Barchester City CarPark System

Design Project for Software Systems

Group 36
Jan Reinder Zeinstra
Maximilian Stein
Janwillem te Voortwis

Table of contents

Introduction

Group 36 was contacted by the Barchester City council to design a software system for a car park. The requirements were defined in a project description, and refined in an interview. This document is meant to show the latest progress in the systems design. To finalize it, Group 36 asks the Barchester City council to review the proposed design.

The appended design files were mostly created with the uml-modeling software papyrus (https://eclipse.org/papyrus).

To open a file in papyrus, click on file->open file, select the .di file in the appropriate folder. It opens either an overview, where you need to click on the view in the "Notation Views" section, or the model view itself. Duo to different environments, the output can differ from the exports included in this document, nevertheless the content is the same.

A few sequence diagrams and activity diagrams are modeled with Modelio (https://www.modelio.org).

Minutes interview city council representative 23-11-2016

Hoe werkt het huidige systeem?
"Er zijn geen specifieke problemen op dit moment, we missen vooral een aantal dingen, zoals managementfuncties".

Welke managementfuncties hebben jullie nodig?
Belangrijk is vooral de verkoopinformatie, hoeveel tickets, wanneer, in welke parkeergarage. Daarmee kunnen we statistieken maken welke garage op een gegeven moment druk is.

Is ook functionaliteit nodig betreffende onderhoud?
We hebben contracten met onderhouds- en beveiligingsbedrijven, maar willen natuurlijk wel weten of zij hun SLA's naleven. Is het beveiligingsbedrijf er wel 2 keer per dag geweest? We willen hiervoor de aankomst en vertrektijden zien, en in het geval van incidenten hoe lang het duurt totdat ze er zijn. Er zijn ook medewerkers van ons aanwezig, deze hoeven niet bijgehouden te worden.

Moet het systeem rekening houden met uitzonderingen zoals nooddiensten, kwijtgeraakte kaartjes, verwijderen auto's?
Hiervoor zijn medewerkers ter plekke aanwezig, het systeem hoeft hier geen rekening mee te houden.
Als je je ticket kwijt raakt, betaal je het maximale tarief.

Hoe zullen seizoentickethouders behandeld worden als de garage vol is?
De plek is gereserveerd, dus zij mogen ook naar binnen als het bordje aangeeft dat de garage vol zit. Voor anderen blijft de slagboom dicht.

Design conclusions from interview

New requirements:
To keep track of selling information (parking, when, how many tickets)
To keep track of arrival and departure times of contractors
To record time to arrival for contractors in case of emergency
To allow employee input/overwrite on system actions
To allow season ticket holders to enter full garage in case of free season-spots

Glossary

Term	Description
Season Pass	Pass that can be used to enter and exit a specific car park. A parking spot is reserved on workdays for the owner of the card. In the weekend the reservation does not count.
Ticket	Piece of paper that is printed at entrance of a car park when not using a season pass. Contains information about the time of entry and is used to pay for parking and to leave the car park.
Control pillar	Pillar that controls the barrier at entrance and exit of car parks. When entering either a season pass needs to inserted in the pillar or a ticket printed. At exit a season pass or ticket needs to be inserted.
Pay station	Machine used to pay for parking when using tickets. Validates tickets once payment is done such that the control pillar accepts them at exit.
Attendant	Person working in a car park. If problems occur customers should talk to him. Can raise barriers remotely.
City Council	City council of Barchester.
Security personnel	People hired by the City Council to keep the car parks safe.
Security card	Card used by security personnel to check-in and checkout in an office in the car park. Can also be used to enter and exit car parks like a season pass.
Maintenance personnel	People hired by the City Council to do maintenance work on the car parks.

Requirements list

In the requirements list you can find all the requirements found in the documentation or mentioned during the interview. The requirements are associated with certain use cases;

- Enter car park
- Pay for ticket
- Leave car park
- The actions of car park personnel (management)

nr	Requirement	Use Cases
1	To issue parking tickets	Enter
2	To issue season tickets	Management
3	To handle ticket payment	Pay
4	To handle the controlling barriers	Enter,Leave
5	To record problems	Management
6	To pay in advance for parking	Management
7	To reserve spaces for season ticket holders	Enter
8	To use seasontickets on weekdays only	Enter
9	To reserve spaces up to a maximum of 10% for season ticket holders	Enter
10	To use all the spaces for all users in the weekend	Enter
11	To use a sensor to see if a car approaches an entry barrier	Enter
12	To see a button flashing on the control pillar when a car approaches	Enter
13	To press a button to issue a ticket	Enter
14	To print a ticket within 5 seconds	Enter
15	To see a flashing notification "Take Ticket" when ticket is printed	Enter
16	To see a flashing notification with "Full" if carpark is full	Enter
17	To see that space is available when a car leaves	Enter
18	To raise the barrier when a ticket is pulled	Enter
19	To insert a season pass into the control pillar	Enter
20	To check the validity of a inserted season pass	Enter
21	To check for validity of a pass within 5 seconds	Enter
22	To check if it is a weekday	Enter
23	To check if the season pass holder already parked in the carpark	Enter
24	To raise the barrier when a insterted seasonpass is valid	Enter
25	To make a record when the season pass holder enters the carpark	Enter
26	To lower a barrier when a sensor on the other side of the barrier senses car	Enter,Leave
27	To make database entry with details of tickets (ticket number, issue date, issue time, issue machine)	Enter
28	To print the barcode containing date, time of arrival and a number on the ticket (Date:dmmyyyy, time: hhmmss)	Enter
29	To print the date, time of arrival and a number on the ticket	Enter
30	To add one to the carpark counter when a car enters	Enter
31	To check if the carpark is full when a new car enters the carpark	Enter
32	To turn on a sign to show carpark is full, when the carpark is full	Enter
33	To not allow a car a ticket when the car park is full	Enter
34	To allow new cars a ticket when the park is not full anymore	Enter
35	To allow season ticket holders a ticket when reserved spaces are still available when the rest of the carpark is full	Enter
36	To pay for a visit a visitor has to insert a ticket in a paystation	Pay

