

# Advanced Maintenance for the Railroad using Augmented Reality

Front-end development



**Instituto Superior  
de Engenharia**

Politécnico de Coimbra

# SUMMARY

Introduction

---

Research

---

Web Application

---

Mobile Application

---

Conclusion

---

Demo

---

1  
2  
3  
4  
5  
6

# 1. Introduction

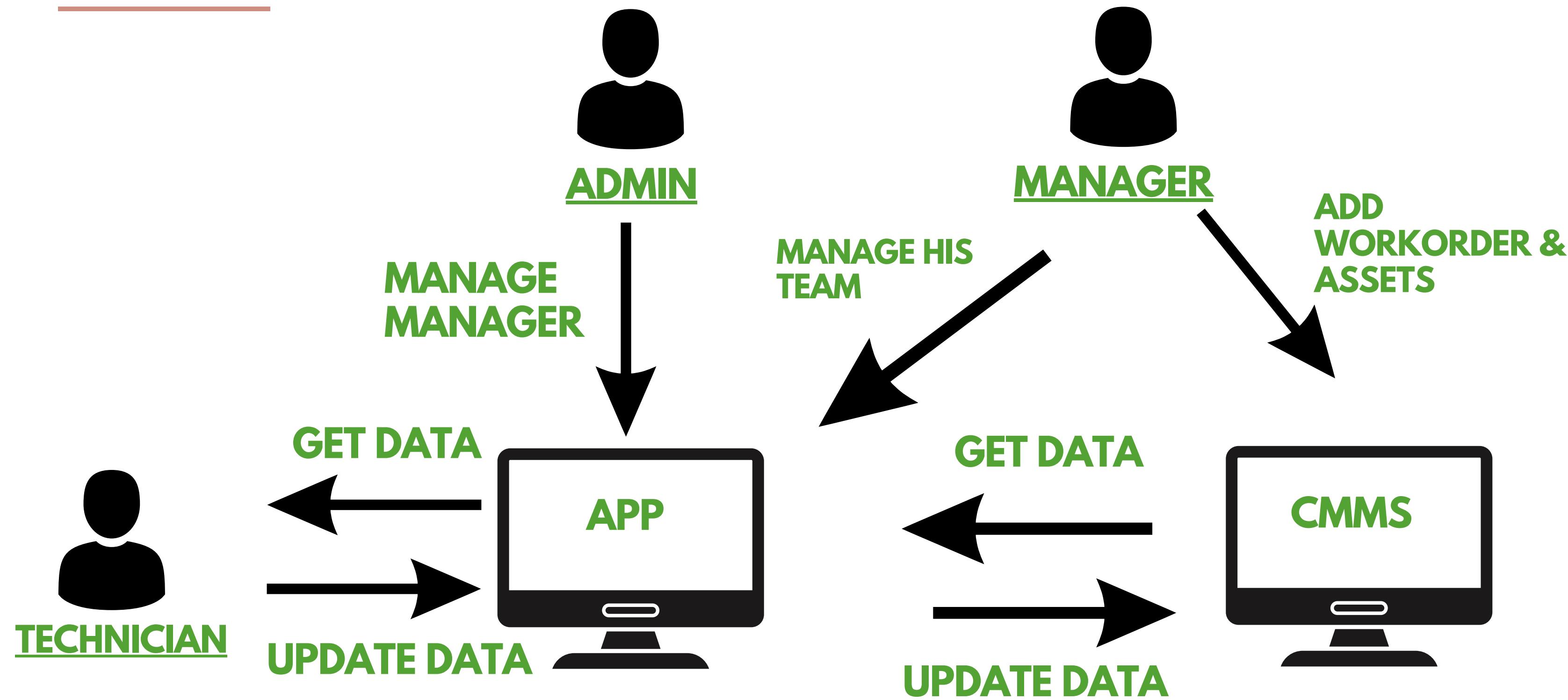
---

# Presentation of the project

---

- Developed a full-stack website
- Utilized Django framework for both backend and frontend development.
- Make the link between a CMMS and the augmented reality
- Part of a larger research endeavor titled "Advanced Maintenance for the Railroad using Augmented Reality"

