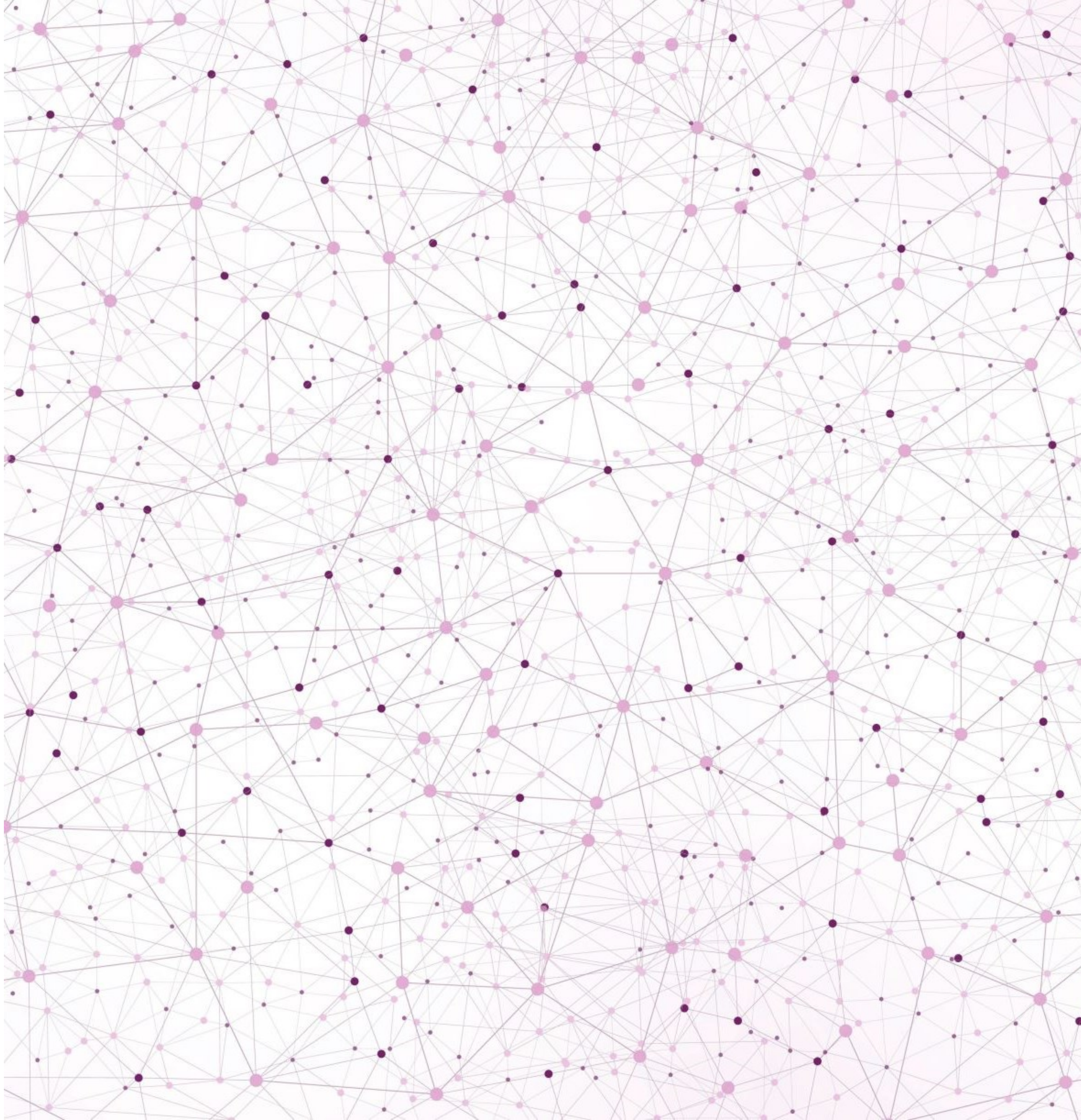
Università di Genova
DIBRIS DIPARTIMENTO DI INFORMATICA, BIOINGEGNERIA, ROBOTICA E INGEGNERIA DEI SISTEMI