|		
37	To compare the ticket data with stored data, the bar code should be scanned	Pay
38	To reject a ticket, the time or date on the ticket should not be the same as the data stored or the ticket is not readable anymore	Pay
39	To pay an rejected ticket, a customer should go to the office	Pay
40	To notify a customer of a rejected ticket and show where it should be payed	Pay
41	To check a ticket for rejection the office attendant uses a bar code checker	Pay
42	To calculate and show the price of a rejected ticket within 2 seconds	Pay
43	To charge, take payment and validate a ticket the office software is used	Pay
44	To calculate duration of stay in the carpark	Pay
45	To have a tariff from weekdays 8am to 6 pm and another one outside of that range	Pay
46	To have a tariff for long and short duration	Pay
47	To calculate price of ticket	Pay
48	To show a message when a coin or note is invalid	Pay
49	To identify note or coin value	Pay
50	To show a message when no change is in the machine	Pay
51	To only accept at least the charge when no change is available	Pay
52	To keep track of how much money is put into the machine	Pay
53	To update the message that shows how much money is already put into the machine	Pay
54	To validate a ticket as soon as the accumulated amount of money is larger than the charge	Pay
55	To store the data of a ticket when the payment is validated (payment date and time)	Pay
56	To show a message when change is calculated and put into the tray	Pay
57	To print the date and time of payment on validated ticket	Pay
58	To eject validated ticket	Pay
59	To show option to print receipt	Pay
60	To print a receipt when the button for receipt is pressed	Pay
61	To print the council office, address of car park, VAT-number, date and amount payed on the receipt	Pay
62	To show a message "leave carpark within 15 minutes" when the ticket is validated and receipt is printed	Pay
63	To print a full price ticket at office when paid for	Pay
64	To show the message "Insert ticket" when a car arrives at a outgoing barrier	Leave
65	To check if the ticket is inserted within 15 minutes since payment	Leave
66	To raise the barrier if the ticket check is passed	Leave
67	To sense and close the barrier when car has left the carpark	Leave
68	To subtract one from the carpark counter when a car leaves the carpark	Leave
69	To talk with a driver remotely when 15 minutes since payment have passed	Leave
70	To raise the barrier remotely from office	Leave
71	To make a record of the leaving time when a season pass holder checks out	Leave
72	To update the status of every pay machine and barrier control pillar every 10 seconds	Management
73	To register the arrival of a security guard when he or she puts the card in the card reader in the office	Management
74	To register the leaving of a security guard when the card is scanned for a second time	Management
75	To create statistics to see if the security does their job	Management
76	To keep a record of all the incomming and outgoing cars	Management
77	To create statistics out of the recorded records	Management
78	To store faillure information into the system manually	Management
79	To store faillure information into the system automatically	Management
80	To warn firm when the system is not working as required	Management
81	To update failure information when system is fixed	Management
82	To create statistics to see if the maintenance firm does their job	Management

Activity diagrams

For the activity diagrams it was convenient to split up the model into a couple of relevant actions. Activity diagrams are made for use of the garage by normal customers. by season pass holders, maintenance and security work and management by the Barchester officials.

Ordinary customer
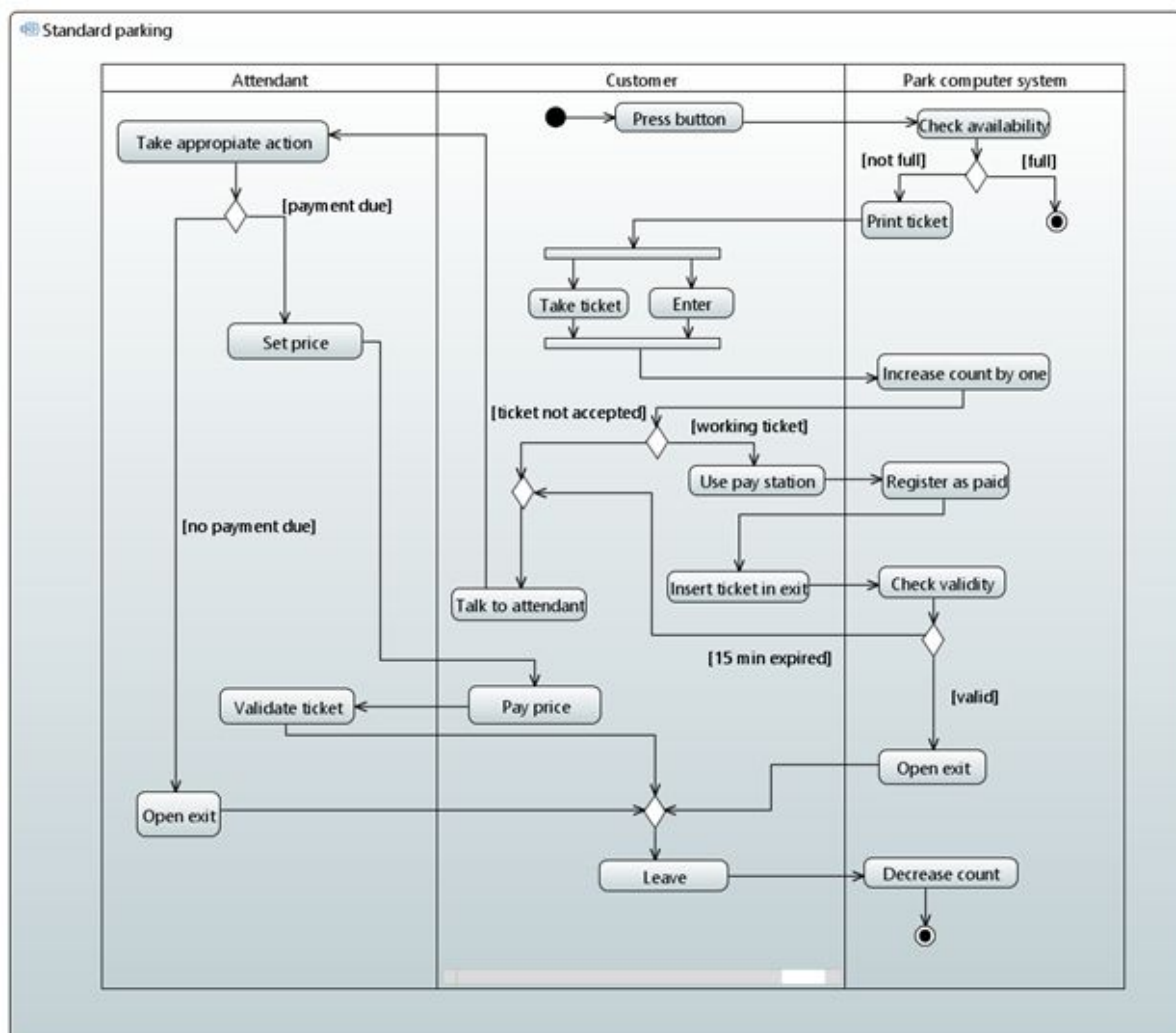
In this diagram the use of carpark for normal customers who use tickets is modelled. Some exceptions such as when the park is full or when people cannot use their ticket are modelled as these situations are relevant for the functioning of the system and are mentioned in the Case Description. These exceptions are relevant because they show that the attendant needs to be able to access the system and be able to validate tickets. The computer system itself is also modelled as an actor because it was needed for checks and to log how many cars are in the garage for the officials to use for management purposes.