# Plan of the project

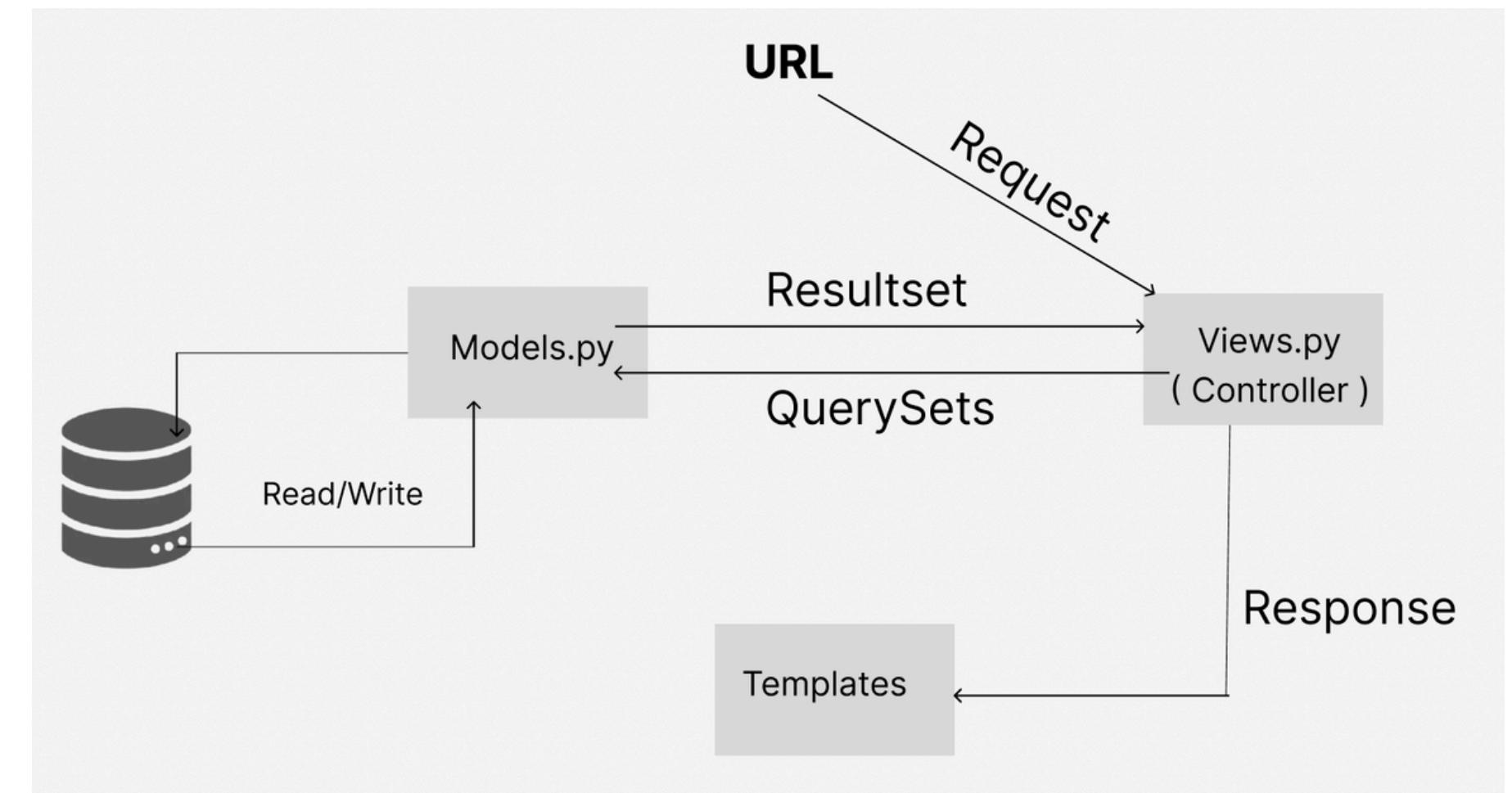


# 2. Research

---

# What is Django ?

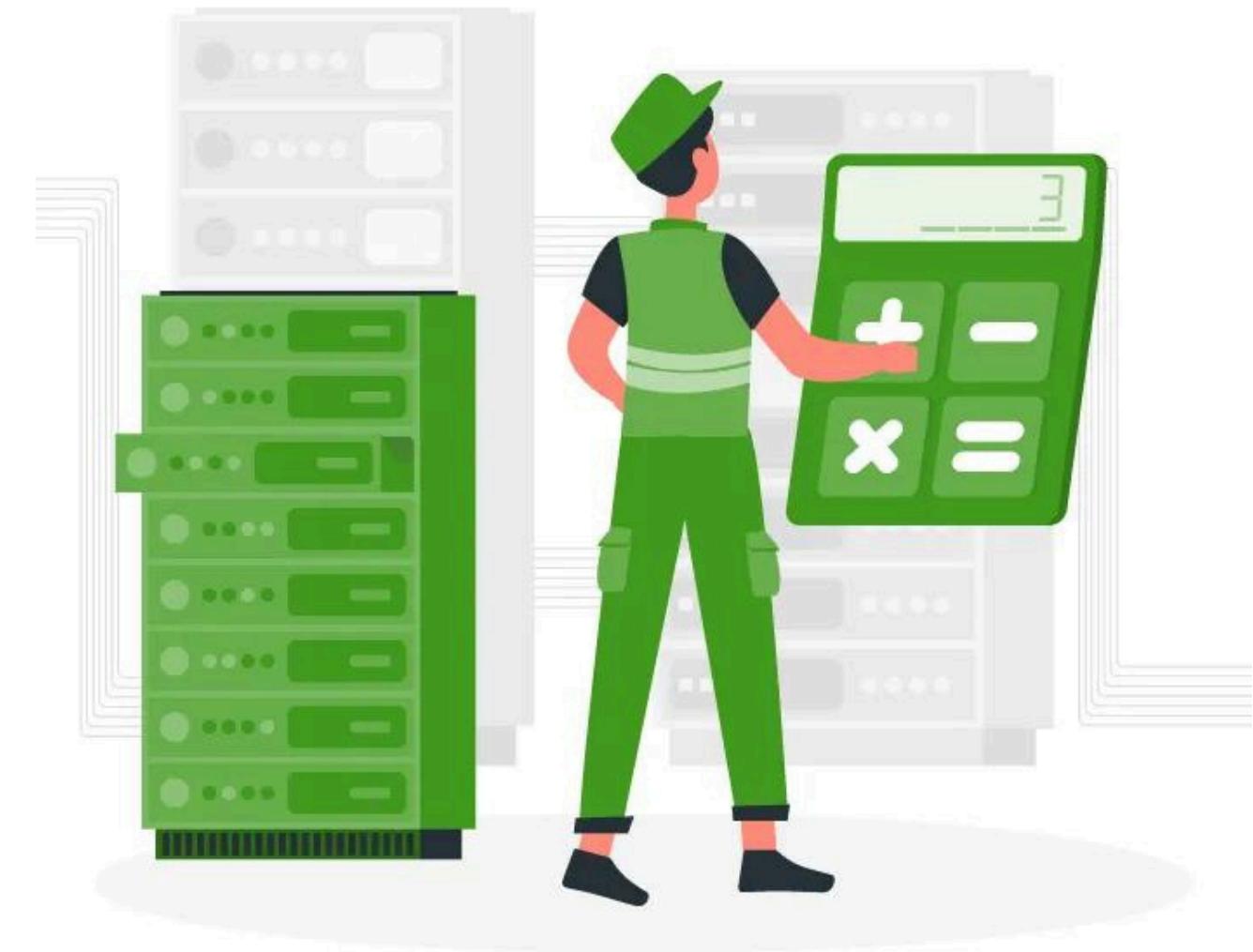
- Launched in 2005
- Open-source web framework
- written in Python
- following the MVT Architecture (Model-View-Template)



# What is a CMMs ?

---

- Computerized Maintenance Management Systems
- Centralized Data Repository
- Streamlined Maintenance Operations
- Instant Information Access
- Improved Efficiency
- Informed Decision-Making



# What is augmented Reality ?

---

- Virtual Elements Overlay
- Enhances Real Environment
- Product Visualization
- Interactive Learning Tool
- Immersive Gaming Experiences



# Integration of Augmented Reality into Maintenance

---

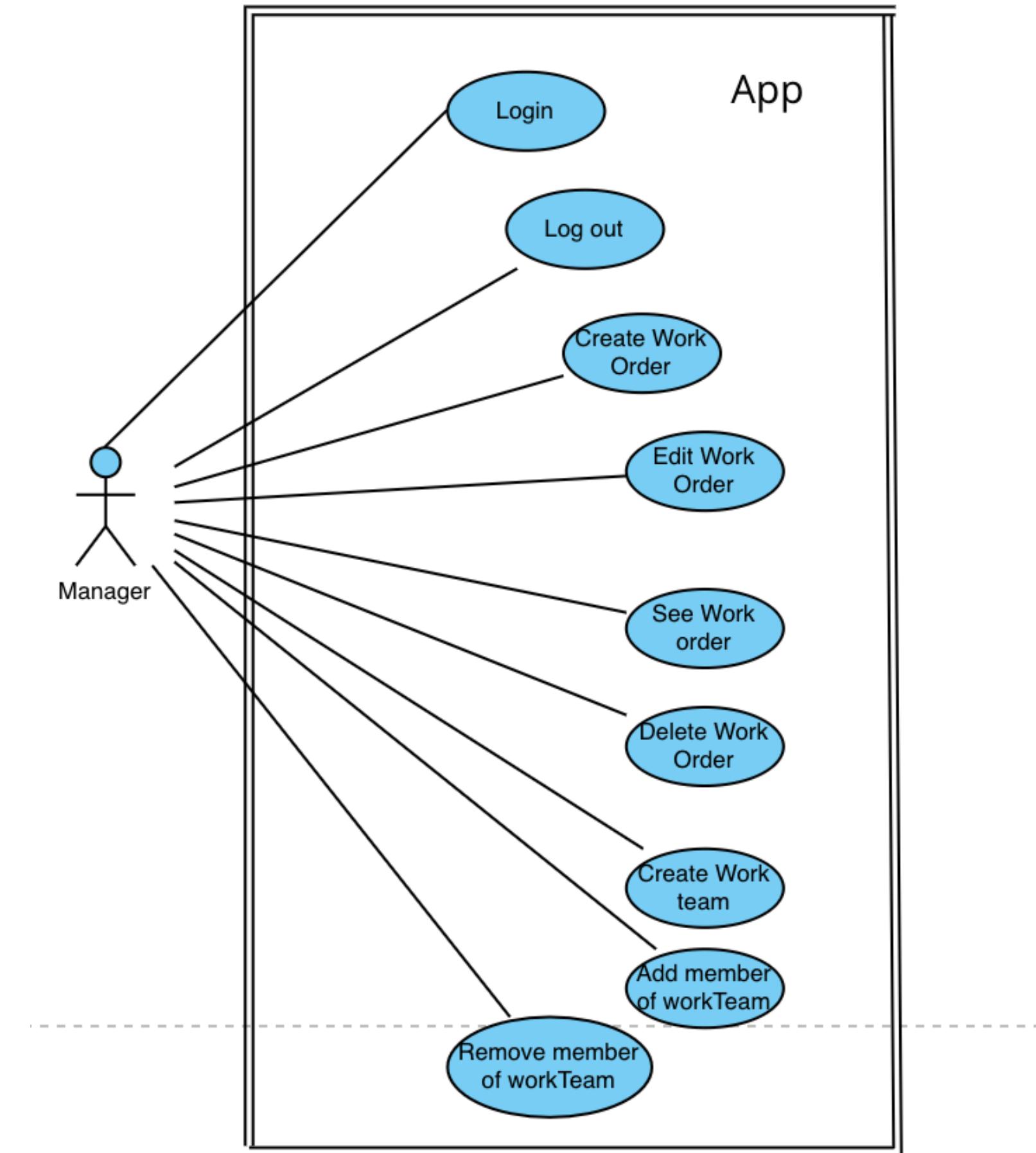
- Operational Efficiency
- Real-time Information
- Technician Assistance
- Error Reduction
- Prototype Development
- Python OpenCV



# Web 2. Application

---

# Requirement Definition with Use Cases



# User Interface Design with Mockups

The image displays four different screens of a mobile application interface, arranged in a 2x2 grid.

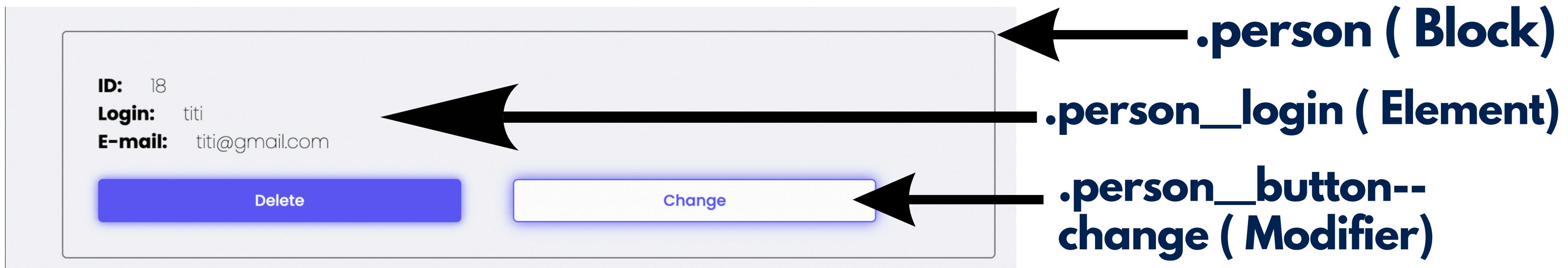
- Top Left:** A login screen titled "Login into your account". It features a large illustration of a person in a dynamic pose, surrounded by network nodes and data points. Below the illustration are fields for "Login" (with placeholder "Enter your login") and "Password" (with placeholder "Enter your password"). There is also a "Forgot password?" link and a "Login now" button. At the bottom is a "Sign Up Now" button.
- Top Right:** A dashboard titled "My App". The header includes a profile icon for "BOUIN Julien" and a search bar. The main content area is divided into sections: "Work Orders" (with "All Work Orders" and "My Work Orders" options), "Messaging", and three repair tasks: "Repair climatisation" (due 24/05), "Repair filter" (due 28/05), and "Repair others" (due 31/05). Each task has a description and a "Select" button.
- Bottom Left:** A dashboard titled "My App". The header includes a profile icon for "BOUIN Julien" and a search bar. The main content area is divided into sections: "Work Orders" (with "All Work Orders" and "My Work Orders" options), "Messaging", and a welcome message: "Welcome BOUIN Julien, in this space you can see all the information relating to your work orders. You can also chat with your work colleagues via integrated messaging".
- Bottom Right:** A dashboard titled "My App". The header includes a profile icon for "BOUIN Julien" and a search bar. The main content area is divided into sections: "Work Orders" (with "All Work Orders" and "My Work Orders" options), "Messaging", and a welcome message: "Welcome BOUIN Julien, in this space you can see all the information relating to your work orders. You can also chat with your work colleagues via integrated messaging".

