Final Presentation
Advanced Distributed Systems Project

Team: **Ctrl_Alt_Elite**
Members:

- Mergea Sergiu
- Marius Panduru
- Pop Tudor
- Pop Adrian (adrian.pop2@student.upt.ro)

GitHub: https://github.com/tudorrpop/ctrl_alt_elite (public)

Trello: https://trello.com/b/N1t9sIdN/ctrlaltelite (public)

General Description & Features

This distributed application is concieved to offer a complete car parking management solution across multiple stores of the same brand in a city. The application is designed to serve both driving users and store administrators.

The first category of users are going to use a mobile solution where they will register and login into an application and utilize a digital generated card to enter the parking space. The mobile component will have an AI module integrated. This AI model will be capable of offering responses to the users's questions, related to both the store parking lots and their availability and it will be capable of making predictions based on the history data. The interactions will take part through a chat interface and the customers will have the possibility to manage their own account.

A web component will also be available for the store administrators. They will be able to see the live state of the parking lots and several statistics regarding the traffic matters, as well as administrating all the stores and all the customer accounts. The administrators will be notified in this application regarding sepcific actions in the system.

Each core component will be hosted inside a Container App, in an Azure Container App Environment. An important module named worker will be added and there will be created more workers instances, one for each store. Each of them will be destined to collect the data from the hardware / IoT devices such as sensors that are supposed to be mounted in the parking lots. This route will also be crossed by the QR code on the card used by customers in the mobile module. The database will, of course, be hosted on a cloud server offering security and facile communication with the backend.

Technologies to be used

→ **Servers:** Azure Services

→ **Backend:** Spring Boot (Java)

→ **Frontend:** Angular + Ionic (TypeScript)

→ **Database:** MySQL

→ **AI Technologies:** Llama (llama3)

→ **JSON Web Tokens**

Deployment Platform

The platform on which we intend to deploy is **Azure.** Even though the decision is not 100% made, the development process was somehow led by this option.

Description of the separated Entities

When an admin will create a new store (using the **Web Graphic Interface**) by introducing the necessary data, a new **Worker** will automatically be created for this store. The Worker represents an abstraction of a store (or of an instance), and it is responsible for actioning the barrier from the parking lot and for storing information (in a local database) about the state of the parking lots and other traffic information regarding the store it abstracts.
The Workers will communicate throughout an **Orchestrator** using HTTP endpoints. This communication is bidirectional, and the Orchestrator will provide the requested data for both the previously mentioned web module (dedicated to the admins) and also for the **mobile application** designed for the users and the interactions with the physical systems such as the barrier and the parking lot sensors (via QR codes).

Teamwork

The project has been characterized so far as truly teamwork experience. We believe that a clear monitorization of the effort is hard to be expressed because the design & architecture were created together and the problems that appeared were also faced together. While the Worker module was mostly Marius's responsibility, the Backend (Orchestrator) module was Sergiu's responsibility, the Web frontend was Tudor's priority and the Mobile frontend fell on Adrian's shoulders, all the members contributed on all the modules due to the communication within the entire system.