

eMALL system

Software Engineering 2 - RASD & DD Pitch
Irfan Cela, Mario Cela, Alessandro Cogollo



POLITECNICO
MILANO 1863

Overview

Private mobility: fast changing market
23 million e-cars on the road in 2030
eMALL aim: simplify charging process



Users Description



CPO Charging Point Operators



Registered EV Driver



Unregistered EV Driver



POLITECNICO
MILANO 1863

Charging Point Operators - CPO

owns one or more charging stations
manages bookings and promotions
buys energy from DSOs
has their own IT system



Unregistered EV Driver

anybody who owns an electric vehicle, but isn't registered in the eMALL system. Before accessing its benefits, it needs to get an account

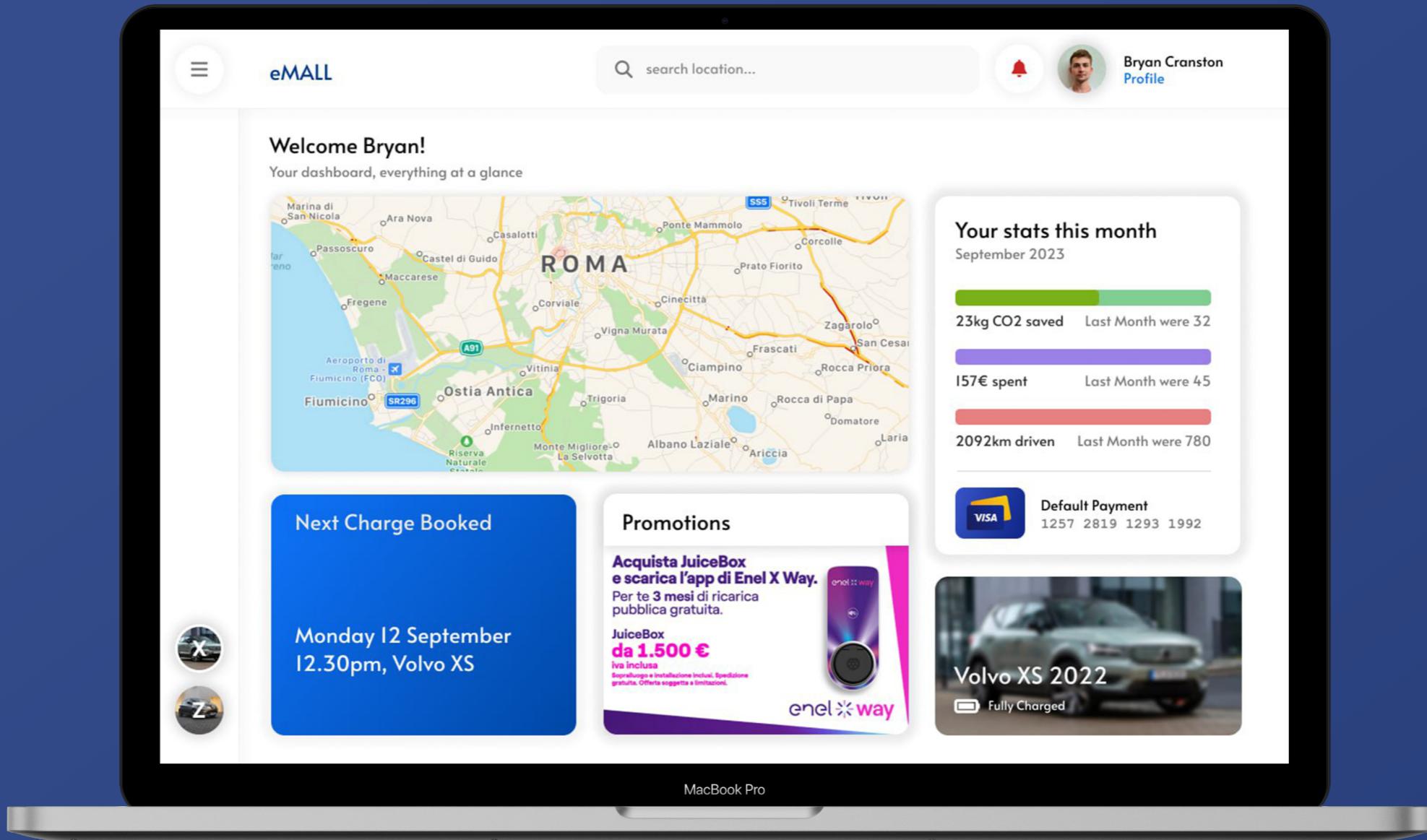


Registered EV Driver

an electric vehicle owner who already joined the eMALL system, and access its benefits. He's identified with a unique ID, and can own one or more vehicles with different specifics. They can check prices and position of charging points, in addition to receiving notifications about promotions reserved to them.