Season pass holder
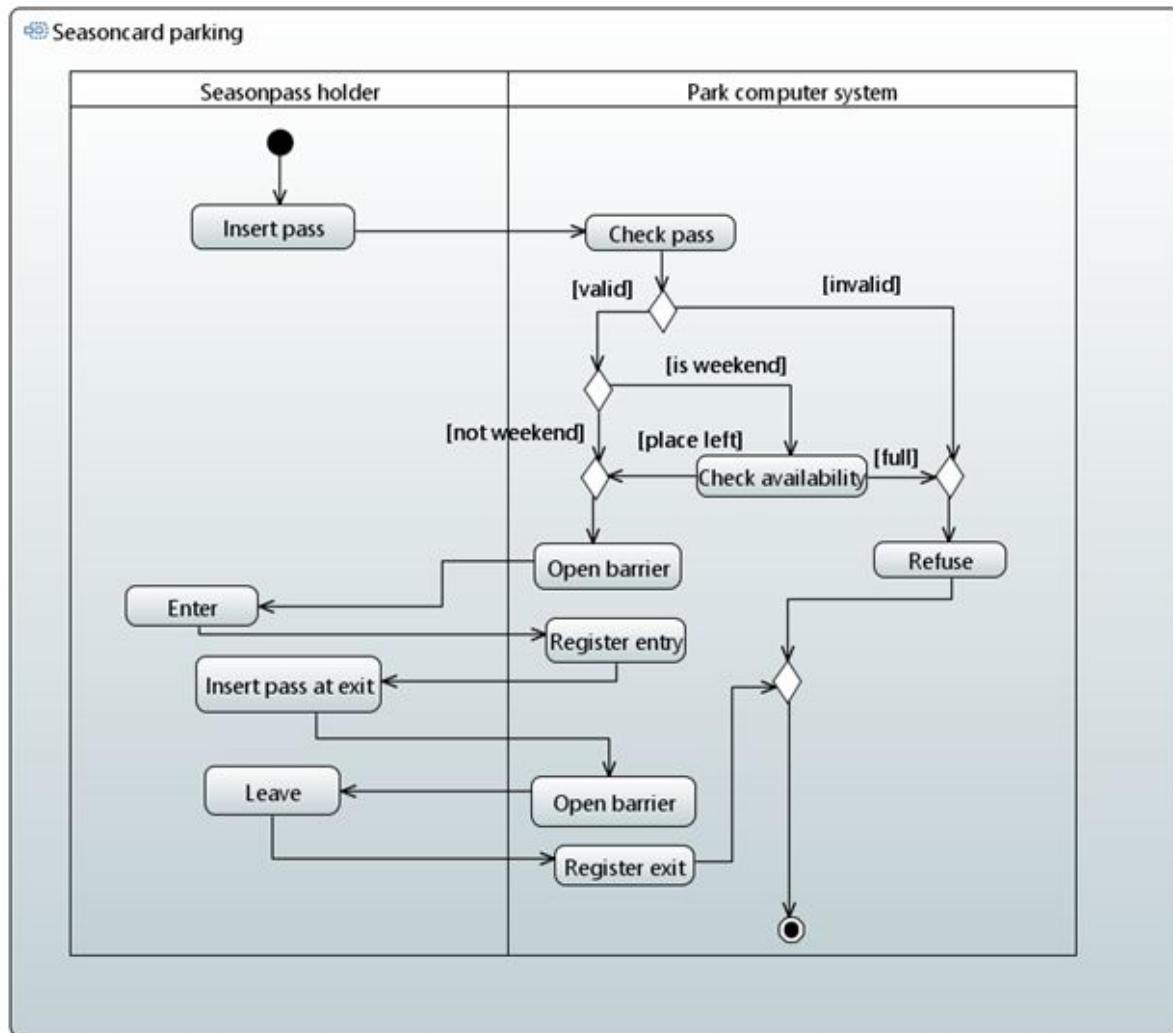
In this diagram the use of the car park for season pass holders. Exceptions to the normal flow are when a pass that is not valid is inserted, this includes not being valid for the current car park or already being registered as parked. Also when it is weekend a check is made if the park is not full already since in weekends no spots are reserved for season pass holders.

Maintenance and security
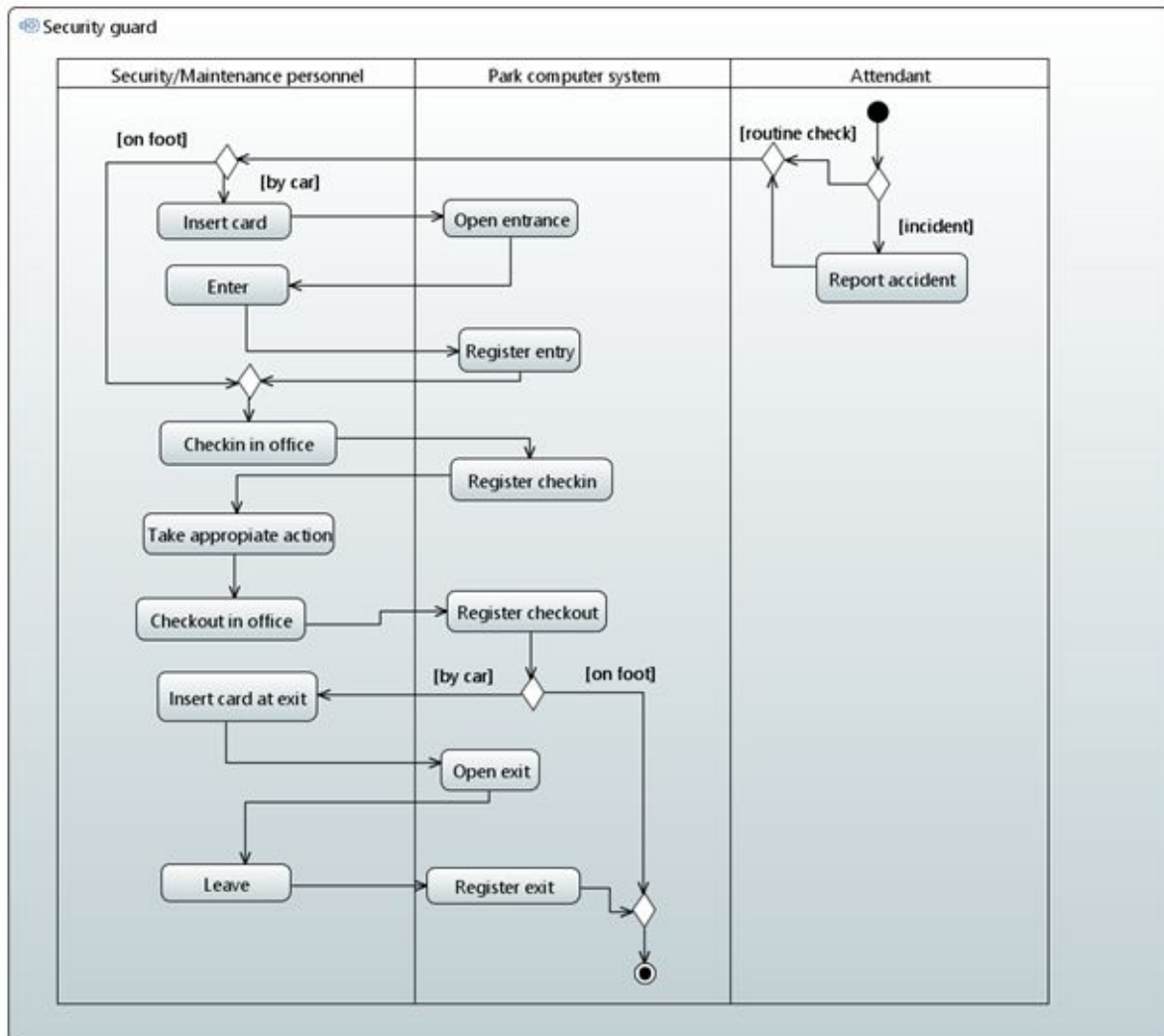
In this diagram the functioning for security and maintenance companies is modelled. While the companies have a different job their use of the system is the same hence they are put in one model. Security personnel always checks in before responding in the model while this might not always be the case for urgent matters this was not deemed an exception worth modelling.

