SDLE - Planned Design

Problem

This project aims to build a shopping list application for a large group of users. It must offer offline access, real-time collaboration, and data synchronization between users, presenting challenges in concurrency, data consistency, scalability, and high availability.

Key Challenges:

1. Local-First Shopping Lists: Users need to create and manage lists on their devices, with offline functionality.
2. Collaborative Lists: Users can share lists via unique IDs for concurrent editing by multiple users.
3. Concurrency and Consistency: Handling conflicts when multiple users edit a list is crucial. Begin with Last-Writer-Wins and explore CRDTs.
4. High Availability and Scalability: Ensure the architecture can scale for a large group of users without data bottlenecks.
5. Cloud-Based Data Management: The cloud component provides data storage and synchronization for backup and sharing.

Architecture

A key element of the project is the architecture of the local-first shopping list application, which establishes the general system architecture and the ways in which the local and cloud components interact to offer a seamless user experience.

Local Components [1]:

The application's local components, which are installed on user's devices:

- User Interface (UI)
- Local Database
- Concurrency management is required to manage possible simultaneous updates of shopping lists by many users. To find the order of actions and settle disputes, the application first uses local clocks and the "Last-Writer-Wins" technique.

Cloud Components [2]:

The cloud's components consist of:

- Cloud Storage
- Data Replication
- Data Sharding

Technologies

Frontend - React

- Description: React is a JavaScript library for building user interfaces. It allows us to create reusable UI components, making it ideal for building the frontend of our shopping list app.
- Key Features:
    - Component-based architecture

    - Virtual DOM for efficient updates
    - Rich ecosystem of libraries and tools

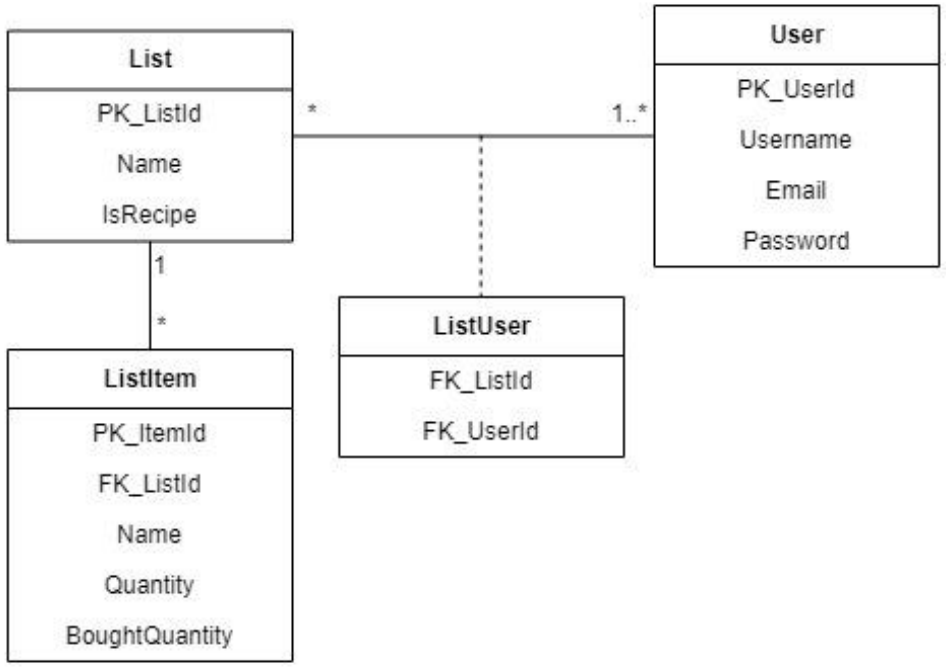
Backend - Python

- Description: Python is a versatile and easy-to-read programming language. It will be the backbone of our backend, handling server-side logic and communication with the database.
- Key Features:
    - Clean and readable syntax
    - Extensive standard library
    - Widely used in web development

Database - SQL

- Description: SQL is a domain-specific language for managing and manipulating relational databases. It will be crucial for storing and retrieving data related to user accounts, shopping lists, and more.
- Key Features:
    - Declarative syntax for querying databases
    - ACID properties (Atomicity, Consistency, Isolation, Durability)
    - Support for relational data modeling

Database diagram

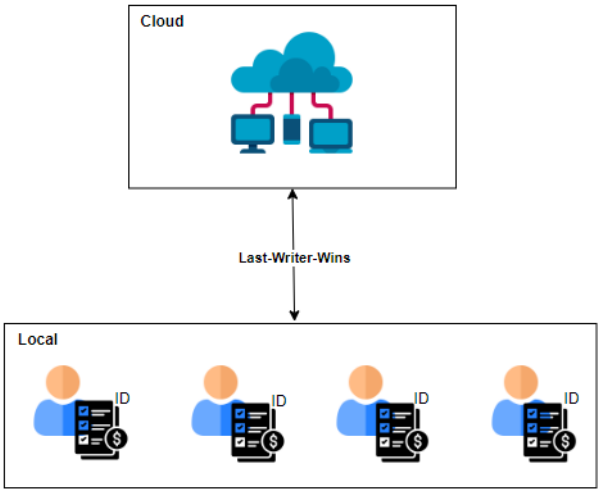


Features

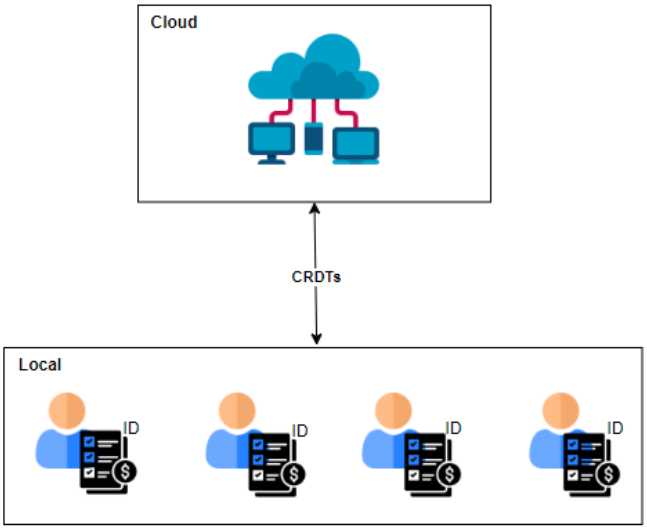
- Create account and access profile
- Login with email and password
- Create new shopping list
- Add items to shopping list
- Delete items from shopping list
- Mark items as bought

Annexes

[1]



[2]



Trabalho realizado por:

- Alexandre Guimaraes Gomes Correia (up202007042@fe.up.pt)
- Eduardo Duarte (up202004999@fe.up.pt)
- João Tomás Marques Félix (up202008867@fe.up.pt)