DISTRIBUTED TIMELINE

Large Scale Distributed Systems
2021/2022

Class 7, Group 15:
Ana Cruz | up201806460@up.pt
André Nascimento | up201806461@up.pt
Gonçalo Teixeira | up201806562@up.pt
Gonçalo Pereira | up201705971@up.pt

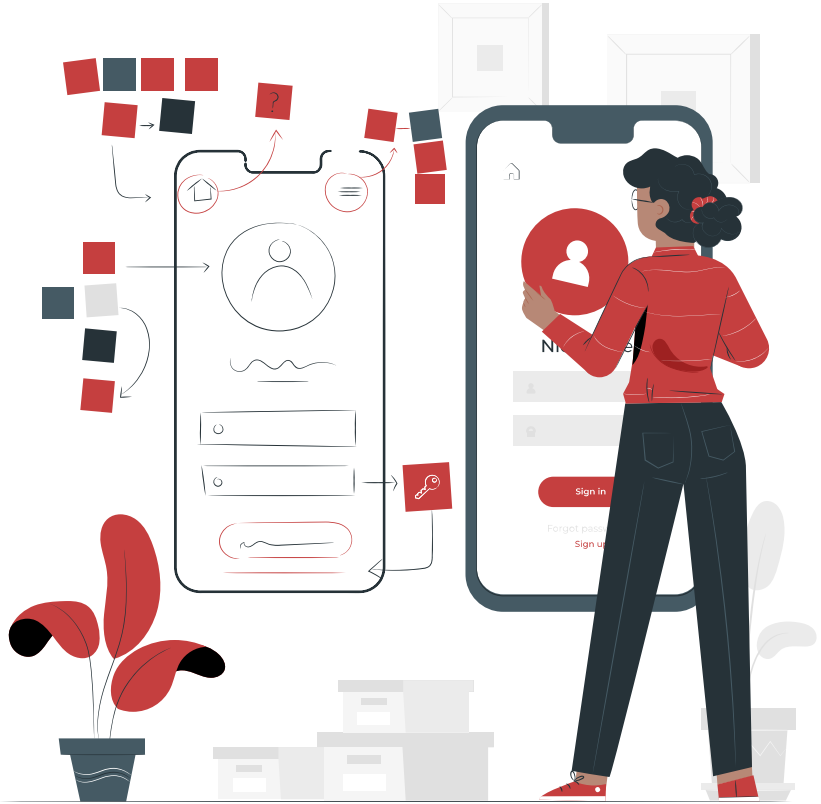
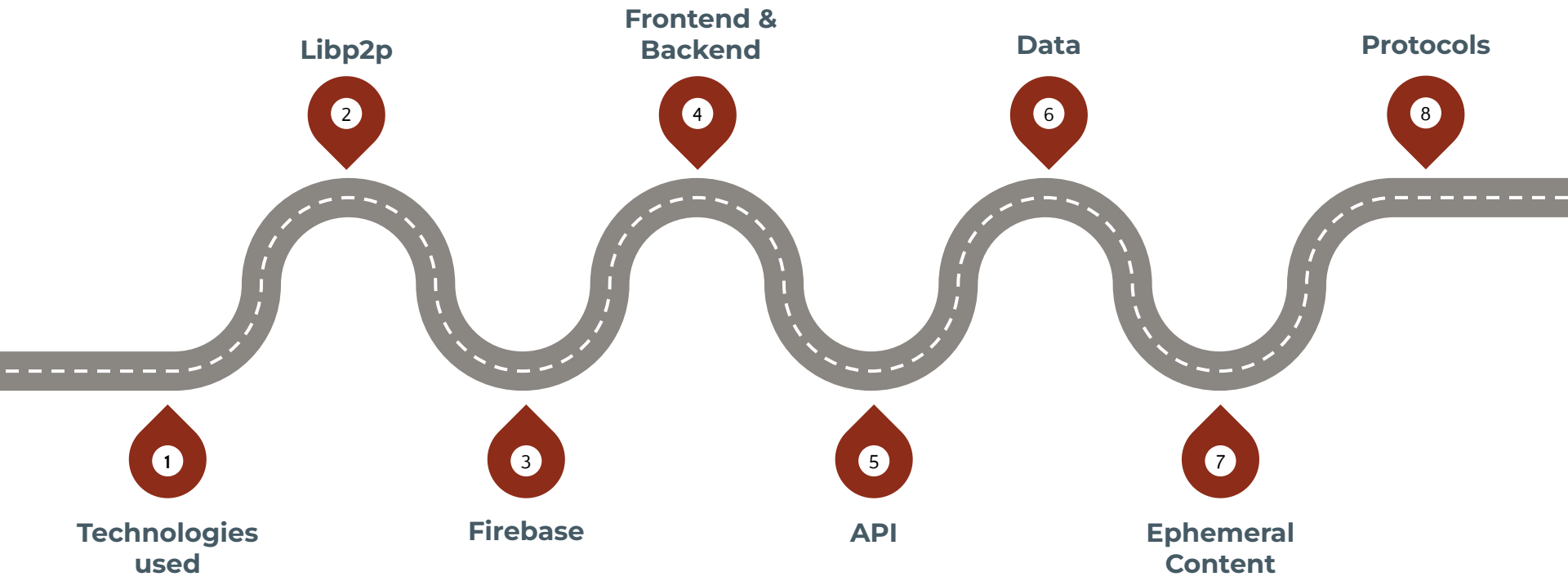