Management
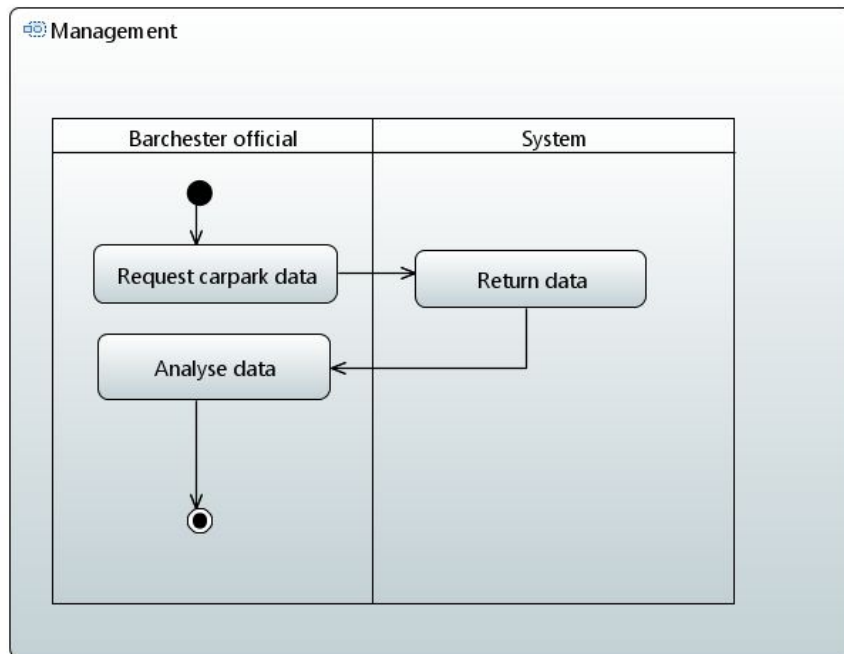
It was mentioned in the interview that Barchester officials would use the system for management purposes and to evaluate the performance of companies this has also been modelled in this diagram. There would be many different options for which specific data to retrieve from the system. However, this has not been modelled since for each different data to request the system would still work the same.
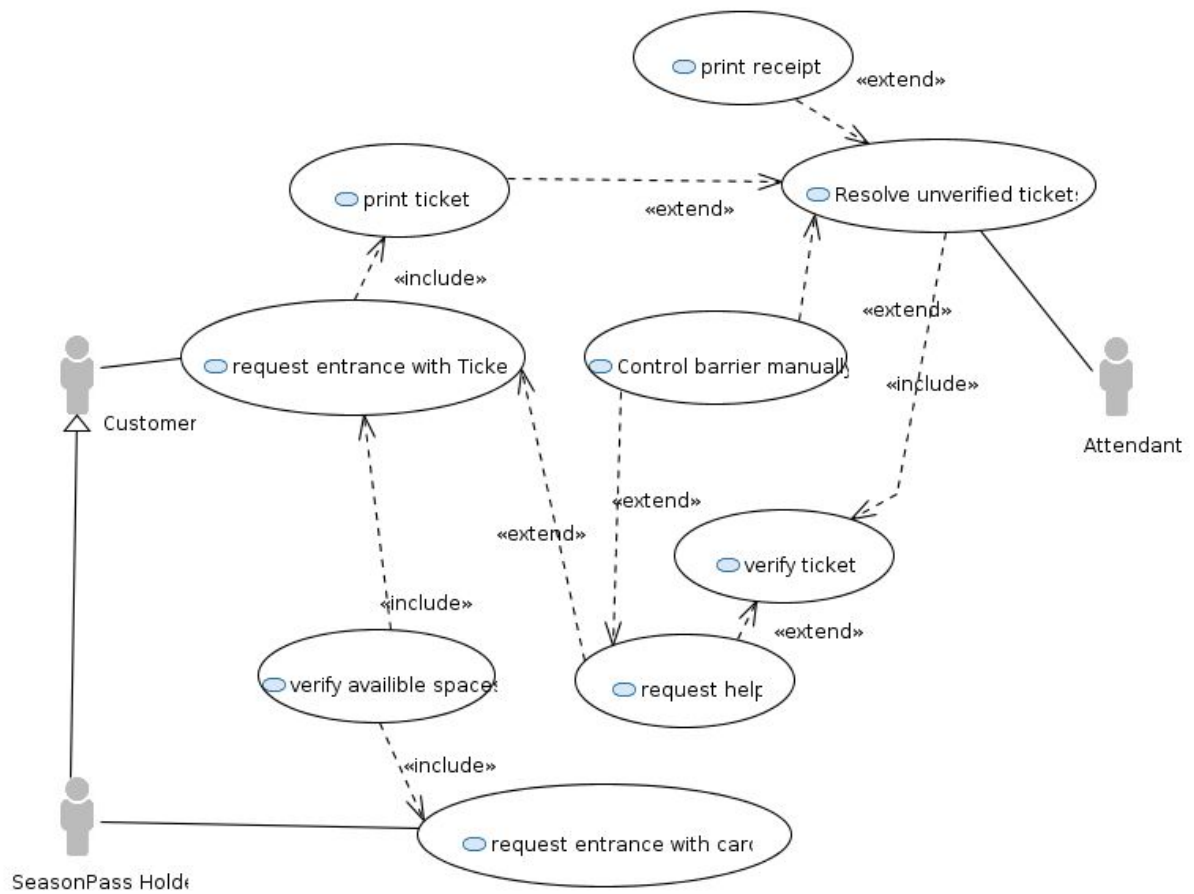
Use case Diagrams

A customer is the representation of a normal ticket user. The season pass holder has it's own way of entering the carpark with a seasonpass.
Both the normal ticket holder as the season pass holder leave the carpark the same way, because the system only has to see which if the products is used to check out and raise the barrier.
If the customer would experience problems, the same action, talk to the office attendant, would be used to solve the matter.
The buying and issueing of season cards is included for completeness. A security officer has the same options to enter the car park as a season pass holder, but he or she also has other responsibilities, so the security officer is placed separately to the right side keep the diagram more clear. The maintence officer also has several actions. It was not clear if he needed to enter or leave the carpark, so it is not modelled into the diagram.