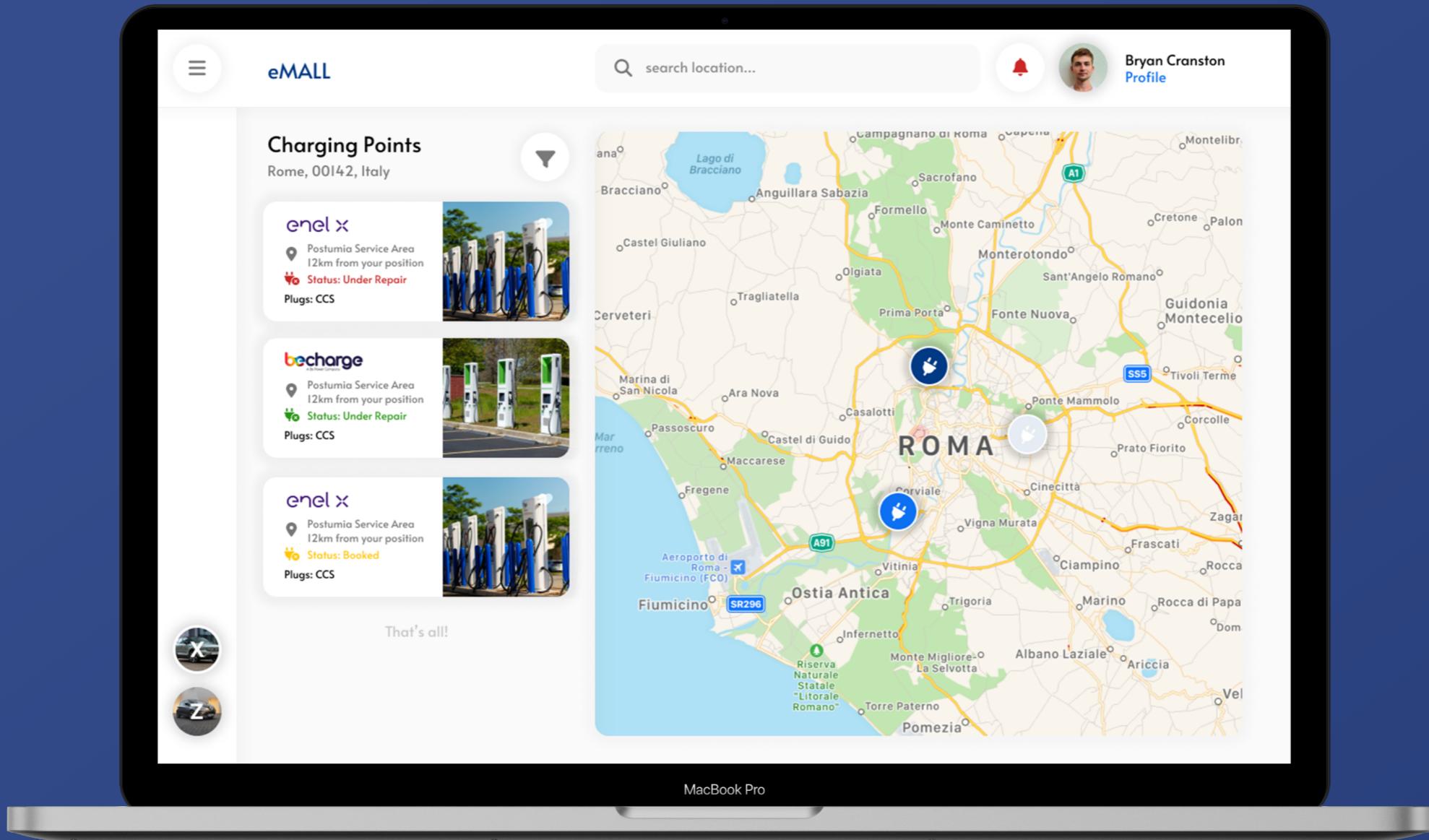


EVD Goals



The EVD can charge his EV (G1) and search for special offers provided by CPO (G2)

G1 - Map

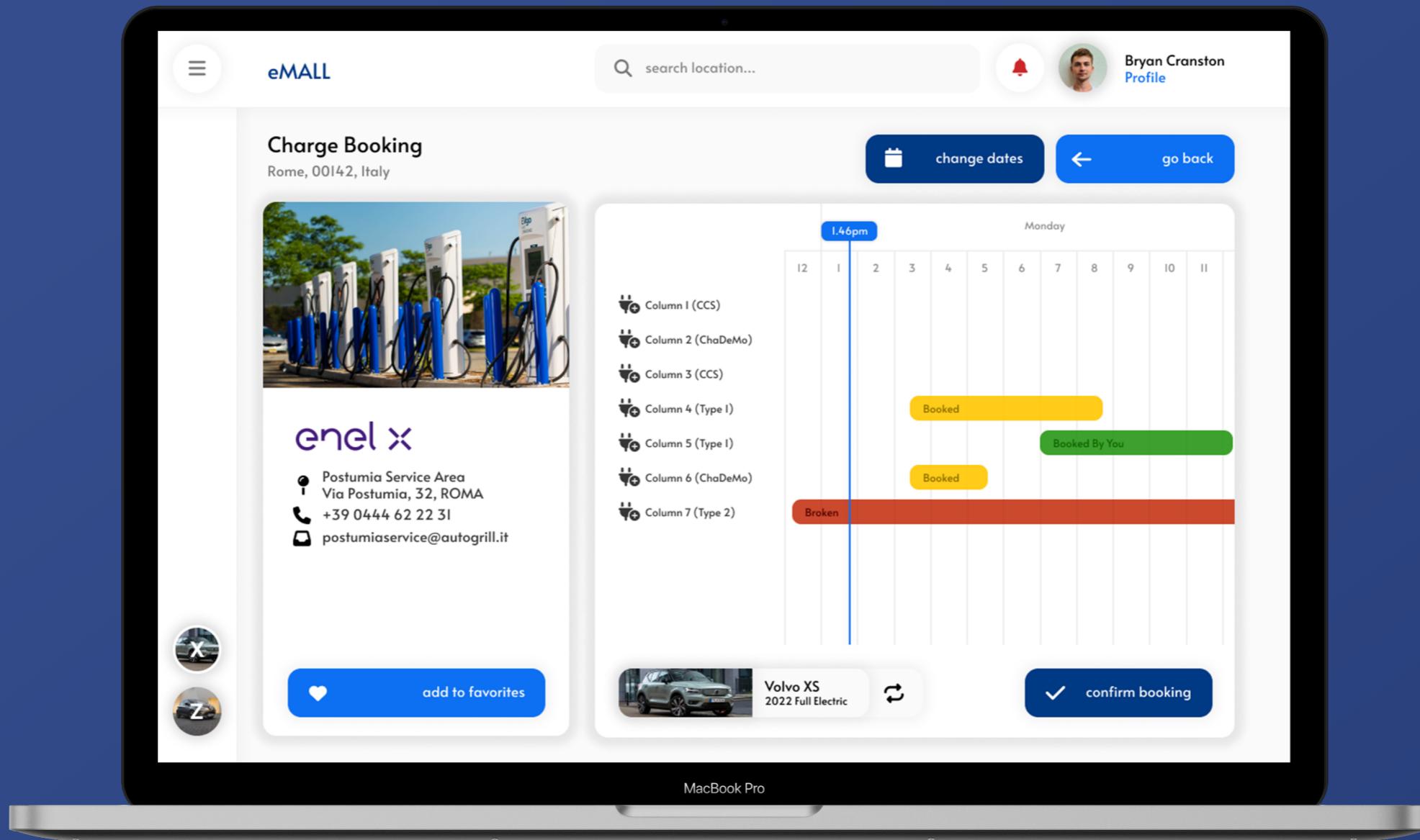


The EVD can get information about charging stations

09.

