



# Web App Restful API System Design

Igor Baio Soares

# Index

General vision

Description high level design

Web App UI

Data Model

Restful API

Web Server

Preview: A note web app

Web/Mobile app (PWA)



# General vision

- Note App
- Run in a client Browser
- User insert blocks of text like a post-it
- User can see the list of saved notes
- User can delete notes
- The app can be a PWA
  - Progressive Web App can be seen as a hybrid evolution between regular web pages (or websites) and a mobile application

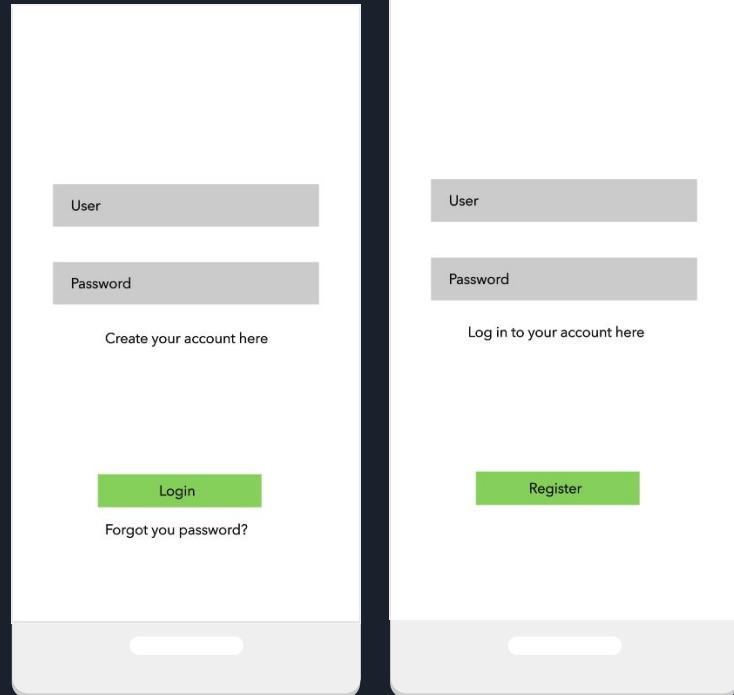


# Description high level design

- 
- 01 User make the login or register in the app
  - 02 User can see the notes' list
  - 03 User can add a new note
  - 04 User can delete some note



# Web App UI



01

Login/Register screen



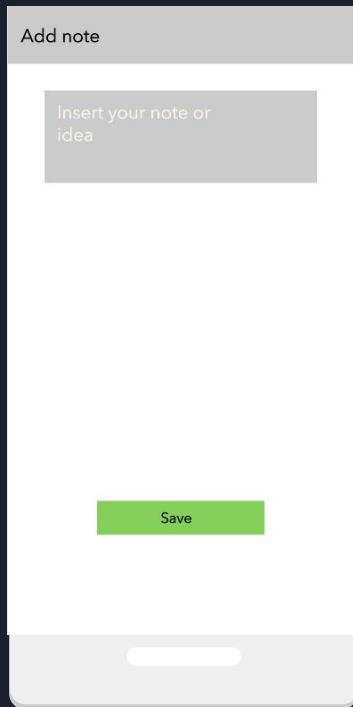
# Web App UI



02

List notes screen

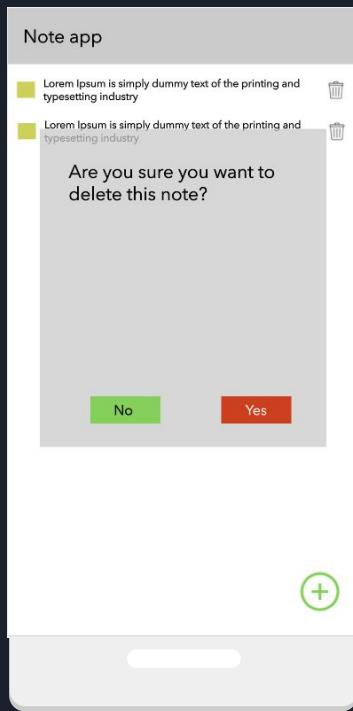
# Web App UI



03

Add note screen

# Web App UI



04

Delete note modal screen



# Data Model

The note will be modeled with the following properties:

- Note ID: Unique identifier for each note.
- Content: Text content of the note.
- Author: Text content the author of note.
- Timestamp: Time when the note was created.



# RESTful API

Resume about HTTP verbs:

- GET
  - Retrieves data from the server.
  - No data sent in the request body.
- POST
  - Submits data to the server.
  - Used for creating new resources or submitting form data.
- DELETE
  - Removes specified resource from the server.
  - Used for deleting resources like files or records.



# RESTful API

In the RESTful API we have three endpoints:

- GET /notes: Retrieve all notes belonging to the logged-in user.
- POST /notes: Save a new note for the logged-in user.
- DELETE /notes/{noteId}: Delete a specific note belonging to the logged-in user.



# Web Server

Once receiving requests from the API, our web server will respond to them with their respective actions:

- GET /notes: Fetch all notes associated with the user from the database and return them to the frontend.
- POST /notes: Receive the new note from the frontend, validate it, associate it with the logged-in user, and save it to the database.
- DELETE /notes/{noteId}: Receive the note ID to delete, verify that it belongs to the logged-in user, and delete it from the database.

Thank you!