Smart Charging Management System

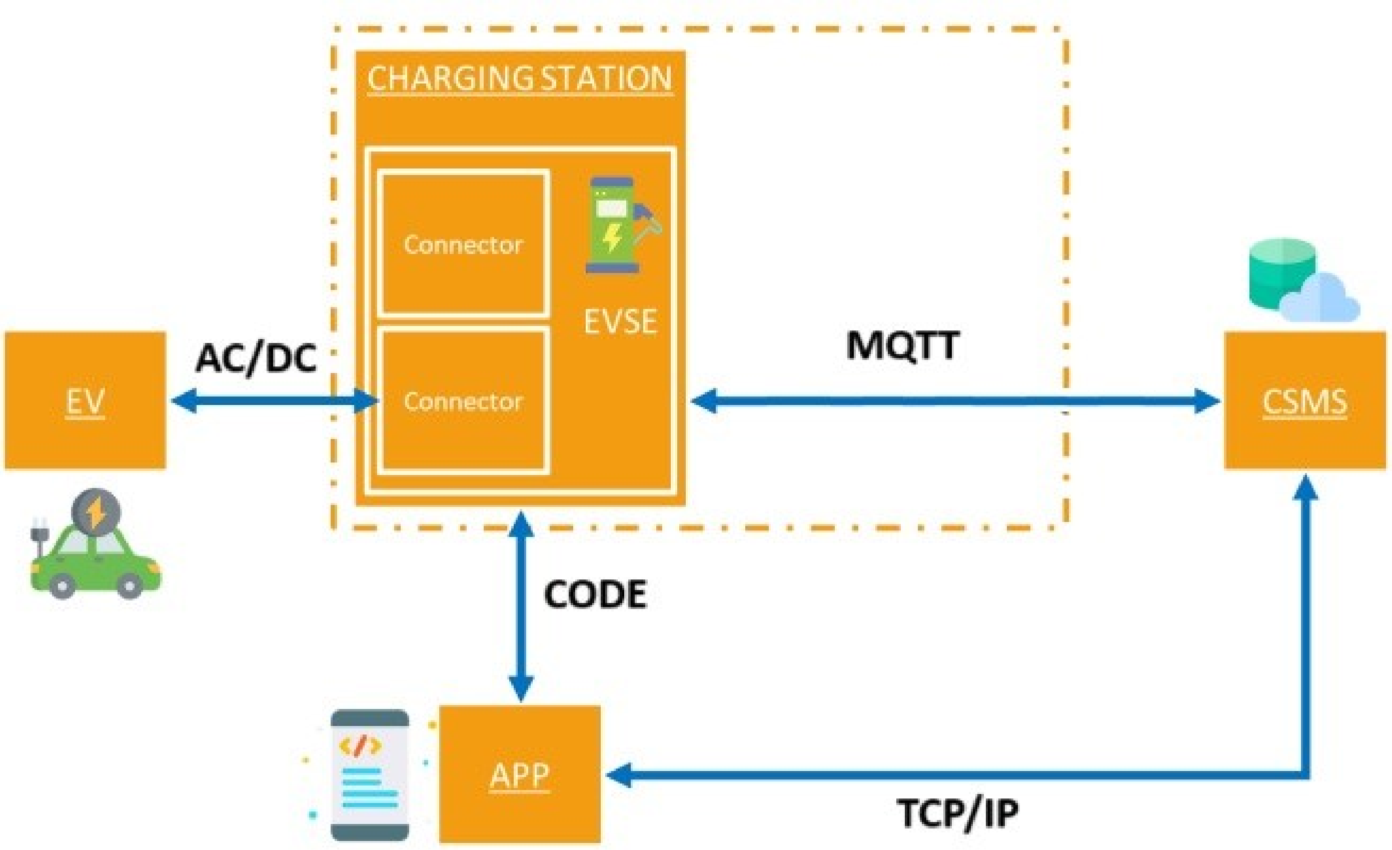
Alexandru Vasile 4812579



OVERVIEW

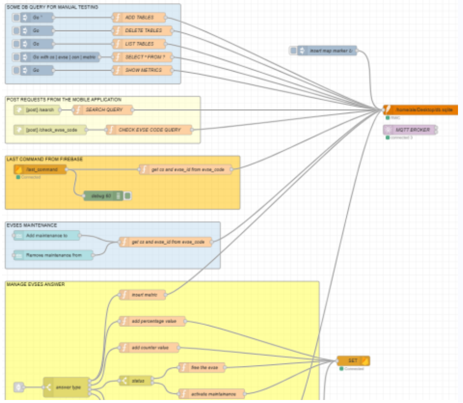
- Architecture
- Implementation
- Communication infrastructure
- Middleware
- Storage
- Dashboard
- Functionalities