G3 - Booking

RASD

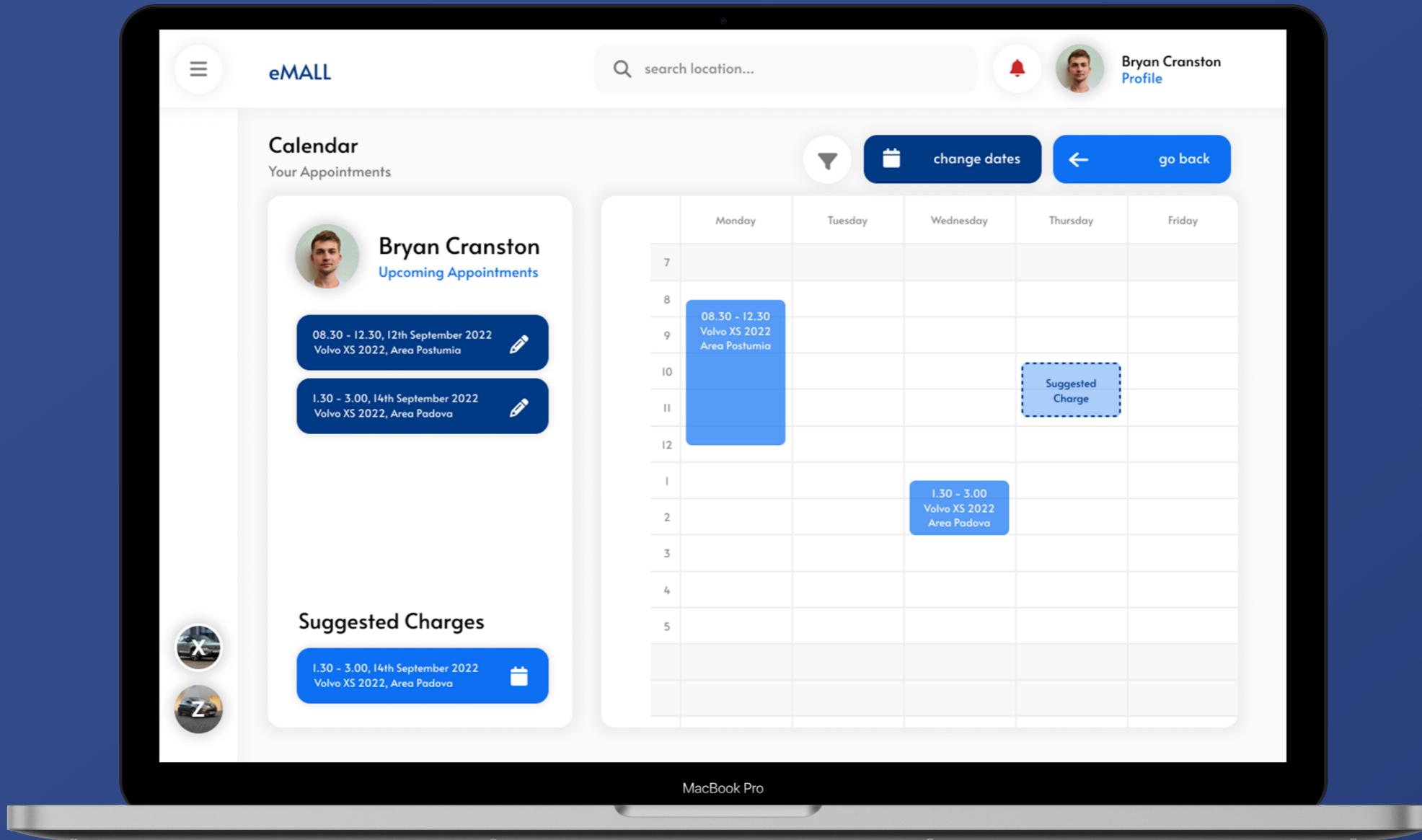


The EVD can book a charge for his EV at a charging station for a specified time frame

10.

G5 - Calendar

RASD

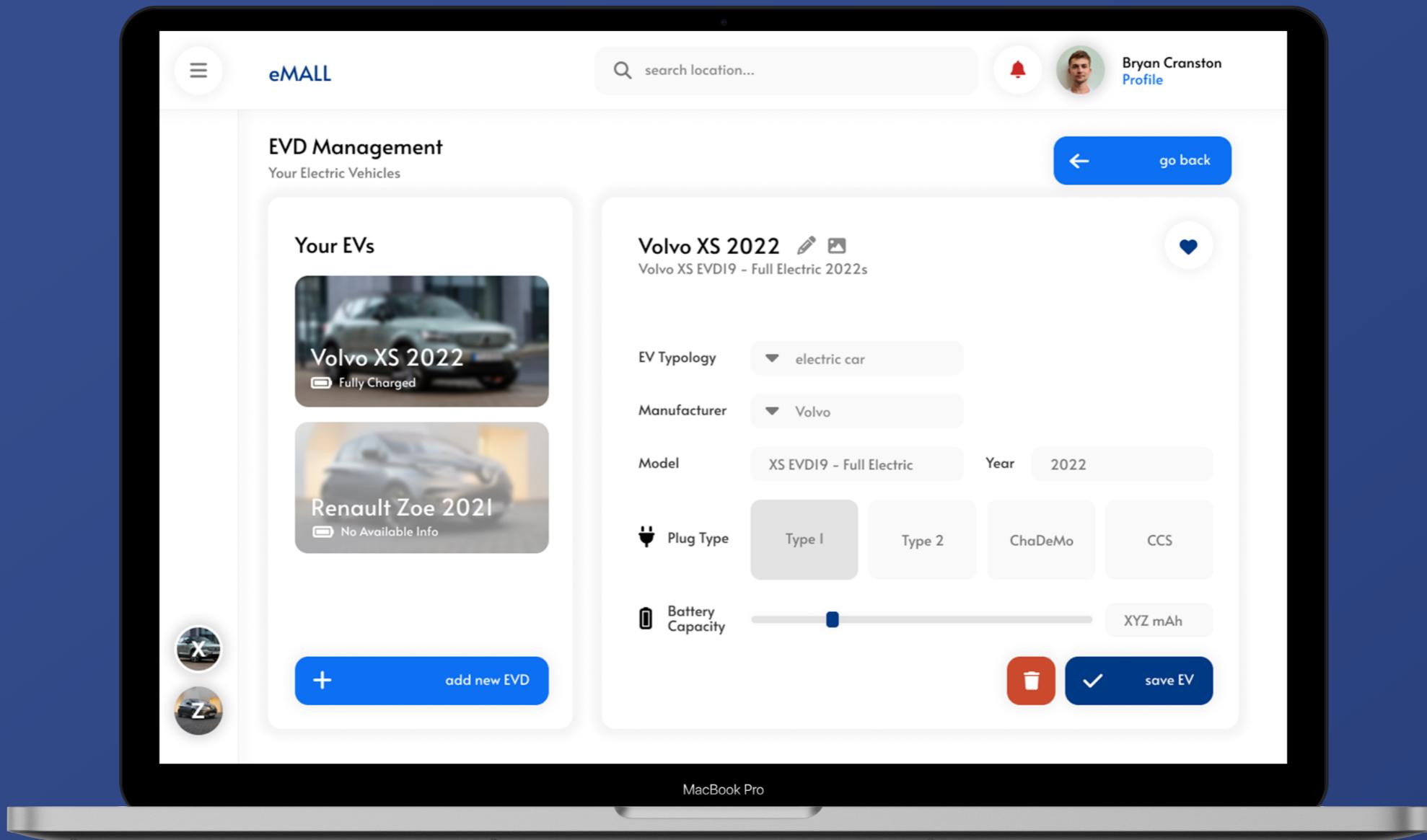


EVD can manage activities
thanks to a calendar

11.