Enter

Pay
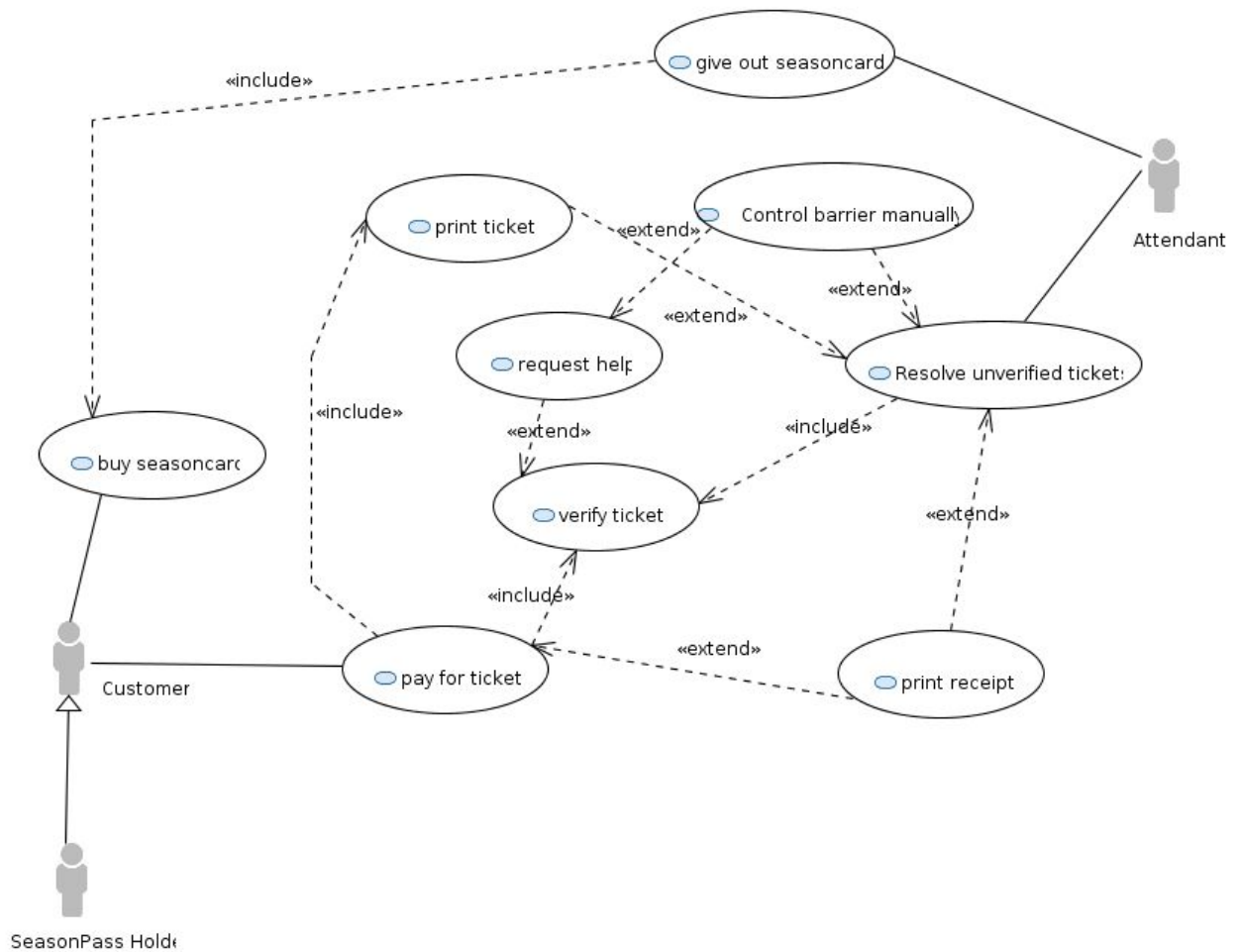
Leave

Management

Brief use case descriptions

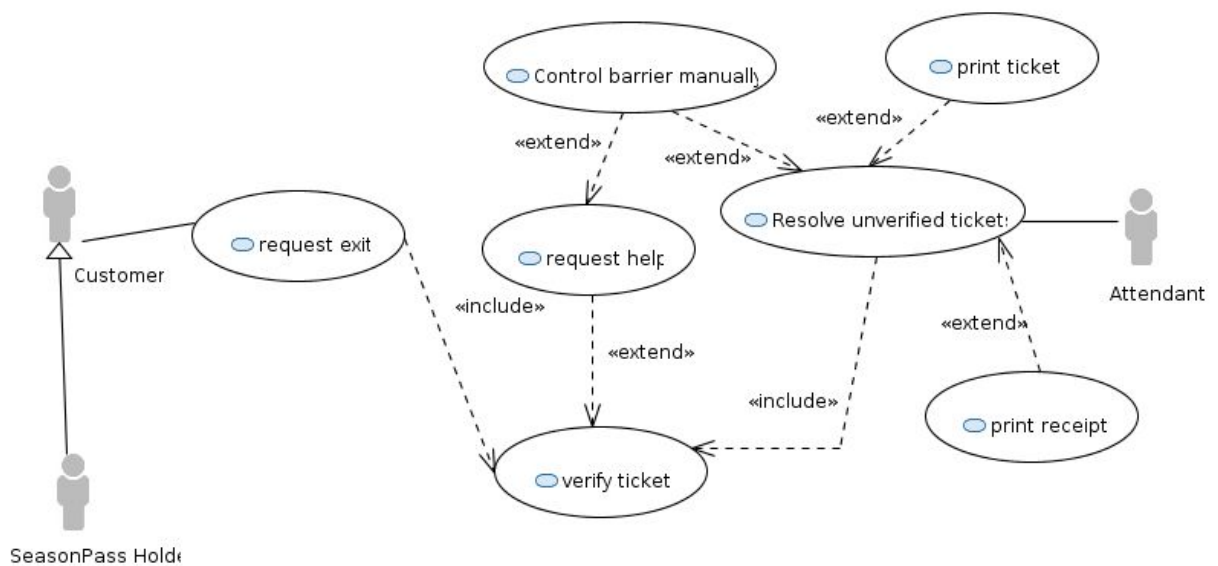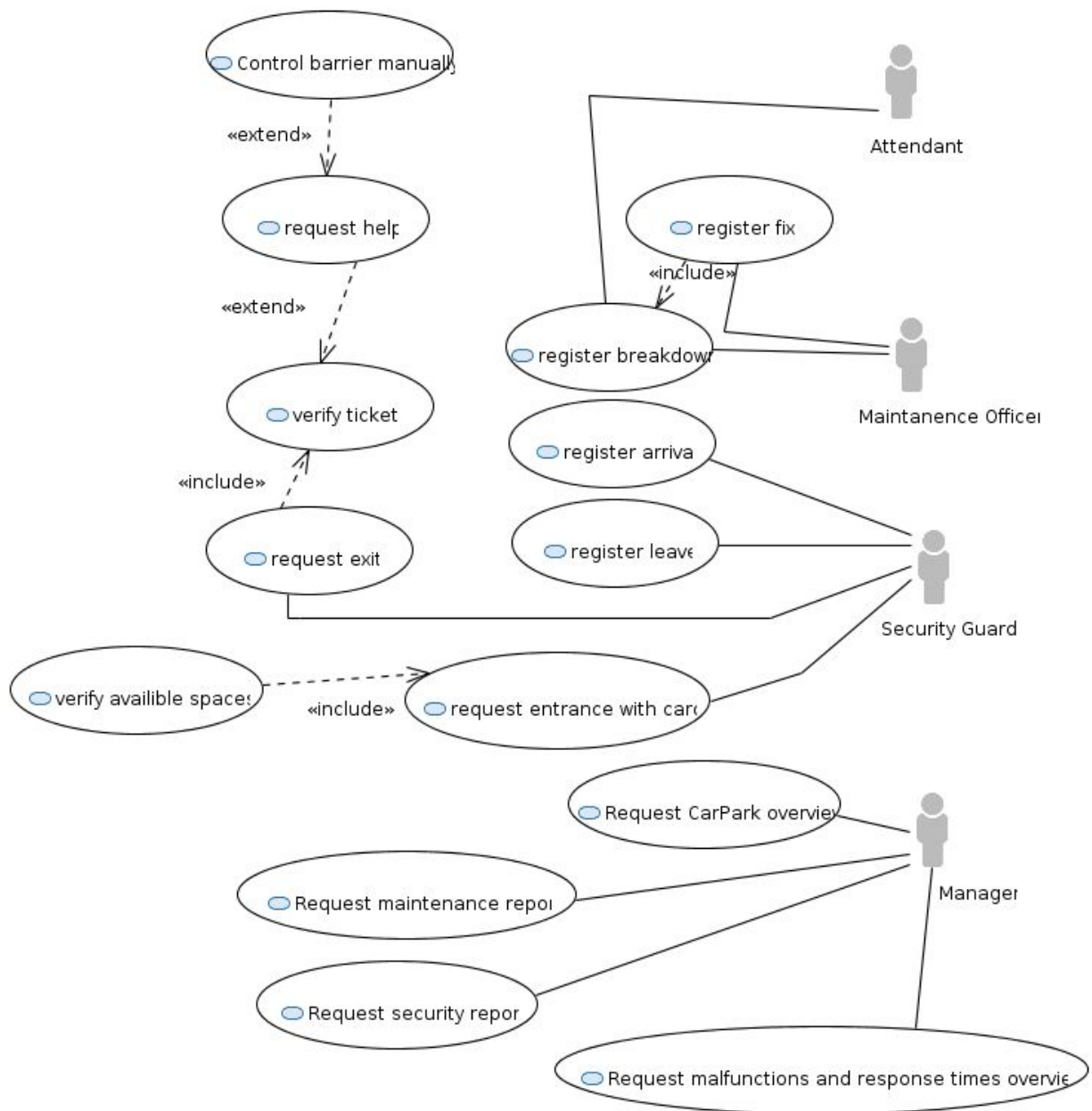|	
Enter car park with ticket	When a customer arrives in a car, a sensor will sense that a car is in front of a control pillar. The customer presses a button on the screen if the control pillar to get a ticket. The barrier in front of the car will raise and the car will enter the car park
Buy season pass	A customer can buy a season pass. How to do this is not clear from the documents.
Pay for ticket	The customer has to pay for the ticket at the ticket machines. The customer inserts the ticket. Then the customer inserts the amount of money needed. If the customer inserts too much money and the machine has change available then the machine will give back change. If payed the ticket is validated.
Leave car park	The customer rides his or her car in front of the control pillar and put a ticket or season card into the machine. If the ticket is validated, the the barrier will be raised and the car can leave.
Enter car park with seasonpass	A season card holder drives his car in front of a control pillar. The customer grabs the card again an drives into the car park.
Print receipt	When a customer wants a ticket he presses the print receipt button on the screen of the pay station if the ticket is paid. The pay will then output a receipt.
Verify ticket	A check to see if the ticket is payed for at the ticket machine and did not exceed the time or is a valid seasonpass
Raise barrier manually	When a customer passes the 15 minute mark, the customer is connected to the office attendant via intercom. If the customer gives a valid reason for the expired ticket, the attended will open de barrier in front of the customers car remotely.
Resolve unverified tickets	When a customer has trouble paying a ticket the customer can go to the office attended who will look at the ticket. If needed the ticket will be payed through the attended and the ticket will be validated manually.
Register Incident	When one of the employees or security employees witnessed an incident, he or she has to report it in the system.
Register Arrival	An security employee has to register his or her arrival by putting a card into the scanner available at the office.
Register Leave	An security employee has to register his or her leave by putting a card into the scanner available at the office.
Register Breakdown	When one of the employees or maintenance employees witnessed a breakdown, he or she has to report it in the system.
Register Fix	When a breakdown is fixed by a maintenance employee the fix needs to be registered in the system via a portal.
Retrieve management data	A manager can analyse the data collected by the system after the management determines which data to analyze.