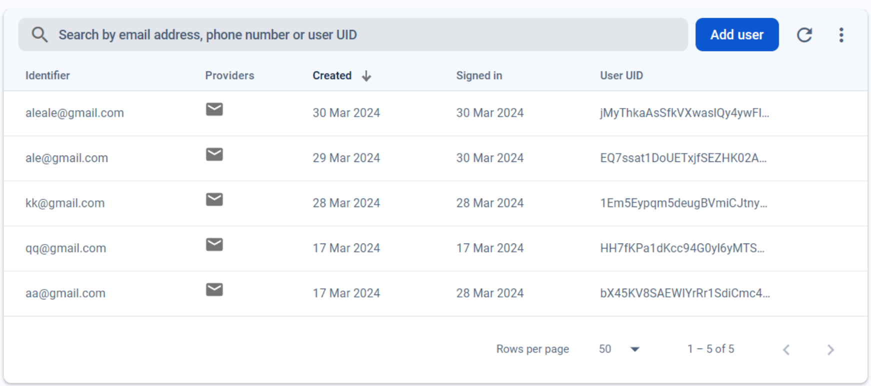
ARCHITECTURE



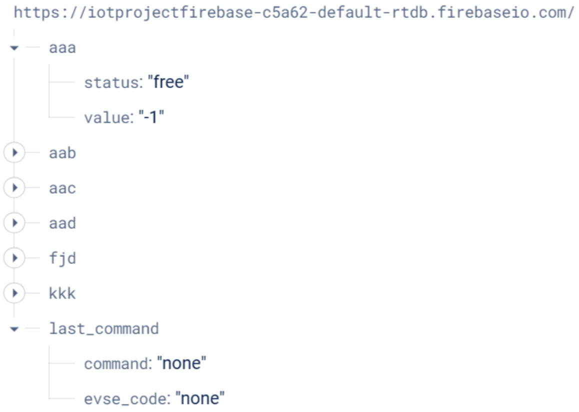
IMPLEMENTATION: CSMS

- NodeRed, a visual programming language
- Firebase, a Google platform that offers different functionalities as authentication and database





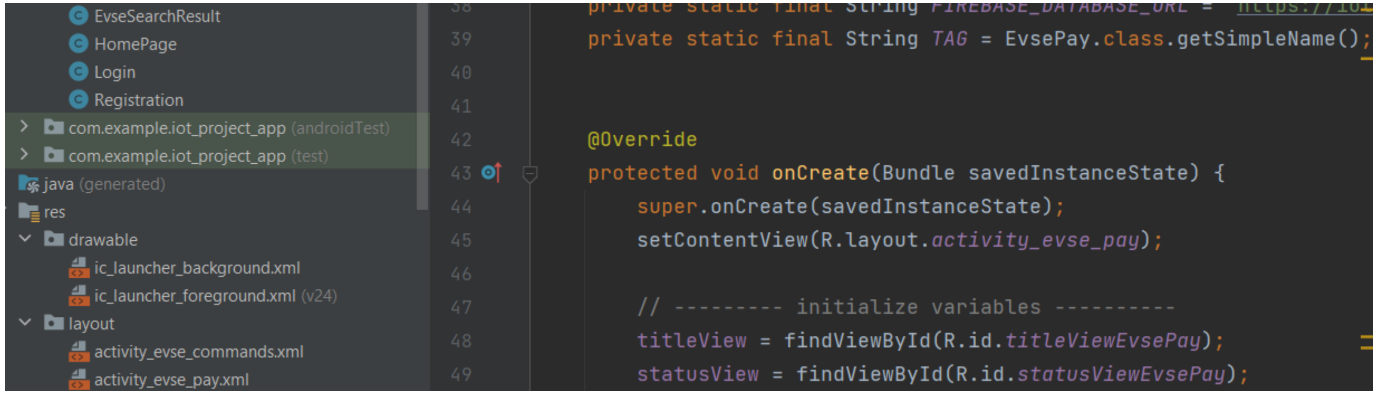
Identifier	Providers	Created ↓	Signed in	User UID
aleale@gmail.com	✉	30 Mar 2024	30 Mar 2024	jMyThkaAsSfkVXwaslQy4ywFl...
ale@gmail.com	✉	29 Mar 2024	30 Mar 2024	EQ7ssat1DoUETxjfSEZHK02A...
kk@gmail.com	✉	28 Mar 2024	28 Mar 2024	1Em5Eypqm5deugBVmiCJtny...
qq@gmail.com	✉	17 Mar 2024	17 Mar 2024	HH7fKPa1dKcc94G0yl6yMTS...
aa@gmail.com	✉	17 Mar 2024	28 Mar 2024	bX45KV8SAEWlYrRr1SdiCmc4...



```
https://iotprojectfirebase-c5a62-default-rtdb.firebaseio.com/
▾ aaa
    status: "free"
    value: "-1"
▸ aab
▸ aac
▸ aad
▸ fjd
▸ kkk
▾ last_command
    command: "none"
    evse_code: "none"
```