At the bottom of the grid, there are small links: "Copyright © 2024", "Terms of uses", and "Privacy Policy".

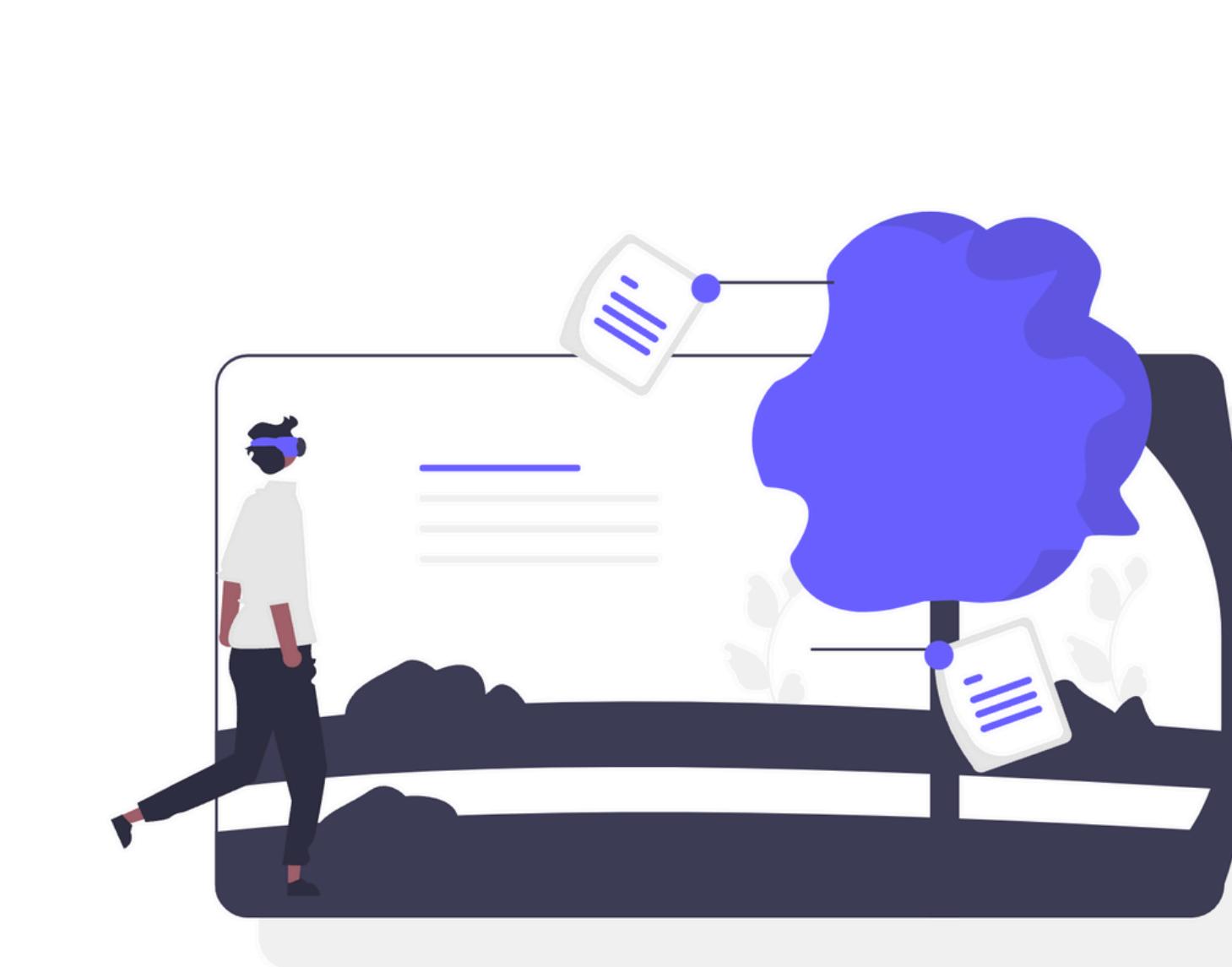
# Frontend Development using Django Templates

---

# BEM methodology ( Block, Element, Modifier )



# Login Page



The illustration depicts a person in a white shirt and dark pants running towards a large, stylized blue cloud or head silhouette. In the foreground, a smartphone screen shows a login interface with a user icon and a password field. The background features a minimalist landscape with small green plants.

**Login into your account**

Login :

Password :

**Log in**

# Main Page

---

The image displays two side-by-side screenshots of a web-based application interface, likely a management or tracking system.

**Left Screenshot (Main Page):**

- Header:** "My App" on the top left, user icon "admin" with a gear icon, and a "Logout" button.
- Sidebar:** "Gestion Manager" section with "Add Manager" and "See Managers" options.
- Content Area:** A large blue banner with the text "Welcome admin ( Admin ) !". Below it, descriptive text: "in this space you can see all the information relating to your work orders." and "You can also chat with your work colleagues via integrated messaging ."

**Right Screenshot (Account Information):**

- Header:** "My App" on the top left, user icon "admin" with a gear icon, and a "Logout" button.
- Section:** "My account Informations" (Note: "Informations" is misspelled).
- Fields:** "Login: admin", "E-mail: Not specified", "Firstname: Not specified", and "Lastname: Not specified".

# Layout

The screenshot shows a mobile application interface. At the top, there is a dark header bar with the text "My App". Below this, the main content area has a white background. On the left side, there is a dark sidebar containing a section titled "Gestion Manager" with two options: "Add Manager" and "See Managers". A red rectangular border highlights this sidebar area. To the right of the sidebar, the main content area features a large, bold, black text "Welcome admin ( Admin ) !". Below this, there are two paragraphs of text in black: "in this space you can see all the information relating to your work orders." and "You can also chat with your work colleagues via integrated messaging .". In the top right corner of the main content area, there is a user profile icon showing a person's head and shoulders, followed by the text "admin" and a small window-like icon.

My App

Gestion Manager

Add Manager

See Managers

Welcome admin ( Admin ) !

in this space you can see all the information relating to your work orders.

You can also chat with your work colleagues via integrated messaging .

admin

5

# Responsive Design

My App

Gestion Manager  
Add Manager  
See Managers

Welcome admin ( Admin ) !

in this space you can see all the information relating to your work orders.  
You can also chat with your work colleagues via integrated messaging .

L'écran 2...11.23.10

# Dynamic data

The screenshot shows a web application interface titled "My App". The top navigation bar includes a user icon labeled "toto" and a file icon.

The left sidebar, titled "Team : totoTeams", contains the following menu items:

- Add Technicien
- See Techniciens
- See our Team Tasks
- Technican without team
- Upload Model
- Model List

The main content area is titled "Technician group members" and displays two entries in a table format:

ID:	Login:	E-mail:	Action
18	titi	titi@gmail.com	<a href="#">Delete</a> <a href="#">Change</a>
20	julien	julienbouin@yahoo.fr	<a href="#">Delete</a> <a href="#">Change</a>

