



Market Tracker

Application For Tracking Variation of Prices Across Different Stores

André Graça, n.º 47224, e-mail: a47224@alunos.isel.pt, tel.: 929130440
Daniel Caseiro, n.º 46052, e-mail: a46052@alunos.isel.pt, tel.: 911124858
Diogo Santos, n.º 48459, e-mail: a48459@alunos.isel.pt, tel.: 939046442

Supervisor: Filipe Freitas, e-mail: filipe.freitas@isel.pt

28 of February of 2024

1 Introduction

This project aims to provide users with a comprehensive web service for comparing supermarket prices, promotions, and product insights. Our platform includes a rating system, advanced filters, and a price and promotion tracking feature, empowering users to make informed purchasing decisions.

2 System Requirements

2.1 Functional Requirements

- Users can authenticate in our application Google's OAuth 2.0 or by creating an account.
- Users can check the graph of variation of price of a product during a period.
- The system was designed for 2 types of users:
 - Partners:
 - Insert and update products, only the ones that belong to them.
 - Insert their stores in our DB.
 - Insert promotions across various products in their stores.
 - Clients:
 - Fill a Cart with products.
 - Rate and comment products.
 - Compare and search by filtering products by various keywords (e.g. brand, store, overall rating, etc...).
 - After completing the cart then can proceed to generate a digital grocery list which will tell how much they will spend in each store and how much stores they need to go to fulfill the cart. They can also generate the lower total price of a basket.

2.2 Non-Functional Requirements

- Software testing to increase our confidence in the system's security and features. It should also allow us to move faster without fear of breaking the previous working code.
- Overall good software practices that should allow the system to grow in complexity more easily.
- Deploying the application components and making them available to the public.

2.3 Optional Features

- Map application to tell the smallest route between each store.
- Android application.

3 Technologies

We plan to use C# [1] alongside it's Entity [2] and MVC framework [3] to serve application backend service. The database we will use a persistently store application like PostgreSQL [4]. The web application we will use Html [5] and TypeScript [6] for web pages, as well NextJS [7] (typescript framework for frontend development). If we can develop a mobile application, it will be developed in Jetpack Compose [8].

4 Risks

Developing the backend service can take longer than usual thanks to our inexperience with C# programing language as well its frameworks (Entity framework and MVC framework) and the same can happen with in the frontend when experimenting with the NextJS framework.

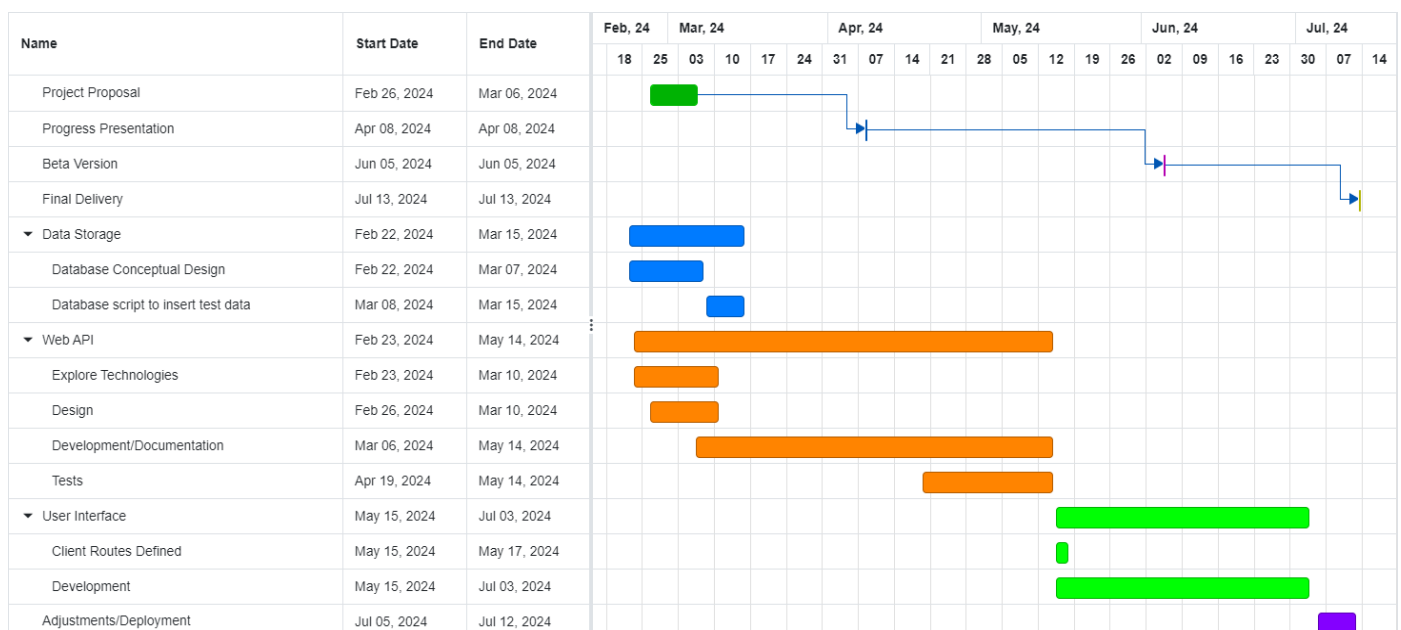


Figura 1: Gantt Chart Project Plan

References

- [1] <https://learn.microsoft.com/en-us/dotnet/csharp/> - C#, a programming language on the .NET platform, last accessed on 29 February 2024.
- [2] <https://learn.microsoft.com/en-us/ef/> - C#'s Entity Framework, the framework for a high-level data access, last accessed on 29 February 2024.
- [3] <https://learn.microsoft.com/en-us/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview/> - C#'s MVC Framework, the framework for architectural pattern base application, last accessed on 28 February 2024.
- [4] <https://www.postgresql.org/> - PostgreSQL, a persistent store application, last accessed on 28 February 2024.
- [5] <https://developer.mozilla.org/en-US/docs/Web/HTML/> - HTML, HyperText Markup Language, last accessed on 29 February 2024.
- [6] <https://www.typescriptlang.org/> - TypeScript, javascript with syntax for types, last accessed on 29 February 2024.
- [7] <https://nextjs.org/> - NextJS, the react framework for the web, last accessed on 29 February 2024.
- [8] <https://developer.android.com/jetpack/compose/> - Jetpack Compose, a toolkit for building native UI, last accessed on 29 February 2023.