Scuola di Ingegneria Industriale e dell'Informazione
Dipartimento di Elettronica, Informazione e Bioingegneria

Design and Implementation of Mobile Applications

POLITECNICO
MILANO 1863

AlGa

Alessandro Falcetta
Gabriele Guelfi

Academic Year 2019–2020

Title: Design Document

Authors: Alessandro Falcetta, Gabriele Guelfi

Version: 1.0

Date: 10-May-2020

Download page: https://github.com/GabrieleGuelfi/Project_DIMA

Contents

Chapter 1

Introduction

1.1 Purpose

The aim of the present document is to describe the AlGa application with its services, a deeply definition of the main assumptions, the goals and a list of requirements, and the proposed solution. The definition of use cases and the scenarios will provide to highlight the features that the software has to offer to the customers and to better specify the boundaries system.

AlGa is an Android application developed within the course of *Design and Implementation of Mobile Applications* ad Politecnico di Milano, Italy. The goal of this course is to design a "mobile"applications by considering both the problem of designing the user experience, that is, the screens used to interact with the user, and the problem of understanding the actual distribution of the components that constitute the application and their interactions.

AlGa provides electric cars' owners with a simple way to find and use nearby charging stations. The main goal of our application is to help electric cars owner in the entire process of recharging their car. Being able to find charging stations tailored to everybody's needs is the key to meet that goal.

1.1.1 Goals

[**G1**] The application should provide the user with a **clear** overview of the nearest charging stations and their vendor.

[**G2**] The application should offer the user the possibility to order them, in a list, according to specific criteria: price, charging speed, distance, vendor.

[**G3**] The application should make it easy and straightforward to start the

navigation toward any charging station.

[**G4**] Users should be able to check their statistics about time spent charging their car, the total amount of money spent, etc. with the minimum interaction required; statistics can be enriched if the user indicate their owned car

[**G5**] The application should provide an easy way to save the profile of the user, with the possibility to log-in, log-out, delete the account and change personal information like e-mail and the owned car.

1.2 Scope

Electric cars are one of the most interesting technologies of the last years, with a possible bright future. Nonetheless, the public is still reluctant to invest money into this kind of products because of many concerns about the autonomy and the charging system. With respect to thermic engines cars, electric cars require more time to be recharged, have less autonomy in terms of distance and the infrastructure of recharging stations is yet to be completed.This is the context in which applications like AlGa can improve the experience of electric cars owners.

An easy and adaptive way find the best charging stations, according to everybody's needs, can be an important boost to the confidence in this technology.

1.3 Definitions, Acronyms, Abbreviations

Definitions

- **Health data:** some vital parameters of the users, for example heartbeat and blood pressure
- **Runner:** athlete who participate in a run
- **Fiscal code:** a code used in Italy that uniquely identifies every citizen
- **Third party:** a usr like an organization that is allowed to request health data
- **Individual request:** request of data of a single user

- **Group request:** request of data of a group of users, based on some parameters
- **118:** Emergency Medical Service, it's the number to call an ambulance
- **112:** Public-safety Answering Point

Acronyms

- **RASD:** Requirement Analysis and Specification Document
- **API:** Application Programming Interfaces
- **GPS:** Global Positioning System
- **HTTP:** HyperText Transfer Protocol
- **TLS:** Transport Layer Security
- **OS:** Operating System
- **SQL:** Structured Query Language

Abbreviations

- [**Gn**]: n-th goal
- [**Dn**]: n-th domain assumption
- [**Rn**]: n-th functional requirement

1.4 Document Structure

This design document is composed by 6 chapter: The document is organized as follows: Chapter 2 states what the app is about and which are the requirement is must satisfy. Chapter **??** explains the reasons behind the choice of the used technologies. Chapter 3 details the architectural choices for the entire application, while Chapter **??** gives an overview on the application from the point of view of the user, including some screenshots and mock-ups. Chapter 5 explains the use of external APIs, while some business-driven considerations are proposed in Chapter 7. Conclusions are finally drawn in Chapter 8.