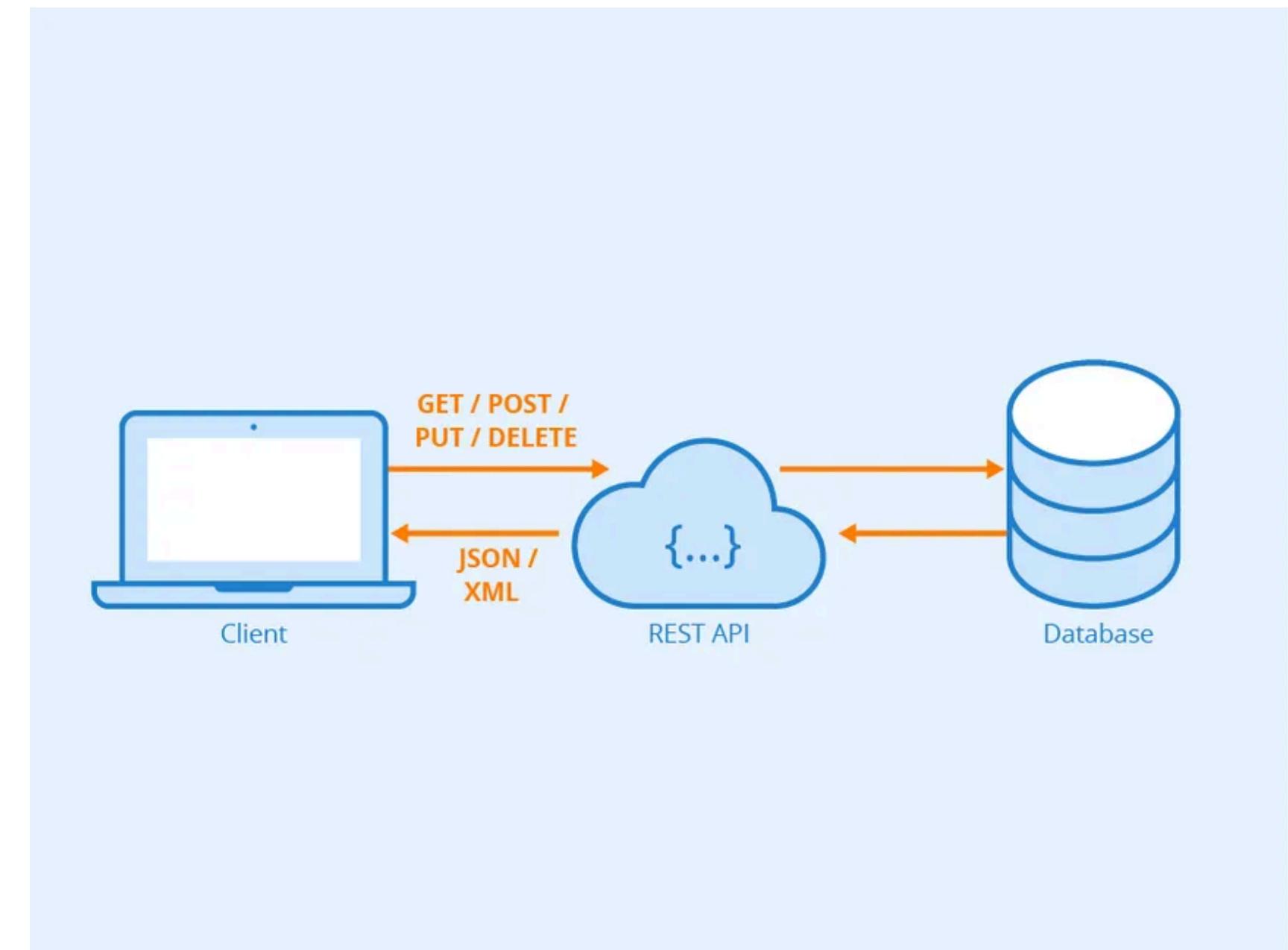
# API Development

---

# What is a API ?

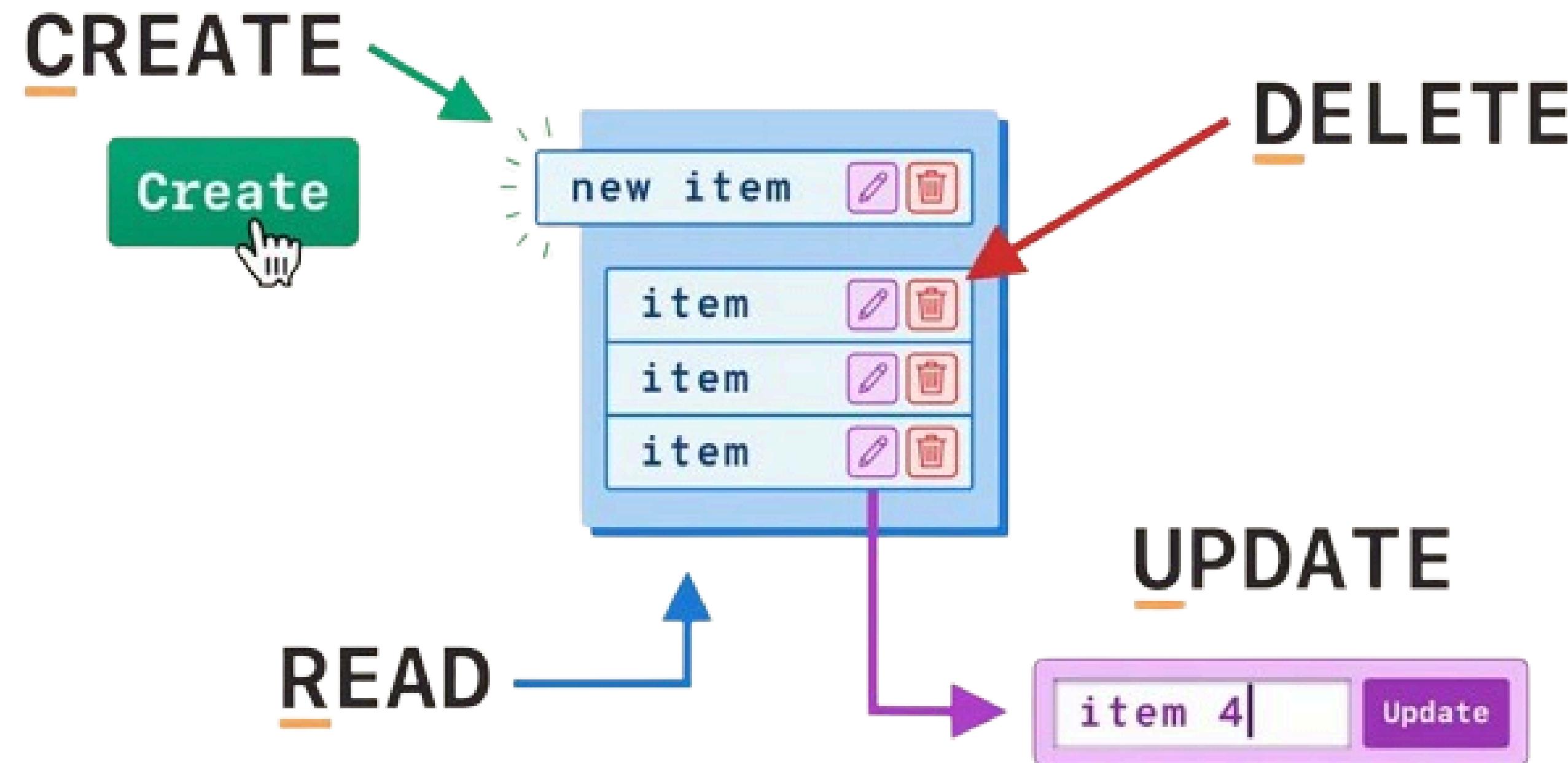
---

- Application Programming Interface
- This is a set of rules and protocols
- Enables communication between software applications
- Facilitates the exchange of data, functionalities, and services between applications.



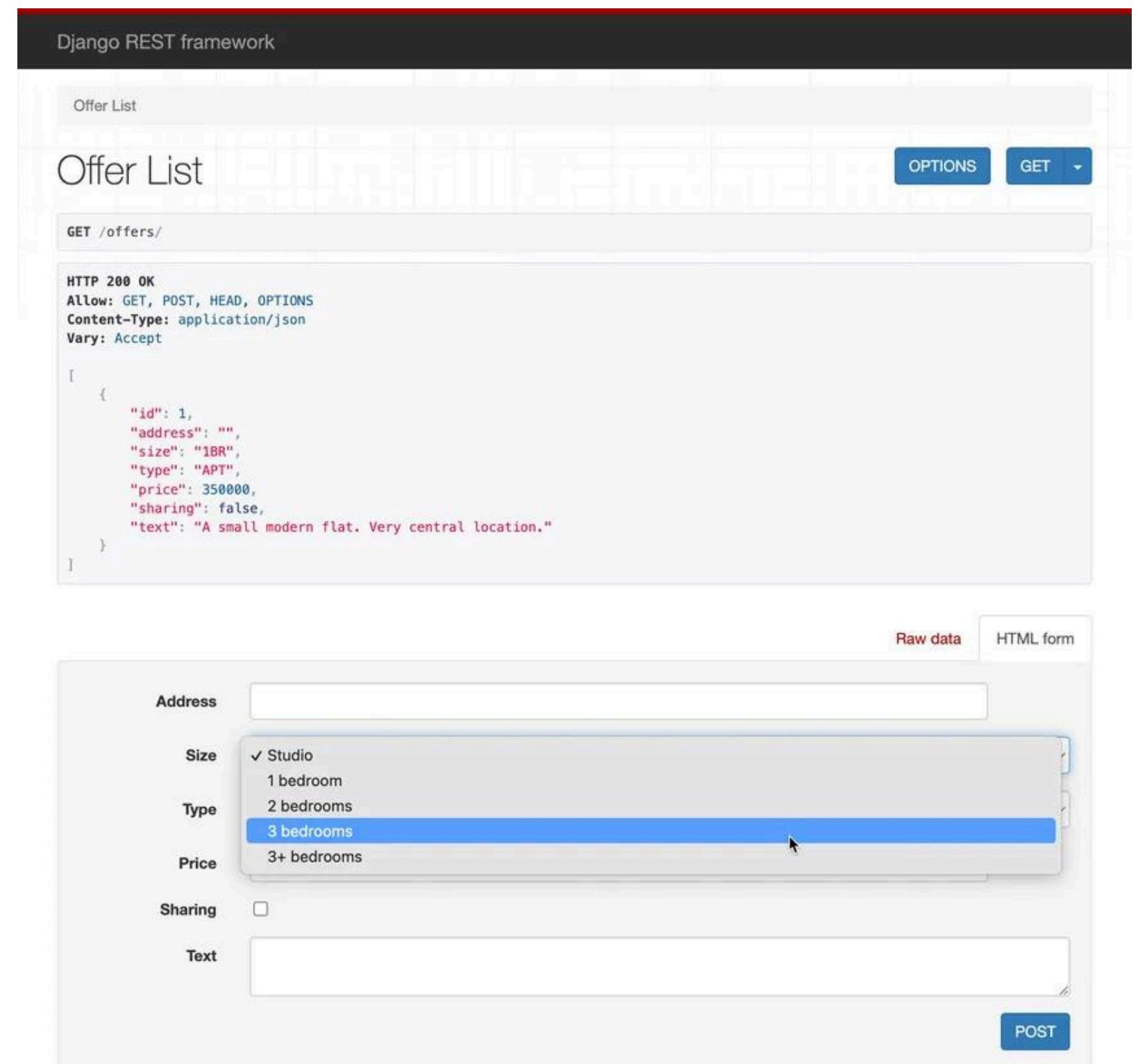
# CRUD

---



# Django Rest Framework

- Used to build Web APIs.
- Extends Django's capabilities to handle RESTful interactions.
- Converts complex data types (like Django models) into JSON or XML.
- Provides a web-based interface to interact with the API for easier testing and debugging.