Extended use case descriptions

These diagrams show the extended use case descriptions. Some diagrams have the actions of a seasonpass holder and a regular ticket holder. These are two different use cases nested in one diagram.

Enter car park

	Customer		System
		1	Show on board if parking is full
2	Approach barrier	3	Flash button on control pillar
4a	Leave carpark if park is full		
4b	Press button to request ticket	5	Check and update if car park (minus season spaces on weekdays) is full
		5a	If full, ask user to leave
		5b	If not full, print ticket with: number, date, time of arrival and barcode on ticket
		6	Generate barcode containing date, time of arrival and number
		7	Make database entry with thicket number, issue date, issue time, issue machine
		8	Push out ticket, flash notification "take ticket"
9	Pull out ticket	10	Open barrier
11	Drive inside	12	Increase car park counter
		13	Wait until car has entered car park
		14	Close barrier
	Season pass holder		**System**
1	Approach barrier	2	Flash button on control pillar
3	Enter season pass into pillar	4	Check if weekend, season pass valid, and if not already inside, and if season spaces available
		5a	If one check not passed, ask user to leave
		5b	Open barrier
6	Drive inside	7	Register season pass id as parked in car park
		8	Wait until car has entered car park

	Security/maintenance personnel		System
		9	Close barrier
1	Approach barrier	2	Flash button on control pillar
3	Enter security pass	4	Check if pass is valid
		5	If pass is valid, open barrier
6	Drive inside	7	Wait until car has entered car park
		8	Close barrier
Alternative courses: Enter by foot, start at 9			
9	Go to registering pillar	10	Register check in of security

Payment

	PAYMENT		
	Customer		System
1	Insert ticket in pay station	2	Validate ticket, compare with stored data
		3	If valid, calculate and display price, display instructions.
		4	Display if change is available
5	Insert money	6	Check money, if invalid return money and display money not accepted
		7	Count valid money, if exceeds or matches price validate ticket
		8	Return change if necessary and possible, display transaction finished and amount of change due and amount of change returned.
		9	Ask if customer wants a receipt.
10	Take change if necessary		

11a	Press receipt button	12a	Print receipt with Council address, car park address, VAT number, date and amount paid.
11b	Press continue	12b	Exit payment menu
		13	Add payment date and time to the ticket and eject the ticket
		14	Display "take ticket back to your car and leave the car park within 15 minutes"
15	Take ticket		
	ALTERNATIVE		
	Repeat step 5 and 6		
	If not valid at 2 continue below		
	Customer		Attendant
1	Hand over ticket	2	Scan barcode
		3	Calculate charge
4	Pay attendant	5	Register payment
		6	Validate ticket and give it back
7	Take ticket		

Leave car park

	Customer		System		Office Attendant
		1	show the message "Insert ticket" when a car arrives at a outgoing barrier		
2	Insert Ticket	3	check validity and time passed since payment is under 15 minutes		
		3a	if check passed raise barrier		
4	leave carpark	5	sense when car has left the carpark and close barrier		
		6	subtract one from carpark counter		
		7	check if the carpark is full		
		7a	if full leave sign on full		
		7b	if spaces available set sign on spaces		
	Alternative course				
		3b	if check not passed activate intercom	4	talk to driver
5	talk to office attendant				
				6	raise barrier remotely
		7	raise barrier		
8	leave carpark	9	sense when car has left the carpark and close barrier		
		10	subtract one from carpark counter		
		11	check if the carpark is full		