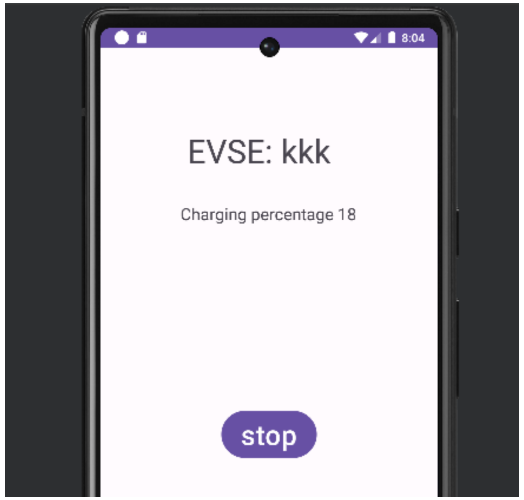
IMPLEMENTATION: APP

- AndroidStudio, an IDE to develop android applications



```
EvseSearchResult
HomePage
Login
Registration
> com.example.iot_project_app (androidTest)
> com.example.iot_project_app (test)
java (generated)
res
  drawable
    ic_launcher_background.xml
    ic_launcher_foreground.xml (v24)
  layout
    activity_evse_commands.xml
    activity_evse_pay.xml
```

```
private static final String FIREBASE_DATABASE_URL = "https://io
private static final String TAG = EvsePay.class.getSimpleName();


@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_evse_pay);

    // --------- initialize variables ----------
    titleView = findViewById(R.id.titleViewEvsePay);
    statusView = findViewById(R.id.statusViewEvsePay);
```