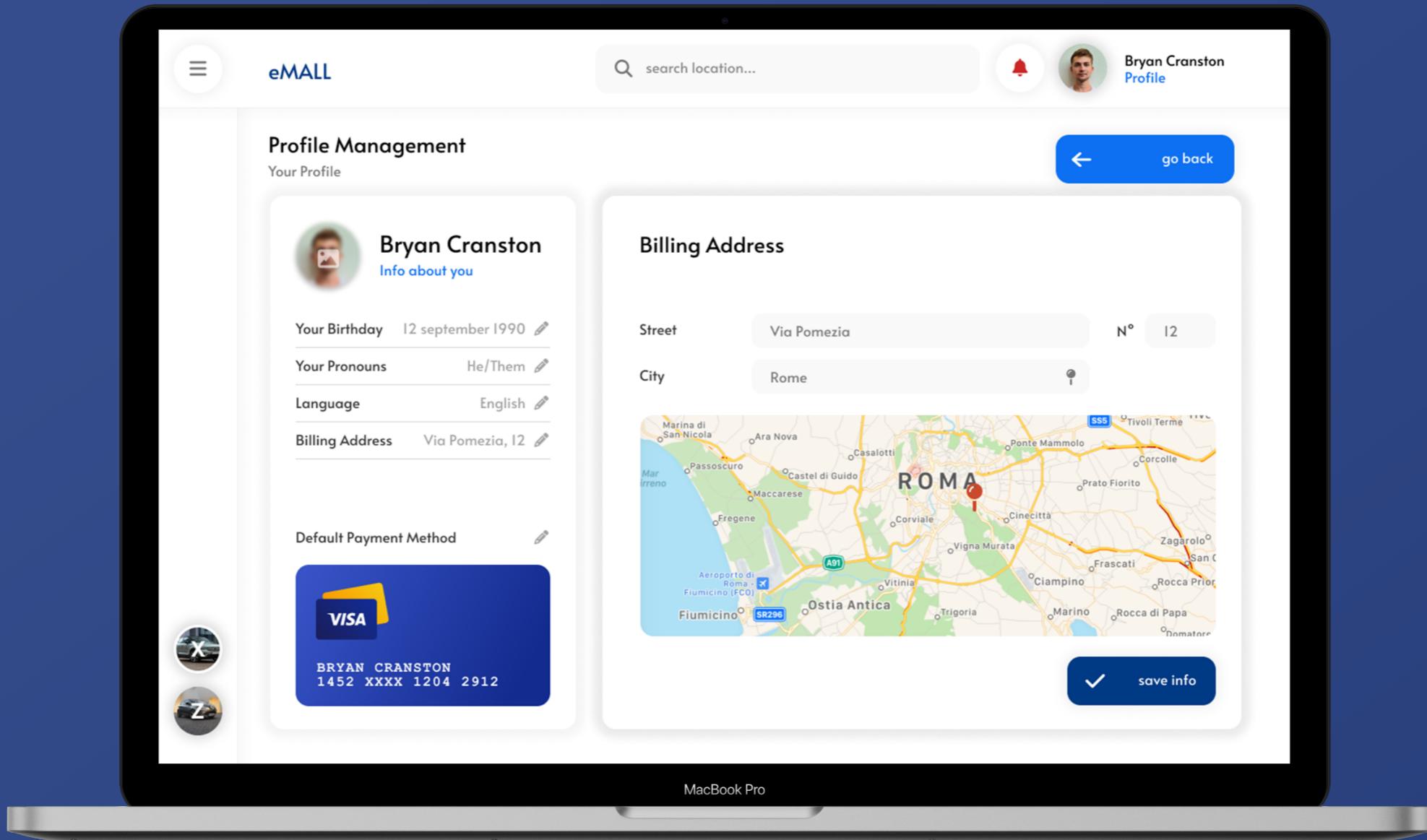
Other Features

RASD



Vehicle Management

Other Features

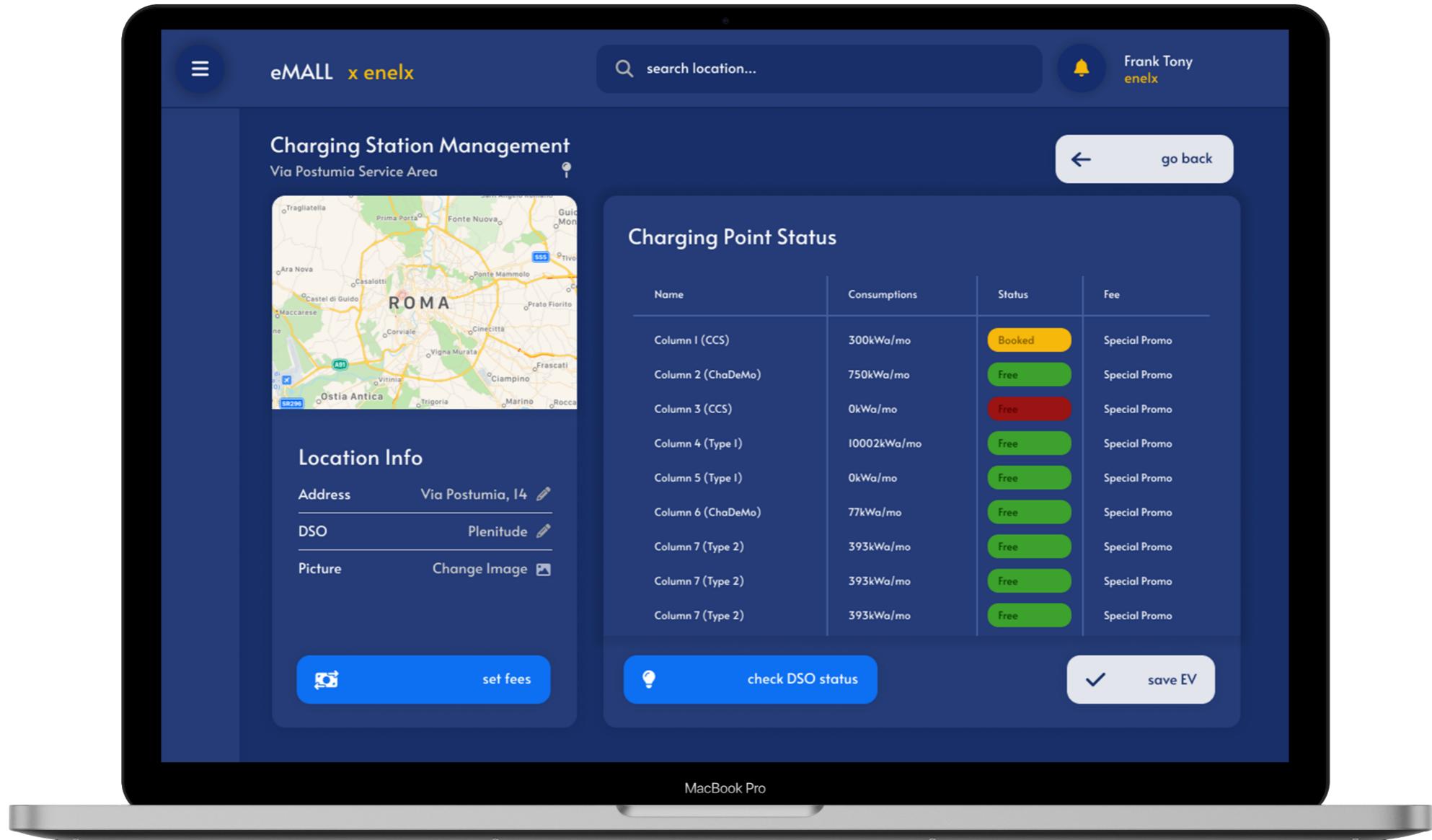


Profile Management

13.