		11a	if full set sign on full		
		11b	if spaces available set sign on spaces		
	Seasonpass holder		**System**		
		1	show the message "Insert ticket" when a car arrives at a outgoing barrier		
2	Insert seasonpass	3	raise barrier and return seasonpass		
4	take seasonpass and leave carpark	5	sense when car has left the carpark and close barrier		
		6	make record of when the seasonpass holder left		
		7	subtract one from the seasonpass spaces counter		
		8	check if the carpark is full		
		8a	if full set sign on full		
		8b	if spaces available set sign on spaces		

Management

|	Manager		System
1	Open system dashboard	2	Show current capacity status of the carpark
		3	Indicate current status of paymachines and control pillars
		4	Show 5 day security statistics
		5	Show 5 day maintenance statistics
		6	Show warnings for system disfunction
7	Request extended carpark statistics	8	Show history view of carpark capacity, states, arrival and departure statistics of customers
9	Request extended security statistics	10	Show history view of security arrivals and departures
11	Request extended maintenance statistics	12	Show history view of maintenance reports and corresponding arrival times of contractors
13	Request extended failure overview	14	Show overview of all system failures
Alternative Courses:			
Actions 7,9,11,13 can be requested at will			
	Attendant		System
1	Open maintenance overview		
2	Input new breakdown	3	Store failure in system
4	Inform maintenance service		
	Maintenance Officer		System
1	Repair failed system	2	Save repair time
3	Extend failure report and add repair report in system	4	Store repair and failure report

Sequence diagrams

The sequence diagrams are an elaboration of the extended use case diagrams.
Where applicable a hardware controller is included as an object to control several hardware functions around the car park. This is used to simplify the system as we don't need to model objects for a barrier or the signboard for example.