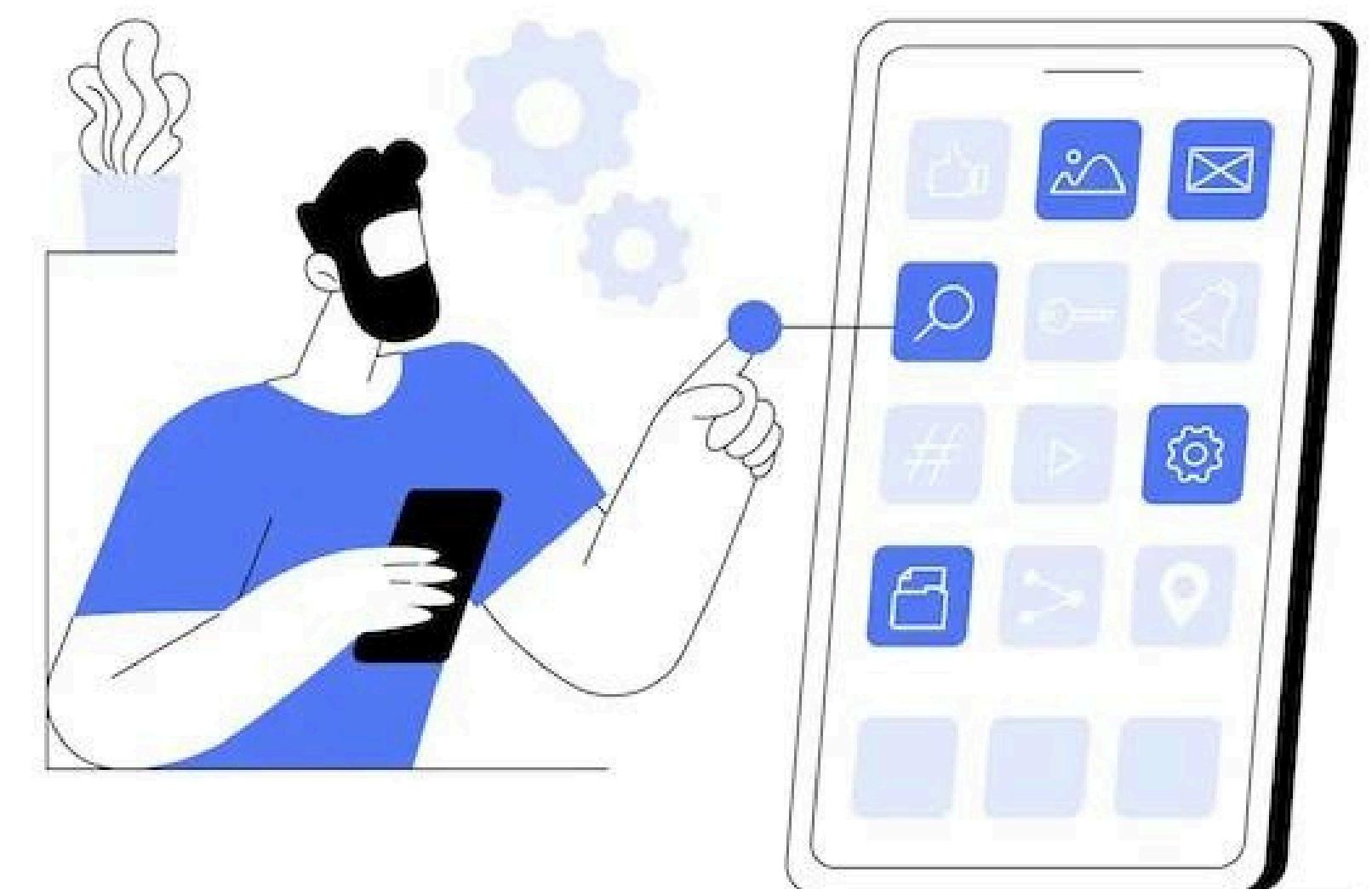
# Mobile

# 3. Application

# Why a mobile application ?

---

- Enables technicians to access the application directly via their phones
- Technicians can log in, view, and select work orders, and add notes similar to the website.
- Major advantage: Technicians can update work orders without an internet connection.
- Allows adding notes and changing work order states offline.



# **Choice of the language**

---

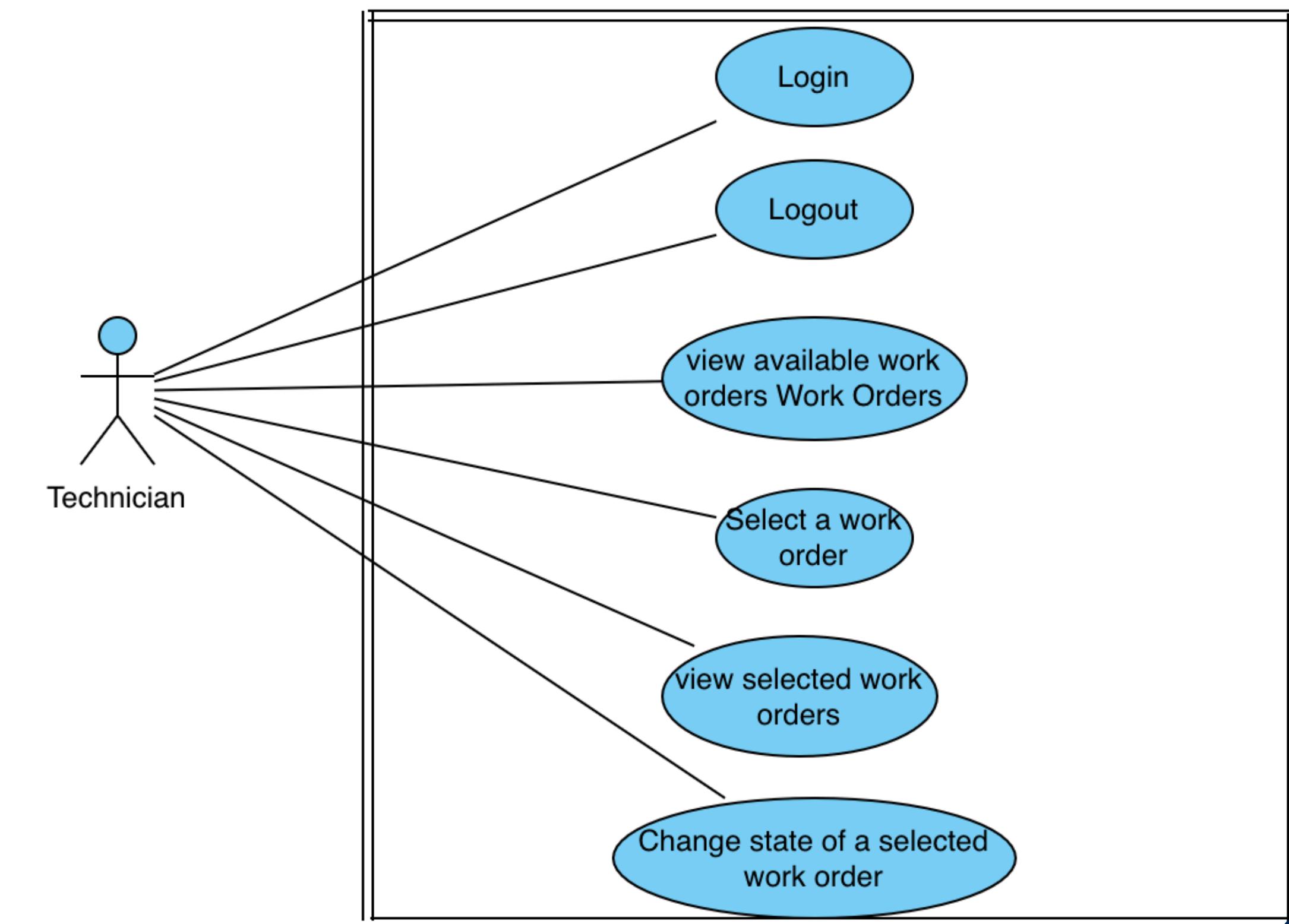
## **Flutter**

- Uses Dart
- Uses own widgets, ensures consistency
- Ideal for high performance and graphics intensive applications
- Compatible with ios and android