G6 - Station MGMT

RASD

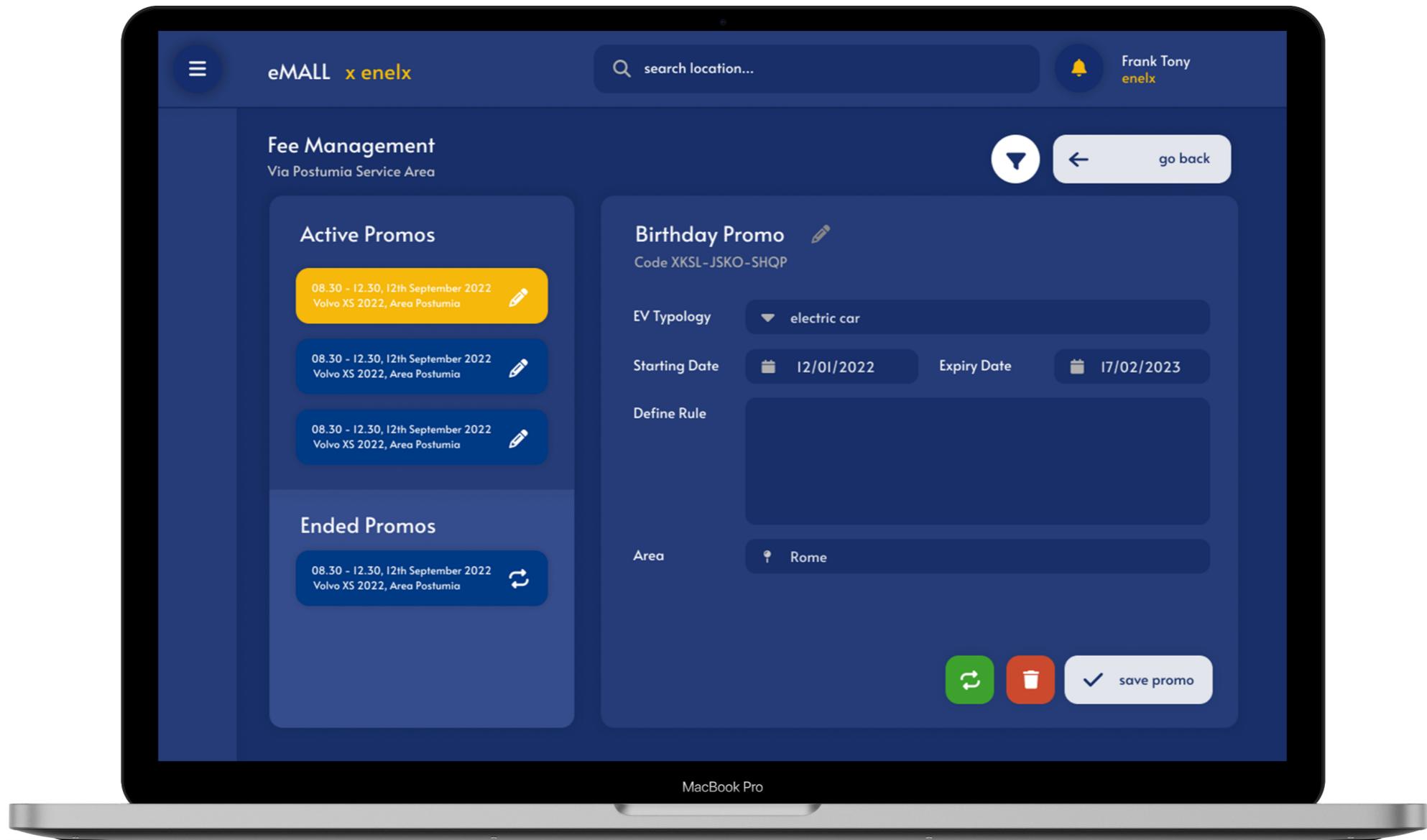


The CPO can manage its charging stations and its charging points

14.

