Shopping Lists on the Cloud

Large Scale Distributed Systems

Deadline: Friday, 13th December

This project will explore the creation of a local-first shopping list application. The application has code that runs in the user device and can persist data locally, and also has a cloud component to share data among users and provide backup storage.

Users can create a new shopping list via the user interface. After creation and until a list is deleted, it exists under a unique ID (e.g., a URL) that can be shared with other users. Users who know that ID should always be allowed to add and delete items to the list. Each item can be associated with a flag, checked after the item is acquired, or with a target quantity that decreases as items are acquired. Since users can concurrently change the list and we aim for high availability, you can first use Last-Writer-Wins with local clocks and later evolve to use Conflict-free Replicated Data Types (CRDTs).

Aiming for millions of users, you should carefully design the cloud-side architecture to avoid data access bottlenecks. Notice that each list is independent and that data can be sharded. It is suggested that you look at the architecture in the Amazon Dynamo paper.

- Choice of language is free. For messaging, you must use ZMQ. You can use also libraries/frameworks not part of the core of the language you choose for message serialization, and for persistence, e.g., a local database. Your application should explicitly handle the replication of the shopping lists, including the consistency of the replicas. E.g., if you use CRDTs, the CRDT implementation must be yours.

- Groups should have three/four participants;

- Groups must make the code of the project available to the staff via the Gitlab project you will be assigned. This Git project must be used also for submitting deliverables;

- Groups must submit a slide deck with the main design challenges and choices, as well as a `README` file with instructions for building and running all code, by the project's submission deadline.

- Before presentation day, every student must fill out the self- and peer-evaluation form that will be made available closer towards the submission deadline.

- On presentation day, each group will do a 10-minute presentation and show a pre-recorded demo of up to 5-minute to the class. This will be followed by a discussion.

- Except for libraries or frameworks you use, all code should be authored by only the group members.

The following papers/links can be useful as starting points:

- ZeroMQ: `https://zeromq.org/`

- Local-first: `https://www.inkandswitch.com/local-first/`

- CRDTs: `https://crdt.tech/papers.html`

- Dynamo: `https://www.allthingsdistributed.com/files/amazon-dynamo-sosp2007.pdf`

To motivate you not to leave the project to the last minute, we will use the class participation component, worth 10% of the course grade, to reward evidence presented in the lab classes of steady progress towards the deliverables.

Furthermore, rembember that late submissions will be penalized 15% per day.