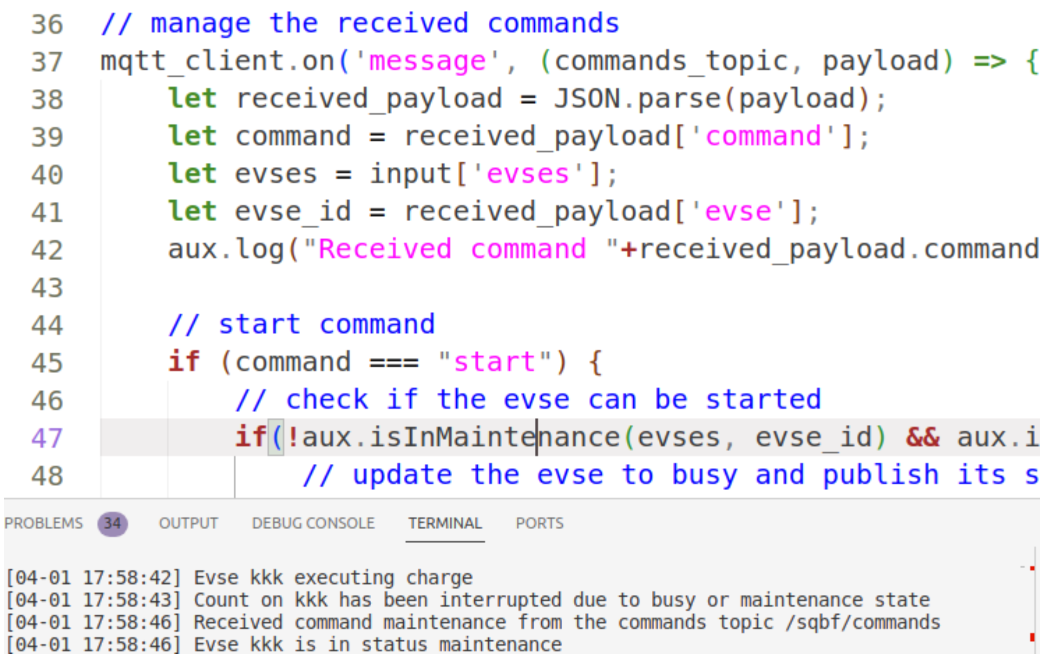


EVSE: kkk

Charging percentage 18

IMPLEMENTATION: CHARGING STATION

- Simulated with JavaScript scripts

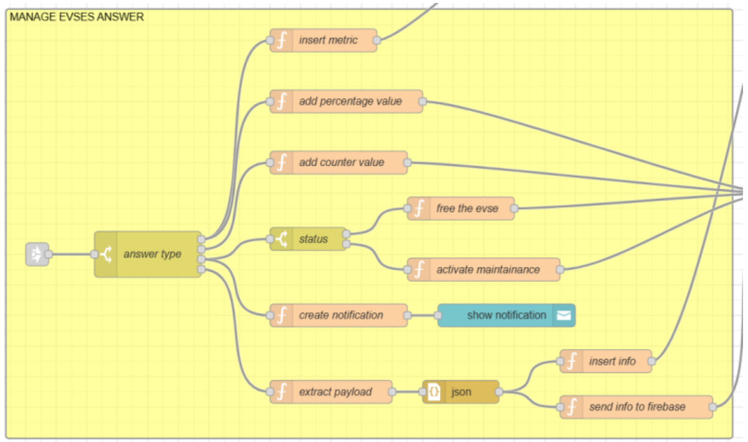


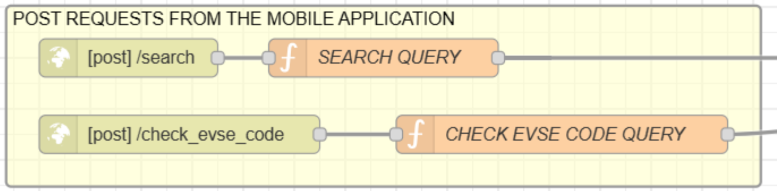
```
// manage the received commands
mqtt_client.on('message', (commands_topic, payload) => {
    let received_payload = JSON.parse(payload);
    let command = received_payload['command'];
    let evses = input['evses'];
    let evse_id = received_payload['evse'];
    aux.log("Received command "+received_payload.command

    // start command
    if (command === "start") {
        // check if the evse can be started
        if(!aux.isInMaintenance(evses, evse_id) && aux.i
            // update the evse to busy and publish its s
```

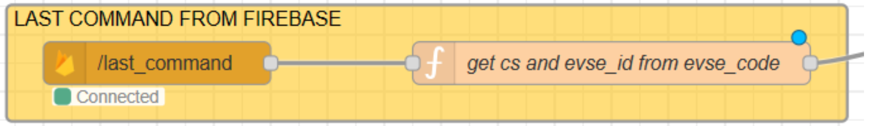
```
[04-01 17:58:42] Evse kkk executing charge
[04-01 17:58:43] Count on kkk has been interrupted due to busy or maintenance state
[04-01 17:58:46] Received command maintenance from the commands topic /sqbf/commands
[04-01 17:58:46] Evse kkk is in status maintenance
```

COMMUNICATION INFRASTRUCTURE

- From CS simulator to CSMS, and vice versa, via MQTT
- From mobile app to CSMS, and vice versa, via HTTP for reading operations
- From mobile app to CSMS, and vice versa, via HTTPS for writing operations