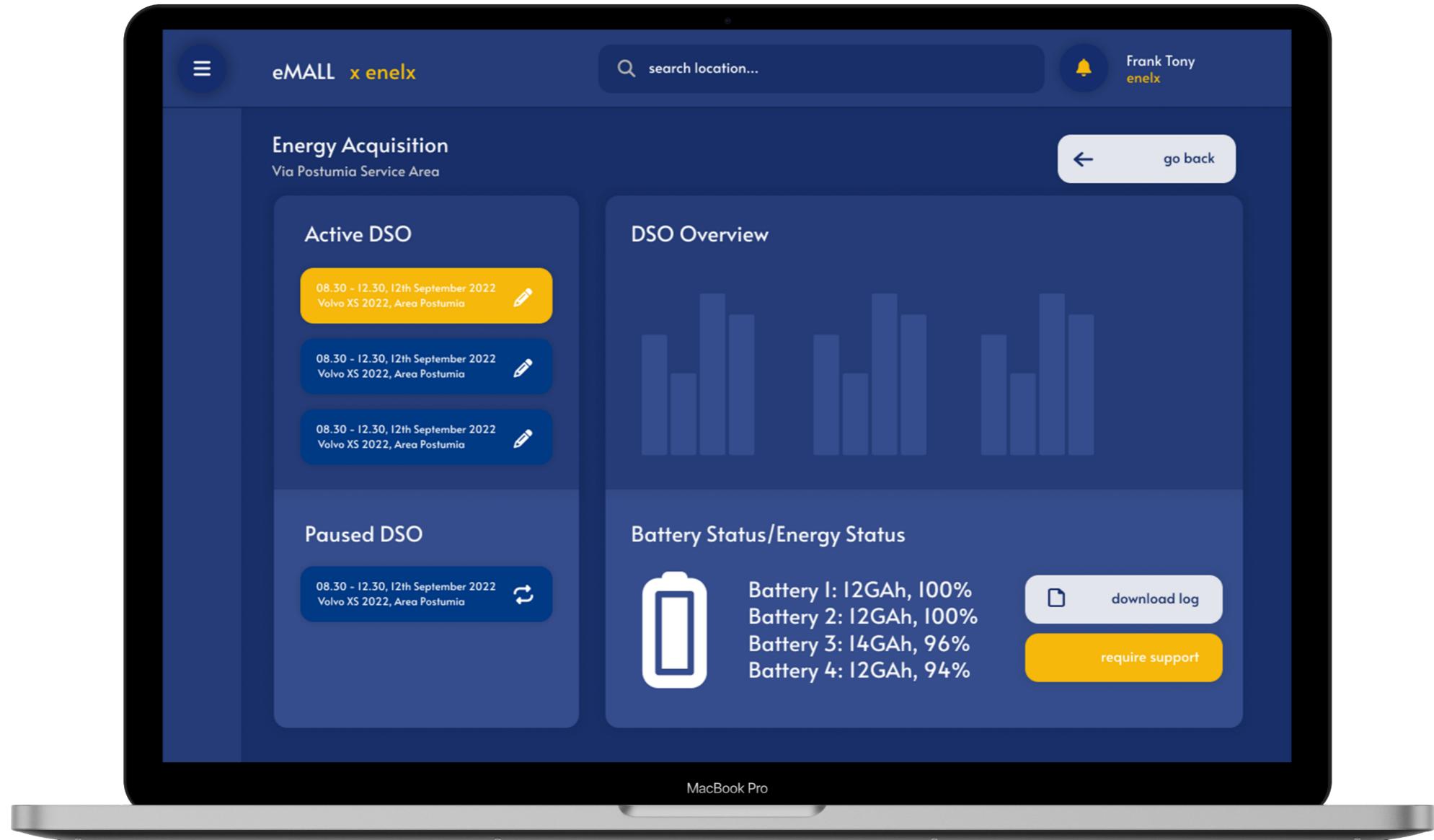
G8 - Promo MGMT

RASD



The CPO can manage its promotions

G9 – DSO MGMT



The CPO can manage
the electricity supply

Client-Server Architecture



Presentation, Business and Data **layers are separated**



Maintenance and updates are facilitated



Client is **lightweight** and **easy to use**



More **security** and **integrity** of data



Microservices Architecture



Client Application sends a HTTP request to the microservice that implements that functionality



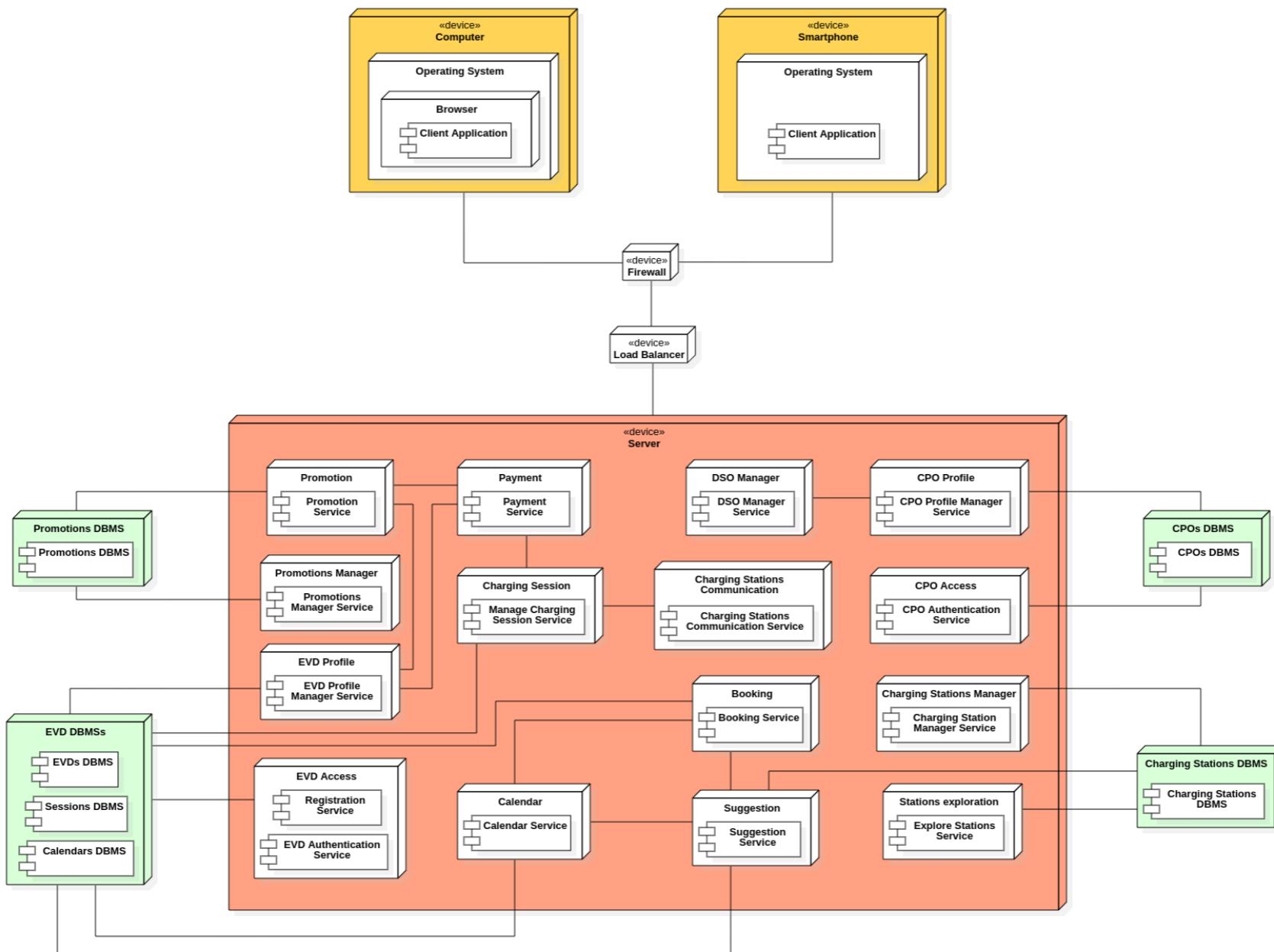
Technology heterogeneity: most suitable programming languages and tools