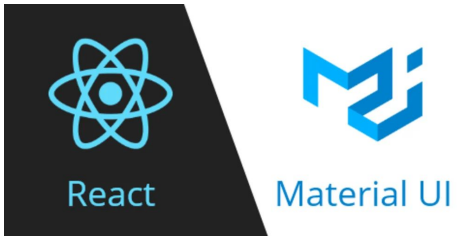
Table of Contents



Technologies used

A network framework that allows you to write decentralized peer-to-peer applications

A development platform known for its realtime database and authentication



A JavaScript library for building user interfaces, with a UI library for interactive components

A minimal and flexible Node.js web application framework that provides a robust set of backend features

Libp2p (1/2)

- **Transport**

  Encrypted: Connections must be end-to-end encrypted.
  Authenticated: The endpoints, Remote Peer and Local Peer, must be authenticated.
  Multiplexed: It must be possible to multiplex multiple reliable streams over a single transport connection.

- **Peer/Content Routing**

  Kademlia: Distributed Hash Table for content and peer routing tasks

- **Publish/Subscribe**

  Gossip PubSub: It is named after the fact that peers gossip to each other about which messages they have seen and use this information to maintain a message delivery network.

Libp2p (2/2)

- **Peer Discovery**

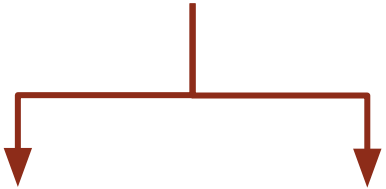
Bootstrap: bootstrap is the easiest way to allow nodes to join a network, we have several bootstrap nodes which accept connections from the application nodes and then allow them to join the network and find other existing and upcoming nodes.

Multicast DNS: since our bootstrap nodes are on the same local area network this discovery mechanism was configured so each bootstrap is aware of every other bootstrap node.

Firebase

Authentication

The sign in method is email/password



Login

email
password

Register

email
username
password

Firestore

The database to store the users' identification

Peer ID
username

Frontend

Axios

Backend

Interactive and responsive UI

- Login/Sign up
- User feed
- Side Panel with all users and search bar
- Users profiles
- Dev Panel

Requests to libp2p

API to handle user requests and call the respective protocols

API

/start	Start the node
/signup	
/logout	
/info	GET method to retrieve all nodes discovered (dev option)
/username/:peerid	GET to retrieve username for peedid
/posts	POST method to insert a new post
/record	GET method for own record
/subscribe	POST method to subscribe a user
/unsubscribe	POST method to unsubscribe a user
/subscribers	GET method to retrieve all followers
/subscribed	GET method to retrieve all following users
/providers/:username	GET method to retrieve providers for username record
/profiles/:username	GET method to retrieve username record
/feed	GET method to fetch the feed for the current user

Data

Record format


```
{
  "posts": [
    {
      "data": "everyone is offline",
      "author": "skdgt",
      "timestamp": 1642682101744,
      "id":
"bagaaiera6oioegf7f7t6dlkbzmgswibe5h3l4b2z2b2
pey2af4pw3ve26yca"
    }
  ],
  "subscribed": [],
  "subscribers": ["demo", "test1"],
  "username": "skdgt",
  "peerId":
"QmcKqmDw4NbiXLw6hEpNGjqyTsMgJLQ3MPvxZ
m5qmcyAGS",
  "updated": 1642682284793
}
```


Local storage

Saving periodically own record and subscribed users' records

Export data

Users can export their own record and/or feed, downloading a JSON file with the respective information

Ephemeral content

Storage Limit

Each user only stores up to 100 post for each subscribed user. When this limit is reached, older posts are removed.