Chapter 2

Idea and Requirements

2.1 Product perspective

A simple diagram of the common use of AlGa is here provided:

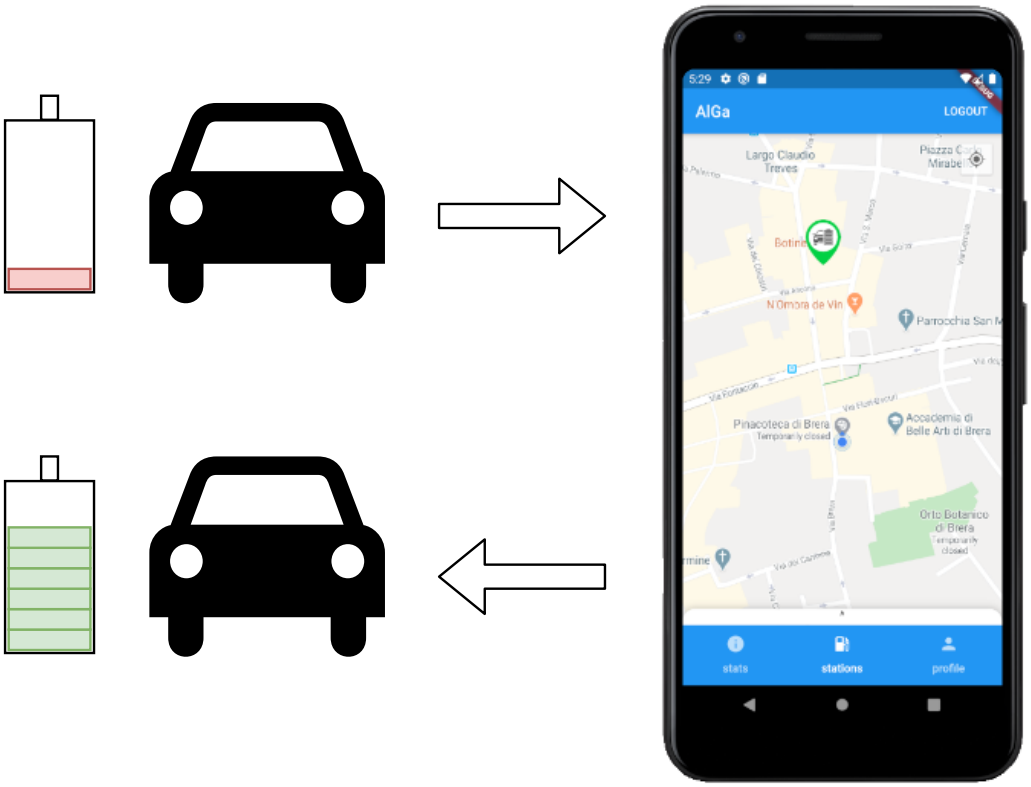

Figure 2.1: Use diagram of AlGa.

As we can see, AlGa is designed to help users directly on the road, in the fastest and easier way possible. Statistics useful for the users are another key part of the application: in fact, during the use, a set of useful metrics are collected and then displayed to the user. They can help her or him understand better the way in which they use their car, and fix some problems and/or wrong behaviors which may arise in the process of switching from a traditional car to an electric one. However, they are not strictly necessary to use the app; users can decide to opt-in to this functionality.

2.2 Product functions

According to the goals of the project, a more detailed description of the various functionalities is here provided:

Stations

This is the default screen. It shows a map, centered on user's position, together with the charging stations. The user can select every station to see its properties like cost, position, vendor, etc (Requirement 1). If the user decides to utilize that station, a simple click on the "GO" button will open the Google Maps application, with the destination already defined on that station (Requirement 3). Moreover, with a simple scroll menu, the user can also visualize a list of the nearest stations, with the possibility to order them by price, distance and speed and vendor (Requirement 2).