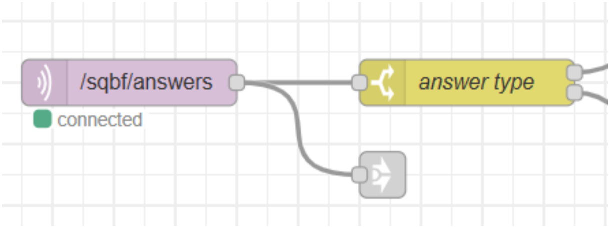


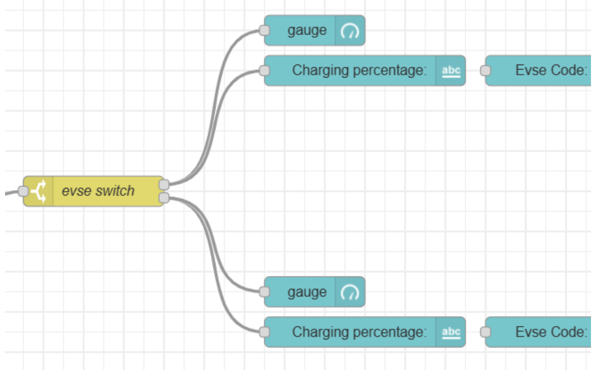


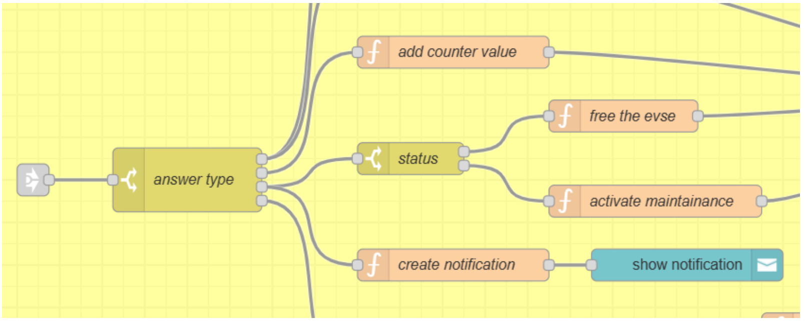


MIDDLEWARE

- One for each EVSE
- It redirects the CS answer to the right UI of the dashboard and to the CSMS

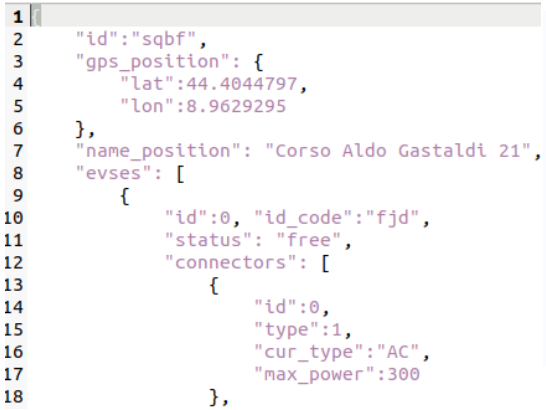




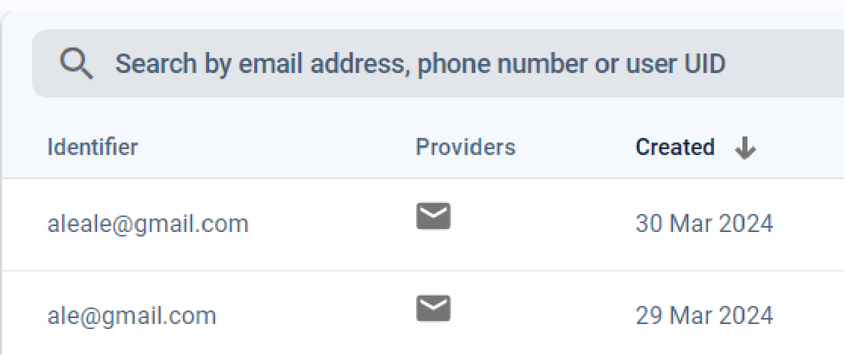


STORAGE

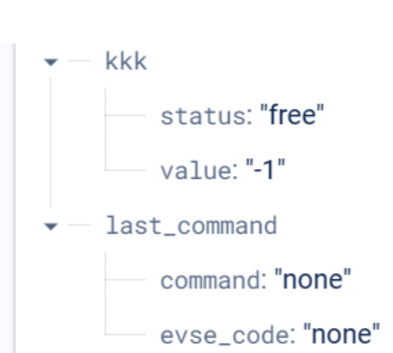
- EVSEs status memorized in a JSON file, one for each CS
- Authentication info in Firebase Authentication
- Current user interaction with EVSEs in Firebase DataBase
- Metrics data in a sqlite DataBase



