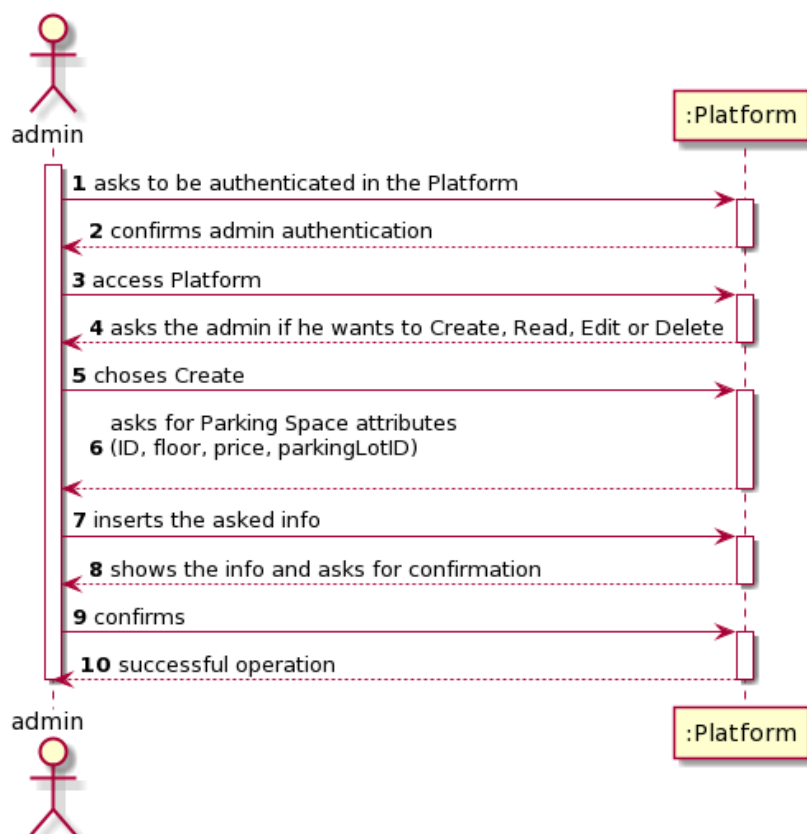


UC1 - Parking Spot Management

Brief

The admin accesses the platform and authenticates in it. The platform asks what action the admin intends to perform (Register, Consult, Edit, Delete). The admin chooses to register a parking space. The platform asks the admin to fill the required data (single designation, floor, price, car park name). The admin inputs the requested data. The platform shows the data to the admin, asking him to confirm it. The admin confirms. The platform saves the data and informs the admin that the operation was successful.

SSD



Fully Dressed

Name: Parking Spot Management

Primary Actor: Admin

Stakeholders and interests:

- Admin: Ability to manage parking spots and relate them to future reservations.
- Parking Lot: Have the Parking spaces full list with their info updated.

Preconditions:

- The Parking Lot to which the Parking Place belongs must exist in the database whether it is public or private.

Postconditions:

- The Parking Lot is updated, according to the Parking Spots that were added by the Admin.

Main Success Scenario:

1. The admin starts the authentication operation.
2. The Platform allows the admin to be authenticated.
3. The admin accesses the platform.
4. The platform asks what action the admin intends to perform (Register, Consult, Edit, Delete).
5. The admin chooses to register a parking space.
6. The platform asks the admin to fill the required data (single designation, floor, price, car park name).
7. The admin inputs the requested data.
8. The platform shows the data to the admin, asking him to confirm it.
9. The admin confirms.
10. The platform saves the data and informs the admin that the operation was successful.

Extensions (alternative scenarios):

***a.**

1. The Admin can't authenticate on the platform.
2. The use case ends.

***b.**

1. The admin asks for the cancelation of the registration of the parking space.
2. The use case ends.

5a. The admin chooses Consult, Edit, Delete but there are no registered parking spots

1. The platform asks if the Admin wishes to Consult, Edit or Delete a parking spot.
2. The Admin chooses an option.
3. The platform asks which parking spot the admin is interested in.
4. The Admin inserts the id of the parking spot to be searched.
5. The platform informs the admin that the parking does not exist.
6. The use case ends. Starts again on step 1.

5b. The admin just wants to consult the data

1. The admin chooses Consult.
2. The platform shows the list of the registered parking spots.
3. The admin selects a parking spot.
4. The platform shows the selected parking spot info.
5. The use case ends.

5c. The admin wants to edit a registered parking spot.

1. The admin asks to edit a specific parking spot.
2. The platform shows the list of all the parking spots that were registered.
3. The admin picks one of the parking spots of the list.
4. The platform shows the chosen parking spot info.
5. The admin edits the info as it wishes.
6. The platform confirms the edited info.
7. The admin confirms.
8. The platform registers the info and the operation is successful.
9. The use case ends.

5d. The admin wants to delete a parking spot

1. The admin selects Delete.
2. The platform shows the list of all the parking spots that were registered.
3. The admin selects a parking spot.

4. The platform shows the selected parking spot info and asks the admin to confirm the action.
5. The admin confirms he wants to delete that parking spot.
6. The platform deletes the selected parking spot info.
7. The use case ends.

8a. The platform detects that the parking spot Unique Designation already exists.

1. The platform alerts the admin to that fact.
2. The admin does not change the entered Unique Designation.
3. The use case ends.

Special Requirements:

Variations in Technology and Data: N/A

Frequency of Occurrence: Whenever a parking place needs to be registered.

Miscellaneous:

Class Table

Main Flux	Question: Which class(es)...	Answer	Reason
1.The admin starts the authentication operation.	...is responsible for the authentication?	JWTToken	
	...interacts with the admin?	WebAppUI	
2.The Platform allows the admin to be authenticated.	...shows the admin that the authentication was or not successful?	AuthenticationService (?) / AuthenticationController (?) WebAppUI	

3.The admin accesses the platform.	...interacts with the admin?	WebAppUI	
4.The platform asks what action the admin intends to perform (Register, Consult, Edit, Delete).	...is responsible for the Register action?	WebApp	
	...is responsible for the Consult action?	WebApp	
	...is responsible for the Edit action?	WebApp	
	...is responsible for the Delete action?	WebApp	
	...who coordinates the ParkingSpot methods?	ParkingSpotController	
5.The admin chooses to register a parking space.	...is responsible for creating ParkingSpot instances?	WebApp	
6.The platform asks the admin to fill the required data (single designation, floor, price, car park name).	n/a		
7.The admin inputs the requested data.	...saves the input data?	WebApp ParkingSpot	IE: ParkingLot has ParkingSpot
8.The platform shows the data to	...who validates the ParkingSpot data? (local validation)	ParkingSpot	IE: Has its own data

the admin, asking him to confirm it.	...who validates the ParkingSpot data? (global validation)	WebApp ParkingSpot	IE: In the DM ParkingSpot is registered in ParkingLot IE: ParkingLot is registered in WebApp
9.The admin confirms.	n/a		
10.The platform saves the data and informs the admin that the operation was successful.	...responsible to save the admin input data?	ParkingSpot WebApp (?) DataBase	

Sequence Diagram