Stats

The statistics screen provides the user with data about the use of their car and of the application. That is, the amount of time spent at charging stations, the amount of money spent in energy, the distance traveled and other aggregate information (Requirement 4). Leveraging the possibilities offered by sensors installed on every smartphone, AlGa can collect these statistics in an effortless way for the user.

Profile

The profile screen lets the user customize their account on AlGa. They can choose a simple username, change their e-mail and password, and select their electric car. AlGa offers a list of cars from which the user can choose; every

car has some statistics about the consumption, the autonomy etc. This leads to more accurate usage data (Requirement 5).

2.3 User characteristics

AlGa User: An individual who has downloaded AlGa.

Registered User: An AlGa User who has created an account on AlGa platform.

Vendor: A company which offers electric recharging stations.

2.4 Assumptions, dependencies and constraints

Domain assumptions

[**D1**] Users can be identified through a couple email/password, unique for every user.

[**D2**] Users' devices can provide precise and correct data on location.

Chapter 3

Architecture

The architecture for the application is composed by the user's device and a server, hosted on Google Firebase. In this diagram the two main components

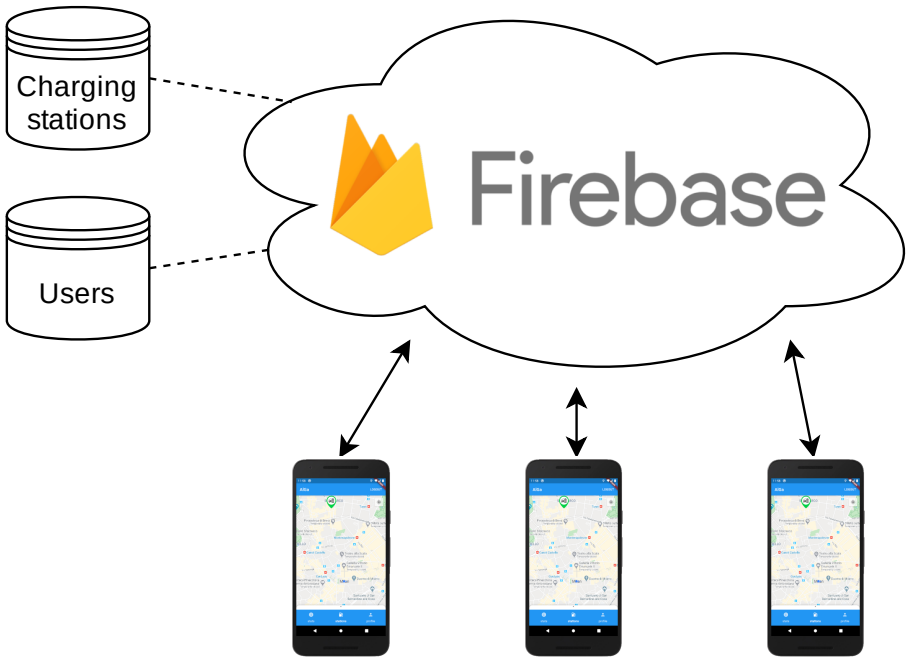

Figure 3.1: Basic architecture of the application.

are shown:

- Mobile app: the user can interact on their device in order to retrieve the recharging stations and their data, along with the functionalities cited in2.

- Firebase: Google Firebase offers a simple way to blabla

3.1 Use case diagrams

Name	Sign up
Actor	User
Entry conditions	User has AlGa app installed on their smartphone
Events flow	1. Click the "Sign Up" button 2. Fill the form providing the requested information 3. Click the "Confirm button" 4. User is now enrolled to AlGa.
Exit conditions	The app shows the main screen to the user.
Exceptions	1. The e-mail is already taken. 2. The format for e-mail or password is wrong. 3. All the exceptions are handled by notifiying the user and taking him back to the main screen.