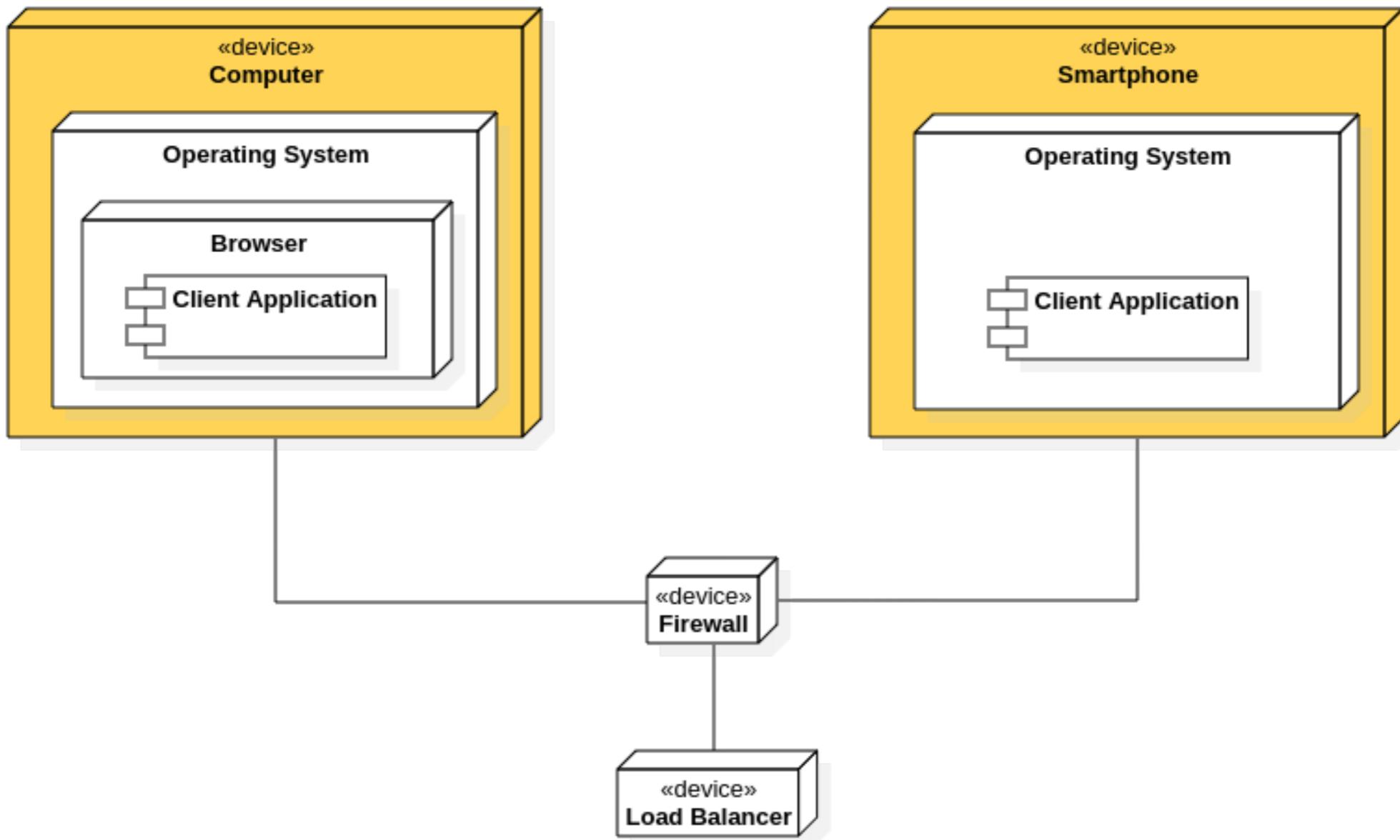
Availability: if a service fails, the system keeps running