## **React Native**

- Uses Javascript
- Uses native components
- Suitable for most applications, but may be slower for graphics-intensive tasks
- Compatible with ios and android

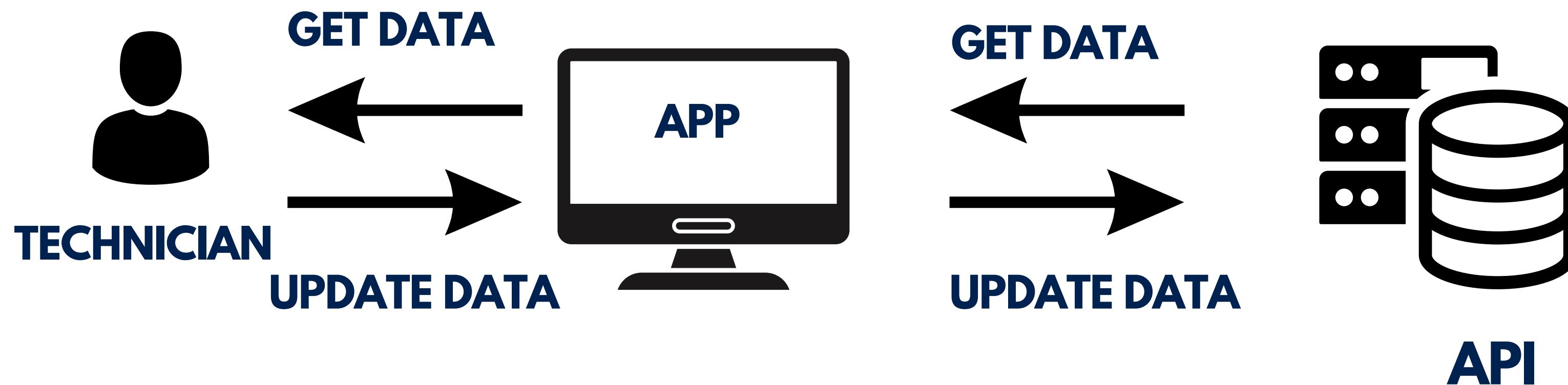
# Requirement Definition with Use Cases



# Development with React Native

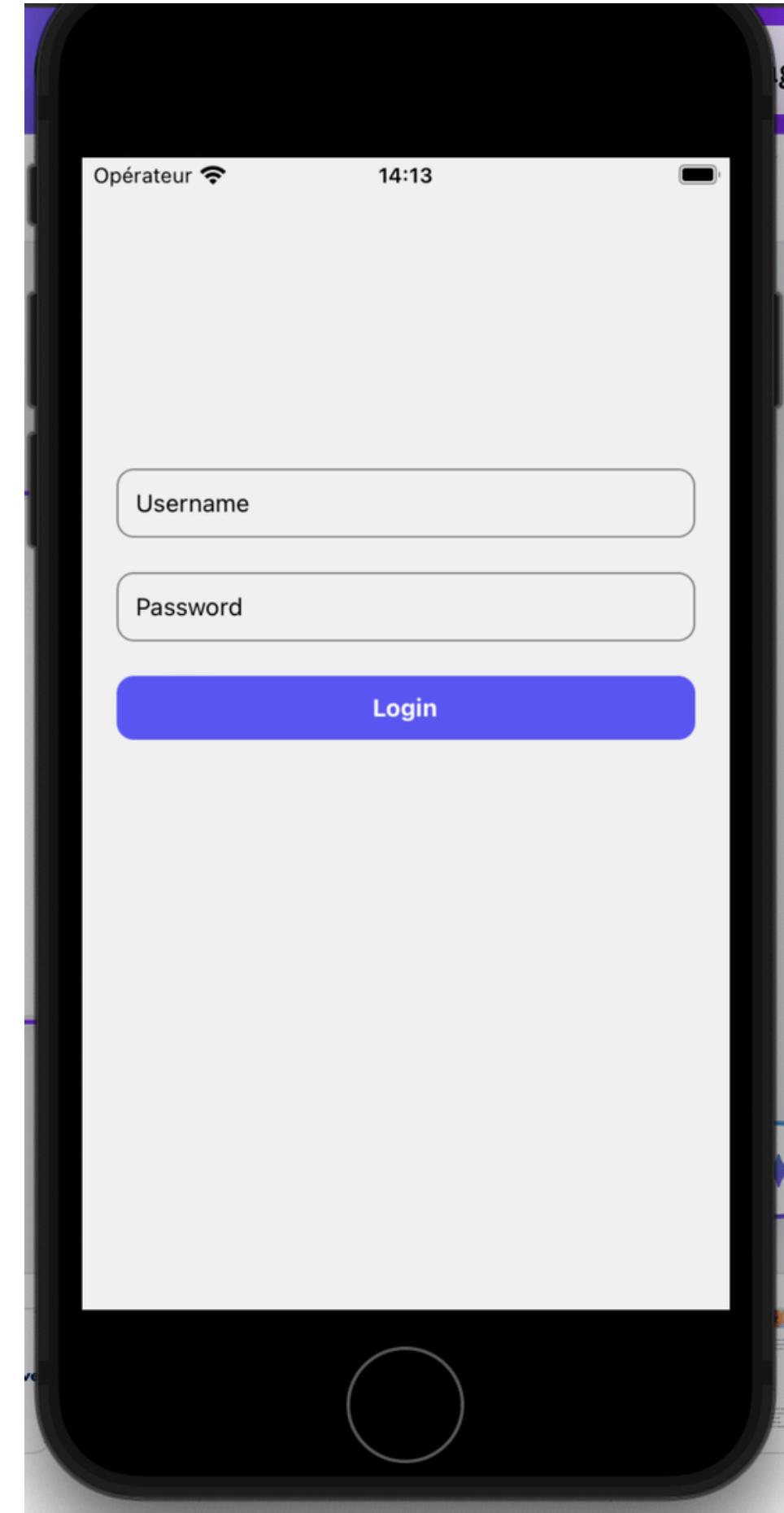
---

# Plan of the application

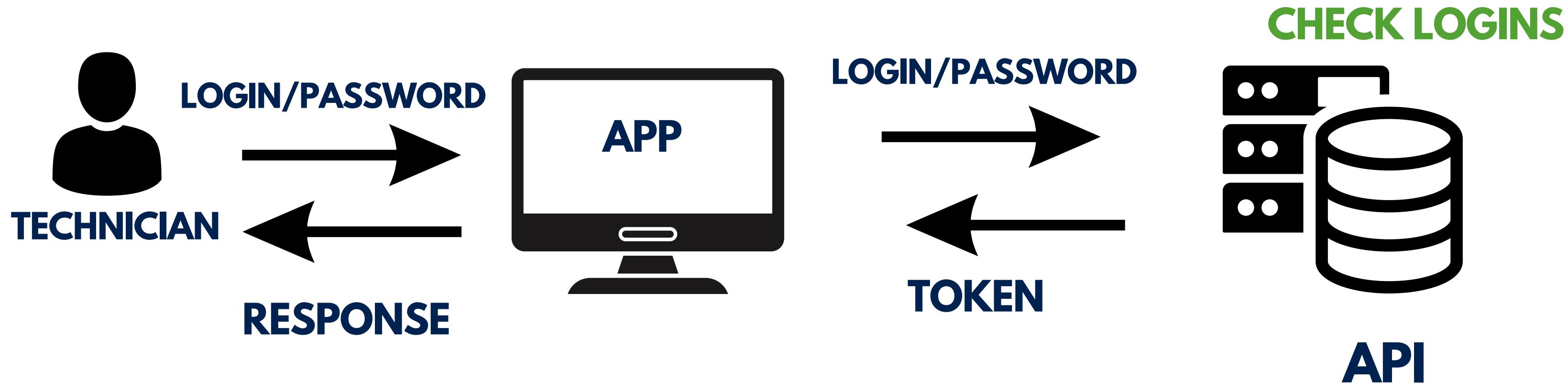


# Login Page

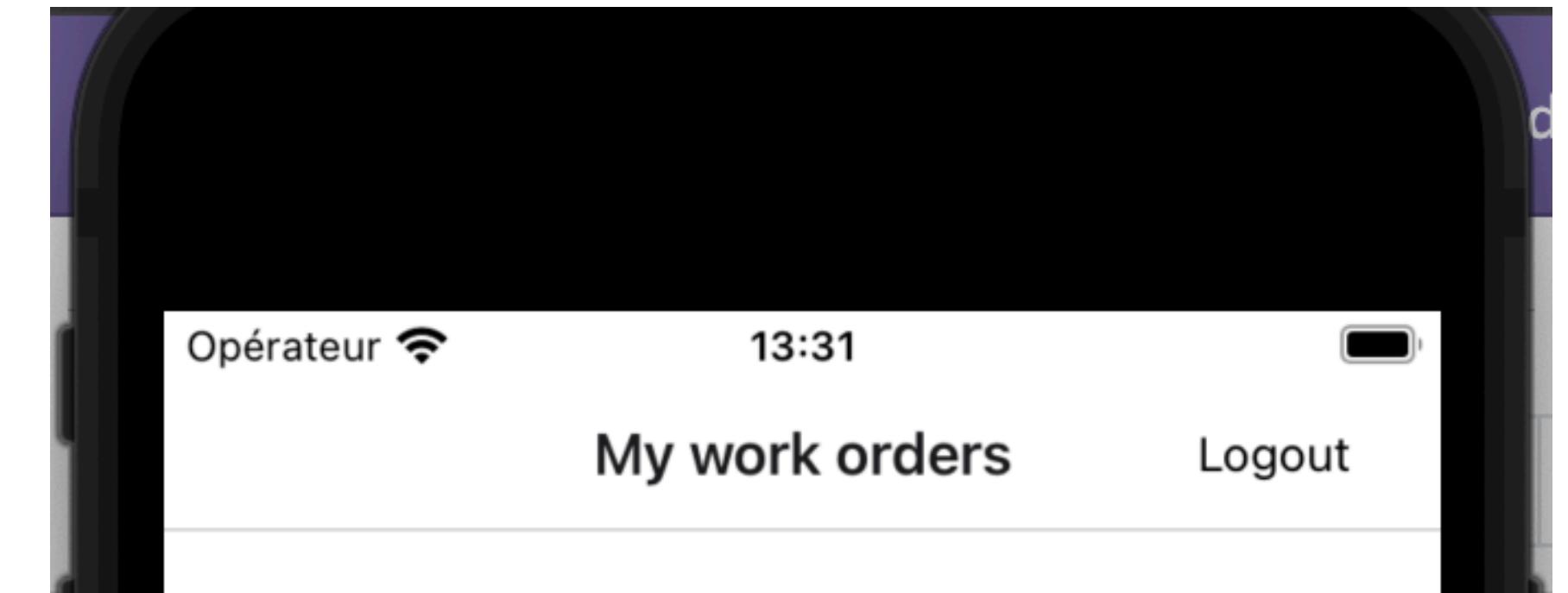
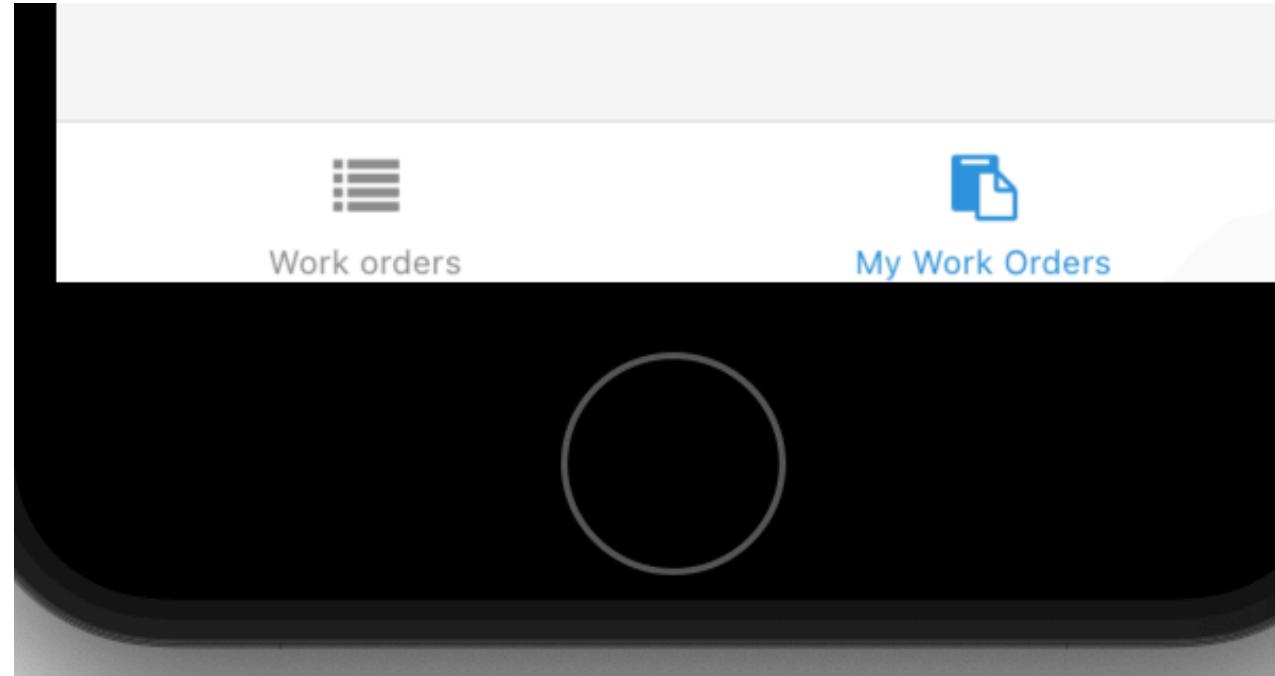
- Frontend : Utilized React Native to create a smooth and responsive user interface.
- Backend : Integrated the REST API to handle authentication with BASIC Token



