rogoza Raluca (Backend developer, BA)

NEW MEMBERS:

- Clem Daria (UI/UX, DevOps, Front-end developer)
- Maftai Valentin (Business Analyst, Front-end developer, QA)

3. Software Architecture Report

a)

- Flutter framework - it's a framework that provides a cross platform solution, with one codebase we have Android, iOS and Web applications.
- Dart - is object-oriented, class-based, garbage-collected language used by Flutter to develop applications.
- Spring Boot framework - web framework that we all wanted to gain experience with
- Java programming Language - language used by Spring Boot

b)

- BLoC - is a State Management Tool that helps us with state management and state lifecycle, this is a very important part for an responsive and interactive UI.
- Domain Driven Design - is a major software design approach of developing, this helps us to easily define and understand what are the entities and how they interact between each other.
- Model-View-Controller architecture - clear separation into logical components, makes it easy to extend and reduces complexity.

c)

- Linting - Yes, we used linter for better code quality and code standards enforcing. Linting was executed on every save of the file so it succeeded.
- Clean Architecture as Code Structure and Dependency management - this helps us to have an easy to maintain and easy to upgrade code base.
- SOLID - We followed the SOLID principles

d)

- Some issues with MaxFlow Algorithm, it doesn't function correctly in some corner case situations.

e)

- Some small code refactoring and clean up will be needed on both Frontend and Backend codebases