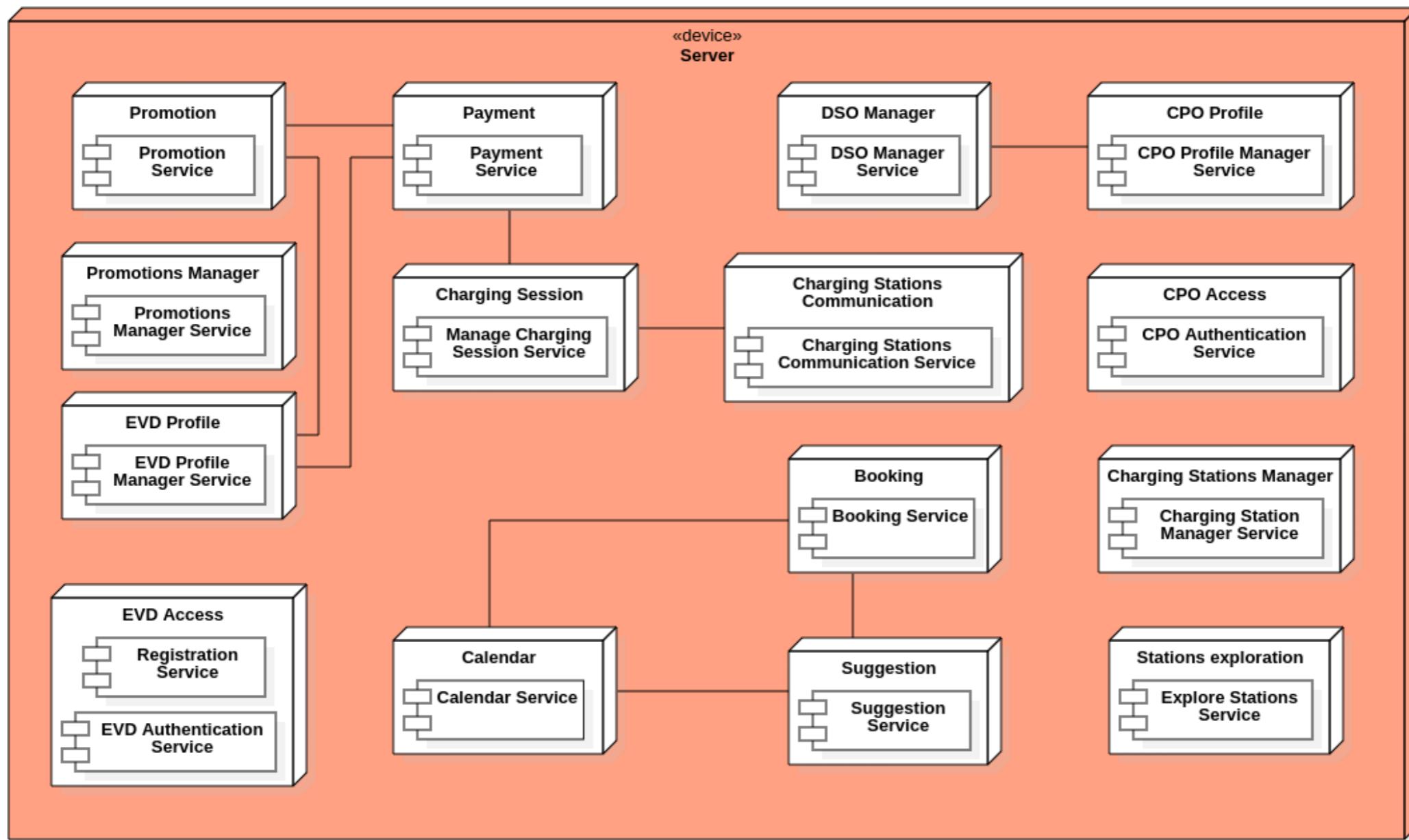
Deployment Diagram



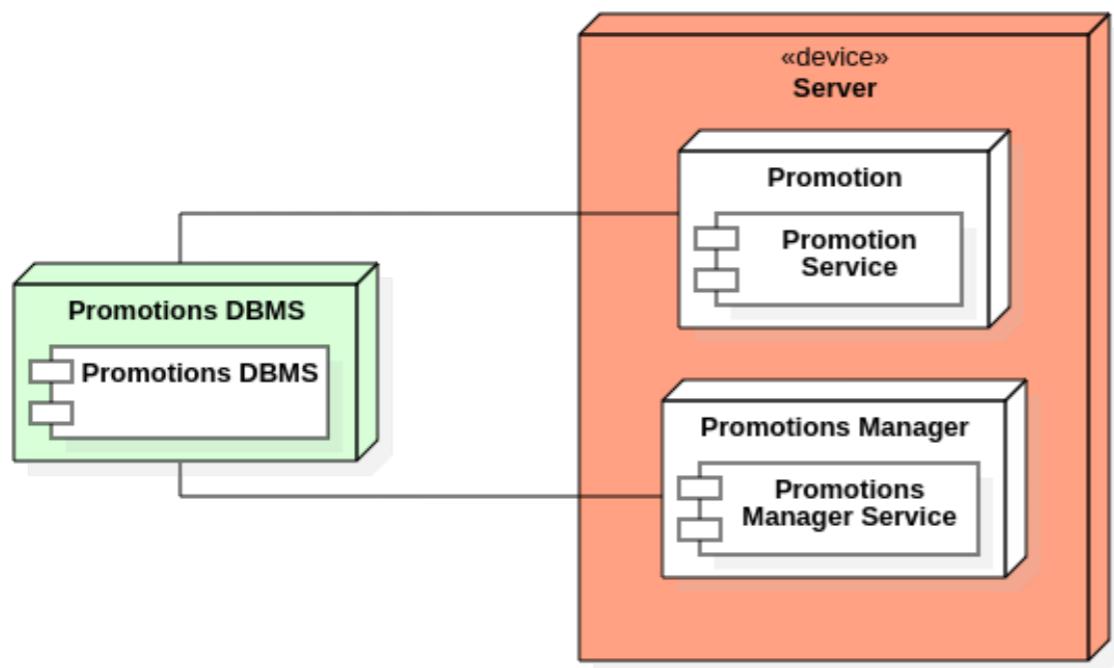
Deployment Diagram



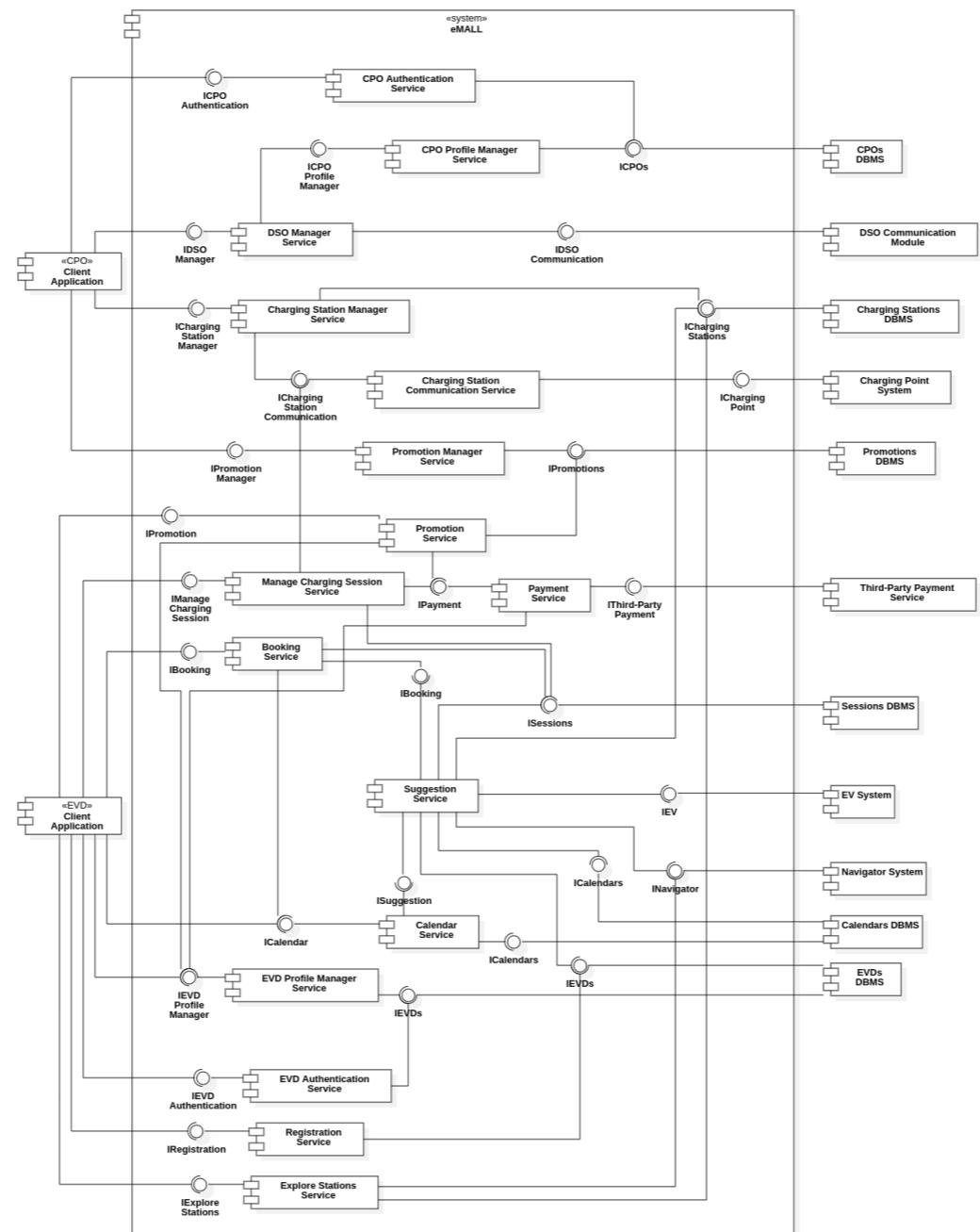
Deployment Diagram



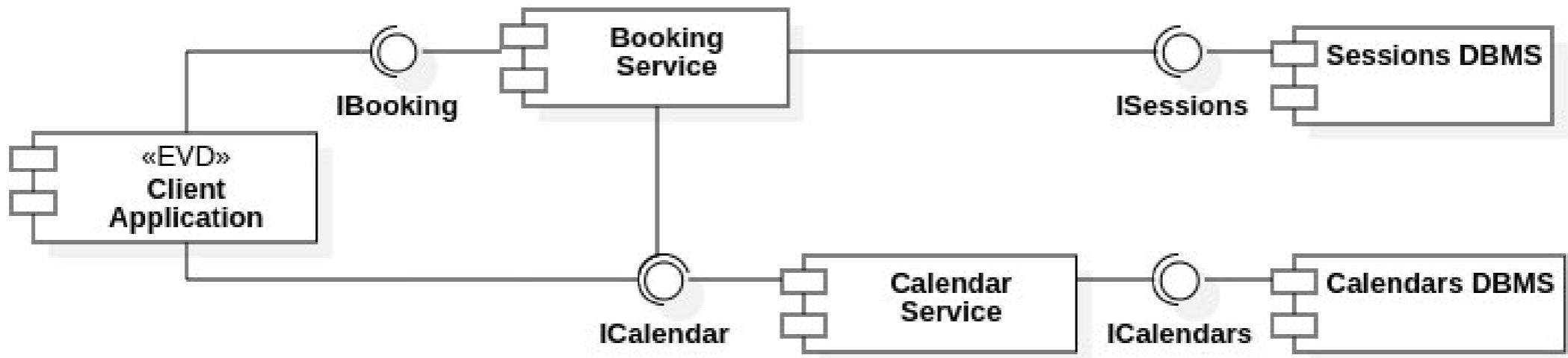
Deployment Diagram



Component Diagram



Component Diagram



Thanks!