# Authentification with BASIC Token

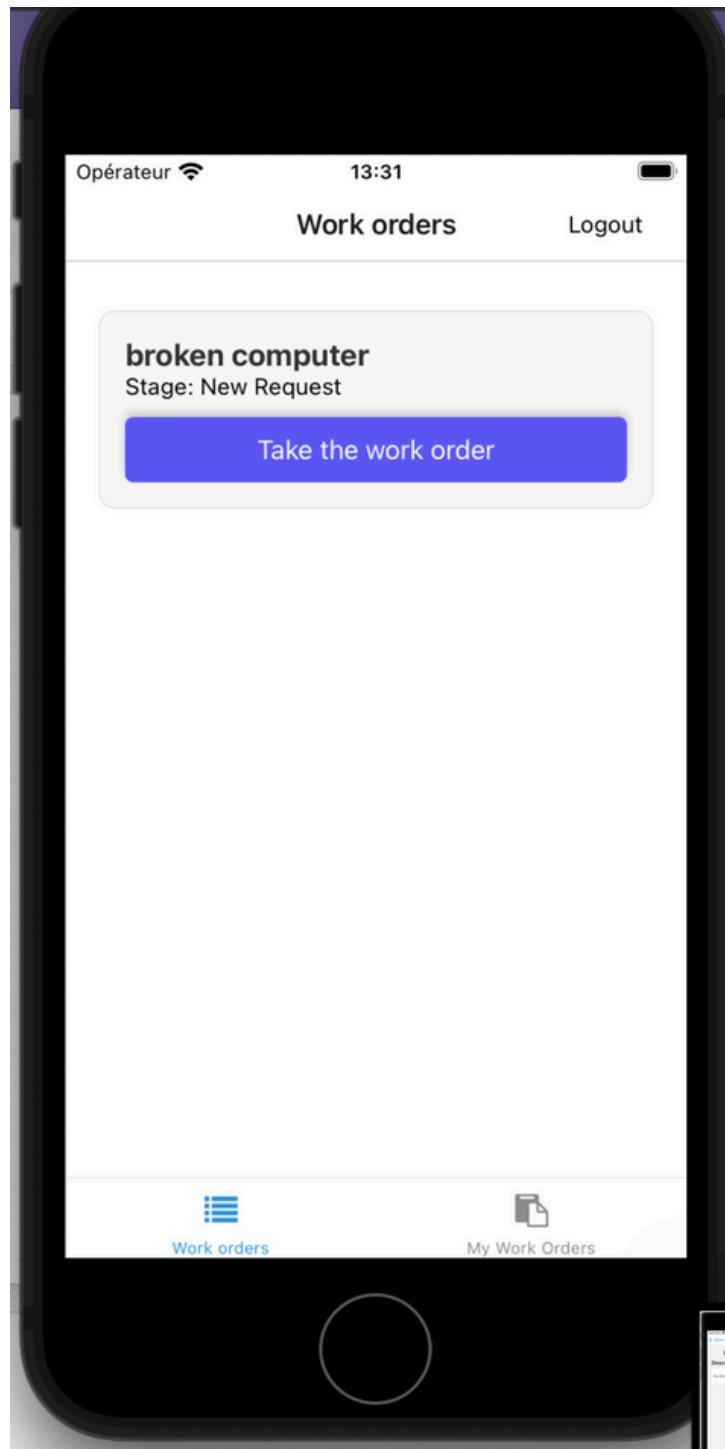


# Overall Structure of the Application

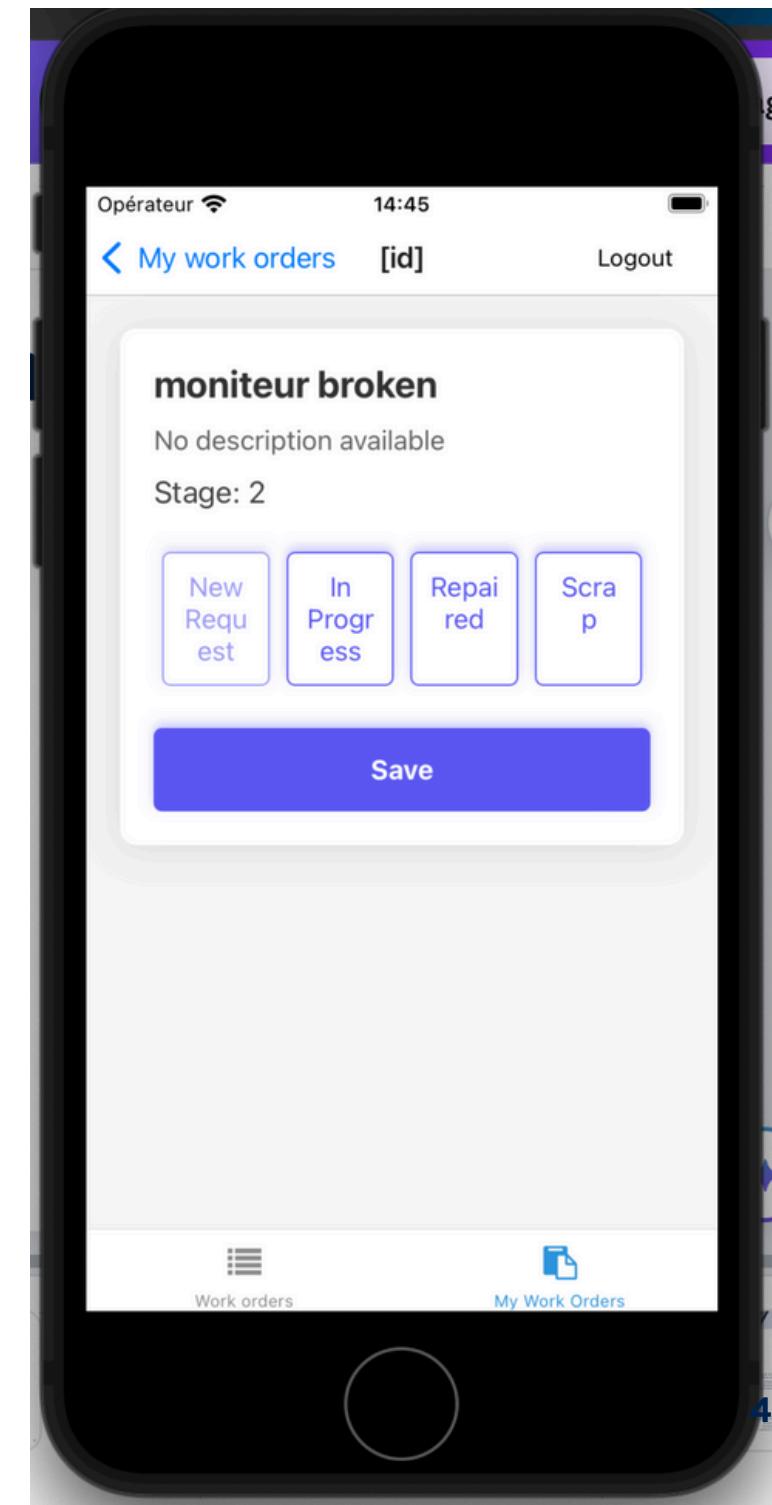


- Navigation : Utilized Expo Router for intuitive and responsive navigation.
- UX : User-centered design with clear visual indicators for location and easily accessible logout options.

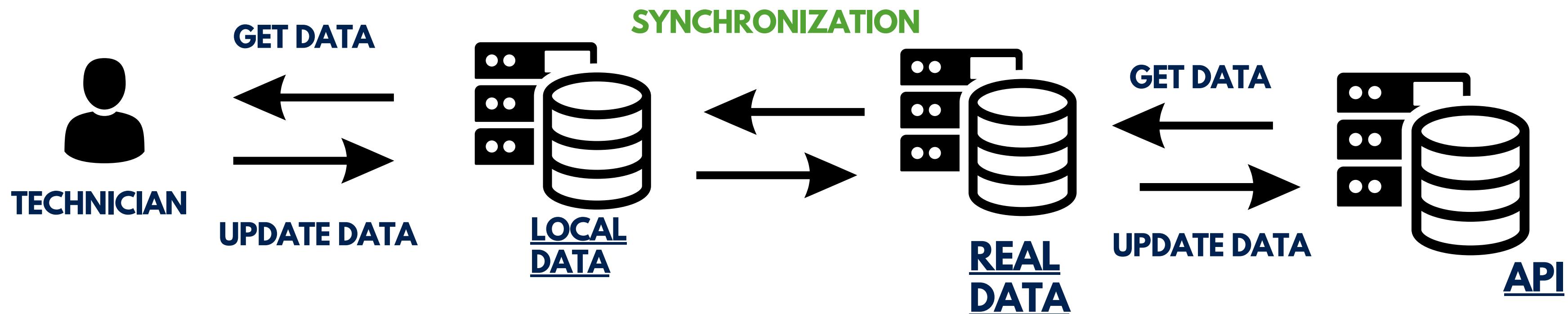
# Displaying and Processing Data



- Using FlatList component
- Custom Component
- Added buttons to modify the status of work orders directly from the app.
- Displayed detailed information for each work



# Offline Data Management



# Test cases

---

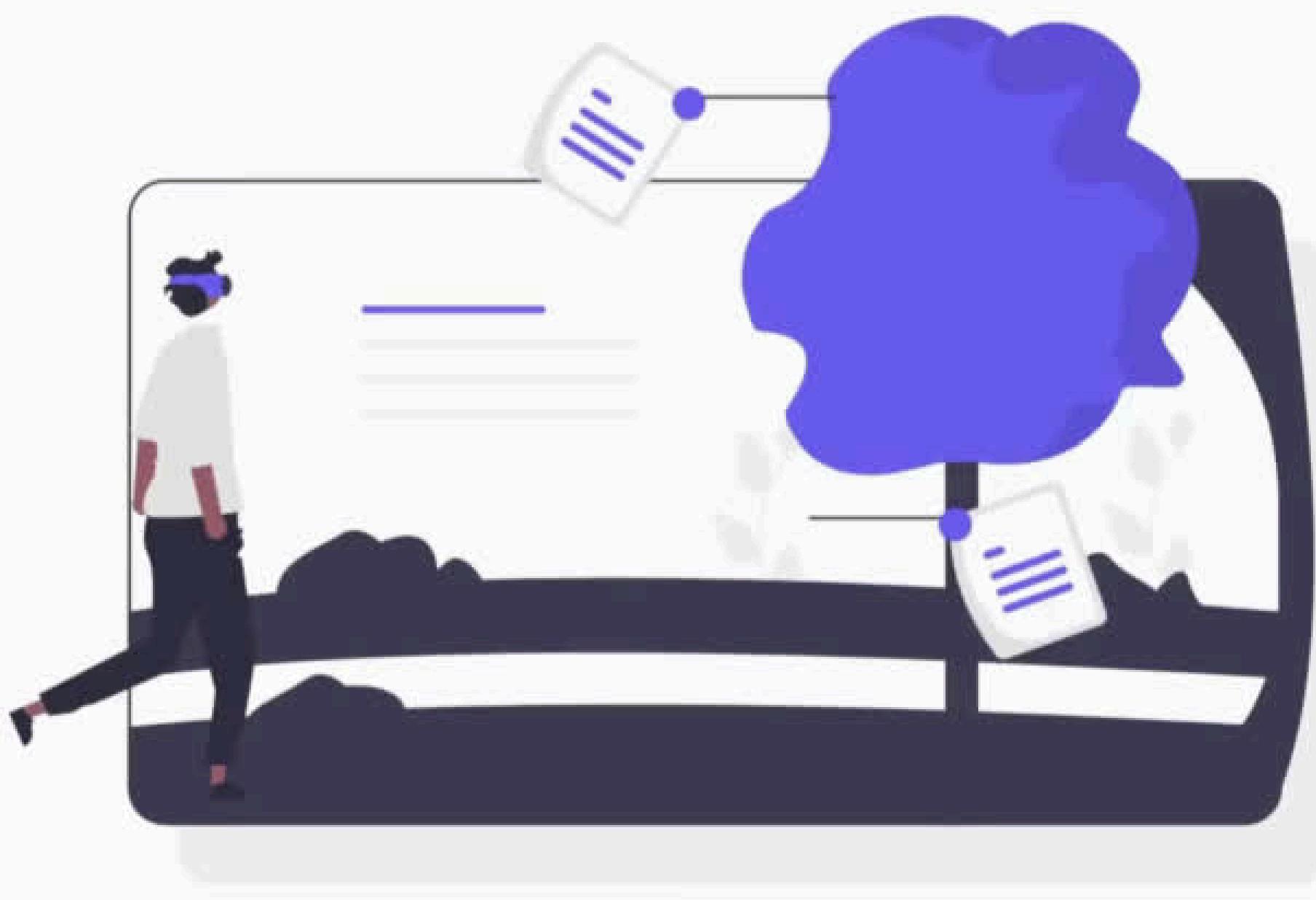
Login	
Logout	
View available work orders	
Select a work order	
View selected work orders	
Change the state of a selected work order	

# 5. CONCLUSION

---

# 6. DEMO

---



Login into your account

Login :

Enter your login

Password :

Enter your password

**Log in**

**Work orders**

Logout

**task3**

Stage: Stage non trouvé

**Take the work order**

Work orders



My Work Orders

Thank you for  
listening

---