(a)

Name	Login
Actor	User
Exit conditions	The app shows the main screen to the user.
Events flow	1. The user opens the app 2. The user fills the “E-mail” and “Password” fields 3. The user clicks on the “Login” button
sxit conditions	The app starts to show the collected data to the user
Exceptions	1. The e-mail is not associated with a registered user 2. The password is not correct for the given e-mail 3. All the exceptions are handled by notifiying the user and taking him back to the main screen.

(b)

Chapter 4

Design

AlGa is implemented in Flutter.

Chapter 5

External services and Libraries

We used many external services and libraries, some of them are useful to implement the core of the application, such as Firebase, while others just enhance the user experience.

Integrating these external services we didn't need to reimplement that functionalities by our own and since they are specialised in their field their efficiency is much higher.

5.1 Firebase

5.1.1 Authentication

This service of course represent the main entrance of the application since it handles the authentication process of users. It was very simple to implement because it requires just client-side code and it allows users to register or login with the classic email and password method or GitHub OAuth. The server-side of the application simply verify that the credential inserted are corrected and allows the user to log in.

5.1.2 Database

The Database provided by Firebase is of the NoSQL type and has several advantages: data can be updated directly in th Web Console or through a simple program, Flutter provides simple APIs to interact with it asynchronously and everything is stored in the Firebase's cloud.

We use it to store different type of cars, charging stations with their location and properties, users with their saved settings(i.e. name and car) and all the recharges made.

5.1.3 Storage

We use this service just to store the profile pictures of each users that eventually uploaded it.

5.2 Google Cloud Platform

5.2.1 Google Maps

It is probably the most important feature of this application, it allows to see and select the charging stations, visualised with a custom pin, check the own position and navigate on the map. Using Google Maps keeps the UI simple for the users because most of them already use it.

The navigation point-to-point is kept outside the application because the free plan doesn't permit that feature.

5.3 Minor Packages

- **image_picker**: it allows to upload images from the smartphone gallery, specifically for users that want to upload their profile pictures
- **email_validator**: it is used to validate the email inserted by a user during registration process
- **url_launcher**: it launches an external URL, we used it to launch the point-to-point navigation in Google Maps
- **sliding_up_panel**: we implemented a sliding up panel to visualise the charging stations by different order
- **flutter_local_notifications**: it is used to display push notification when a recharge is finished
- **background_locator**: it gets location updates even when the application is killed, it's useful to check if the user has arrived at the selected charging station

Chapter 6

Test Cases

Chapter 7

Cost estimation

Our plans made sure to keep into account all possible costs that our team will incur during development. As such we planned to recruit new team members only when needed in order to keep costs as low as possible. Our first research phase won't be very expensive and it will be mainly led by our core team. The main source of cost will be the recruitment of new team members as well as the planning and launching of the marketing campaign in order to ensure a successful launch, which we think is paramount to the long term success of the entire project. Technology wise the infrastructure won't have a great cost in the development and beta phase and we plan to scale according to the demand, in such a way we can be sure to avoid scaling over the demanded size and use the budget for possible extensions of the product or improving upon the existing services offered.

Role	Number	Estimated Cost (monthly)
Core Team Member/Chief	4	2,000€
Senior Project Manager	1	2,000€
Junior Back-End Developer	2 to 3	1,500€
Junior Front-End	1 to 2	1,500€
Security Analyst	0 to 1	2,000€
Junior Graphic Designer	1 to 2	1,200€
Marketing Supervisor	1	3,000€
Social Media Manager	1	1,500€
Digital Marketing Manager	1	1,500€

Chapter 8

Conclusions

Name	Alessandro Falcetta
Effort spent	20 hours
Task	• Group work • Introduction • Deployment view • Runtime view

Name	Gabriele Guelfi
Effort spent	20 hours
Task	• Group work • Overview • Component interfaces • Requirements traceability • UML