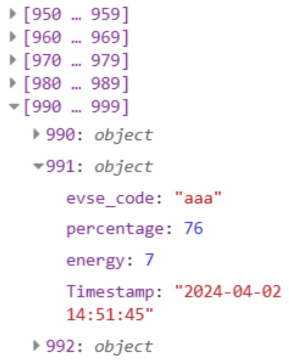
```
{
    "id":"sqbf",
    "gps_position": {
        "lat":44.4044797,
        "lon":8.9629295
    },
    "name_position": "Corso Aldo Gastaldi 21",
    "evses": [
        {
            "id":0, "id_code":"fjd",
            "status": "free",
            "connectors": [
                {
                    "id":0,
                    "type":1,
                    "cur_type":"AC",
                    "max_power":300
                },
```



Identifier	Providers	Created ↓
aleale@gmail.com	✉	30 Mar 2024
ale@gmail.com	✉	29 Mar 2024



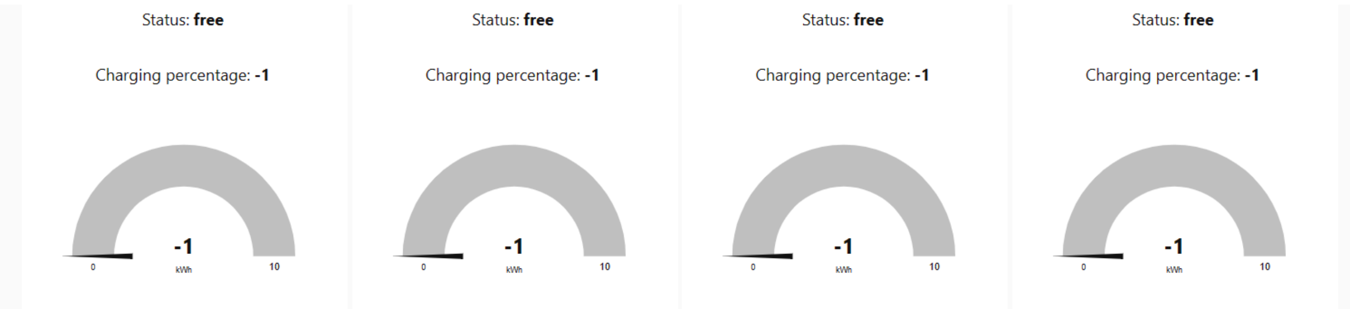
```
▾ kkk
    status: "free"
    value: "-1"
▾ last_command
    command: "none"
    evse_code: "none"
```



```
▸ [950 … 959]
▸ [960 … 969]
▸ [970 … 979]
▸ [980 … 989]
▾ [990 … 999]
  ▸ 990: object
  ▾ 991: object
      evse_code: "aaa"
      percentage: 76
      energy: 7
      Timestamp: "2024-04-02
      14:51:45"
  ▸ 992: object
```

DASHBOARD

- Made with node-red
- It tracks for each EVSE: status, charging percentage and energy
- It enables and disables the maintenance of the EVSEs
- It shows the position of the CSs



Status: **free**

Charging percentage: **-1**

Status: **free**

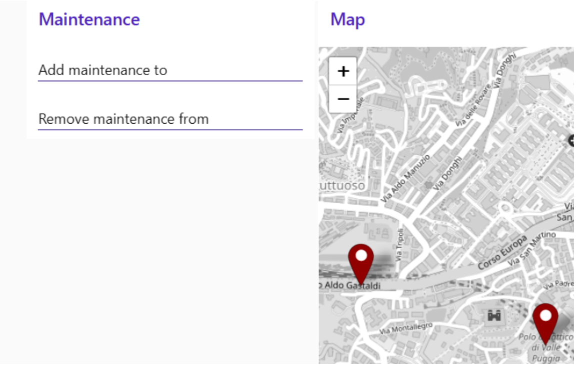
Charging percentage: **-1**

Status: **free**

Charging percentage: **-1**

Status: **free**

Charging percentage: **-1**



Maintenance

Add maintenance to

Remove maintenance from

Map

FUNCTIONALITIES

- Notification inside the app and the dashboard
- Nearest EVSE search inside the app
- Start, stop, info and maintenance commands over the EVSES triggered from the app and dashboard
- Counting inside the app
- Charging inside the app
- Other functionalities seen before as dashboarding, authentication and storage

DEMO