Time Limit

Records not updated within 24 hours are removed from local storage.

Offline Protocols

Prune

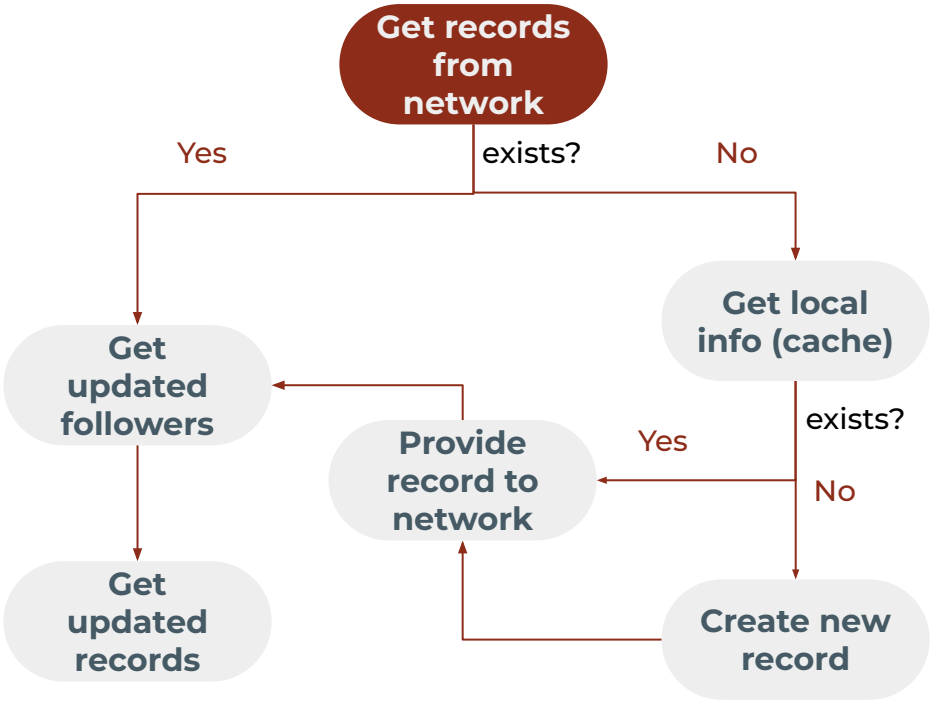
Remove users who are not following us anymore, while we were offline

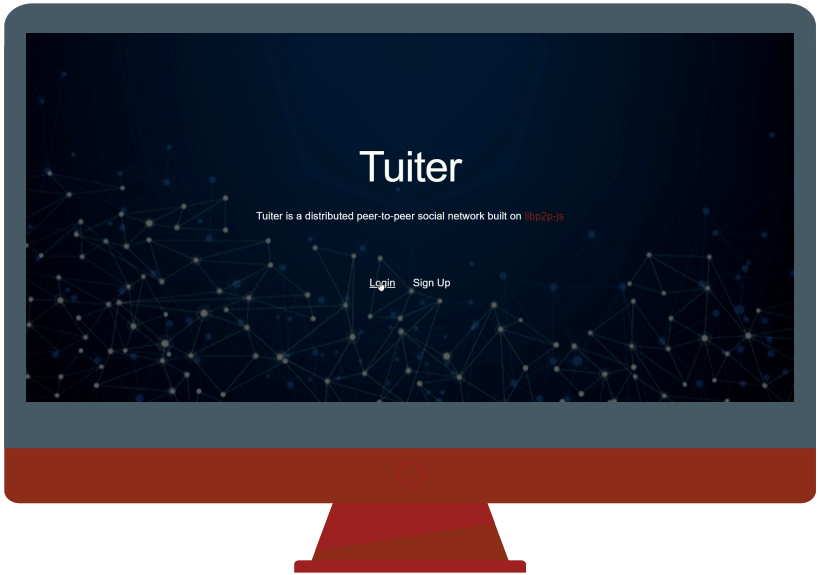
Trigger Update

When a user comes online, issue a trigger for every subscribed user to update their record

Therefore, when the user logs in, has their following users' list updated and makes them update their record on the network.

What happens when a user logs in?





DEMO

A video representation of our implementation

THANKS

Any Questions?

Ana Cruz | up201806460@up.pt
André Nascimento | up201806461@up.pt
Gonçalo Teixeira | up201806562@up.pt
Gonçalo Pereira | up201705971@up.pt