Enter car park

In this diagram the entry of a car park and interaction between control pillar is described. To reduce complexity a reference is made to diagrams with how the process works for normal customers and season pass holders.
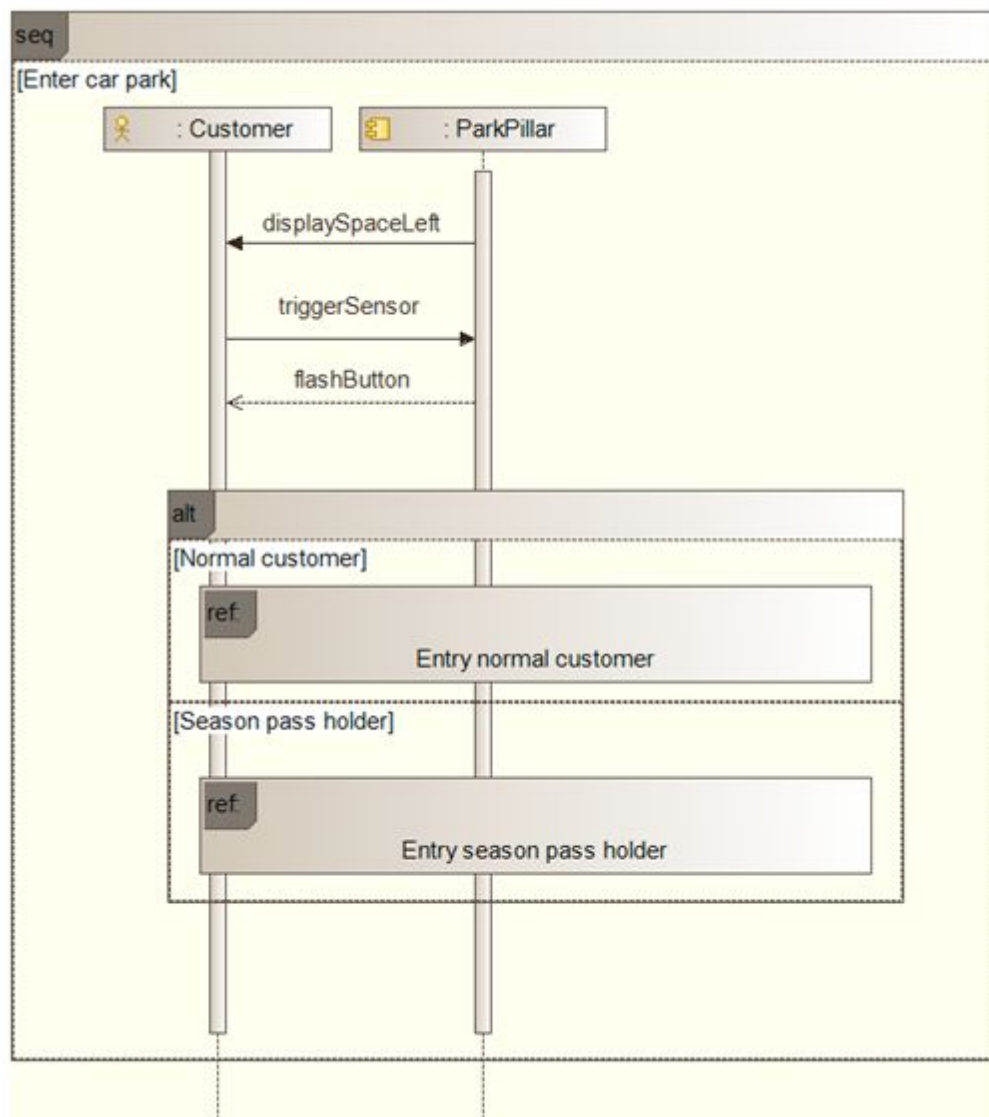
Sequence diagram describing the process for ordinary customers. "GetParkingSpaces" should return the amount of non reserved parking spaces. "GetSeasonSpaces" should return the amount of season pass spaces. The "getParked" functions should return the amount of spaces that are used. With this the program can check if the park is full.
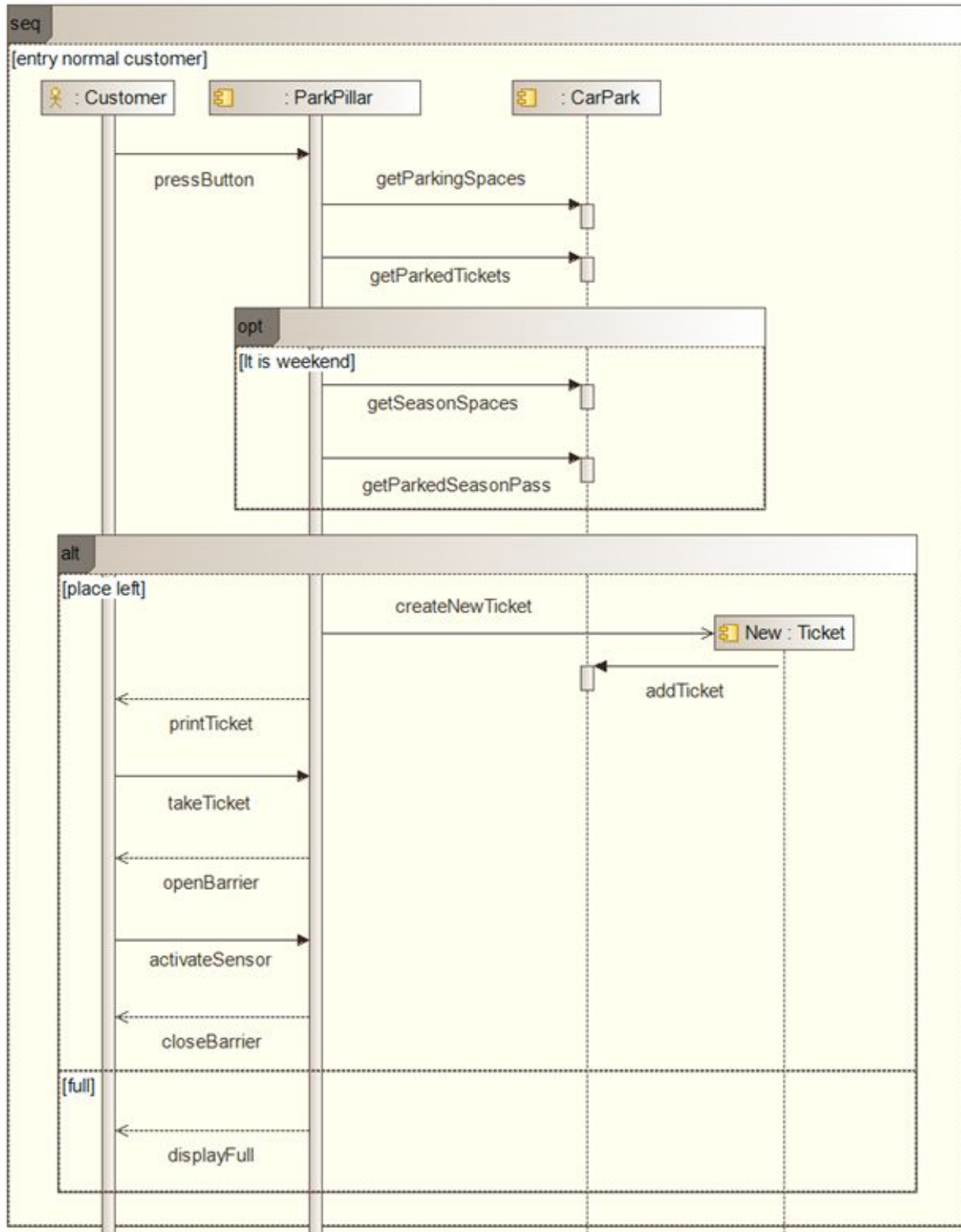
Sequence diagram describing the entry process for season pass holders. The "getTotal" functions work similar to the functions of the previous diagram except then they return the total amount so both reserved and unreserved spots. The "checkIfParked" function checks if the season pass holder has not already parked a car.
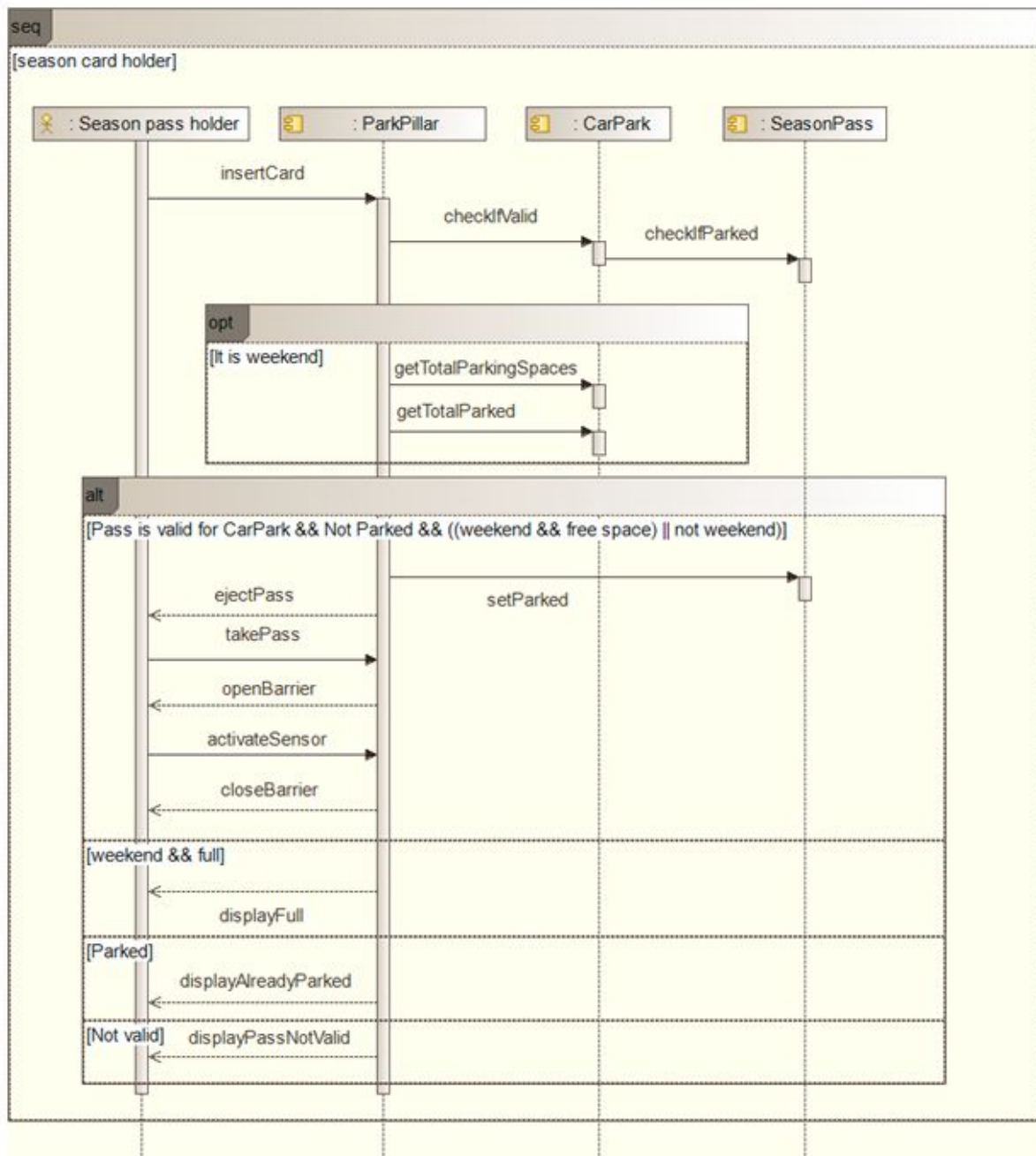
Pay

This diagram shows the process of paying for ordinary customers. Functionality not described in the case has been added in the form of a stop button which is used to exit the process when paying. This can be used if the customer does not have enough money such that they get their money back. The "validateTicket" function may be confusing as it checks if the ticket exists in the database and if payment is due on it rather than actually validating the ticket. Instead "setTicketPaid" is used to validate the ticket in the meaning of what is explained in the case description.
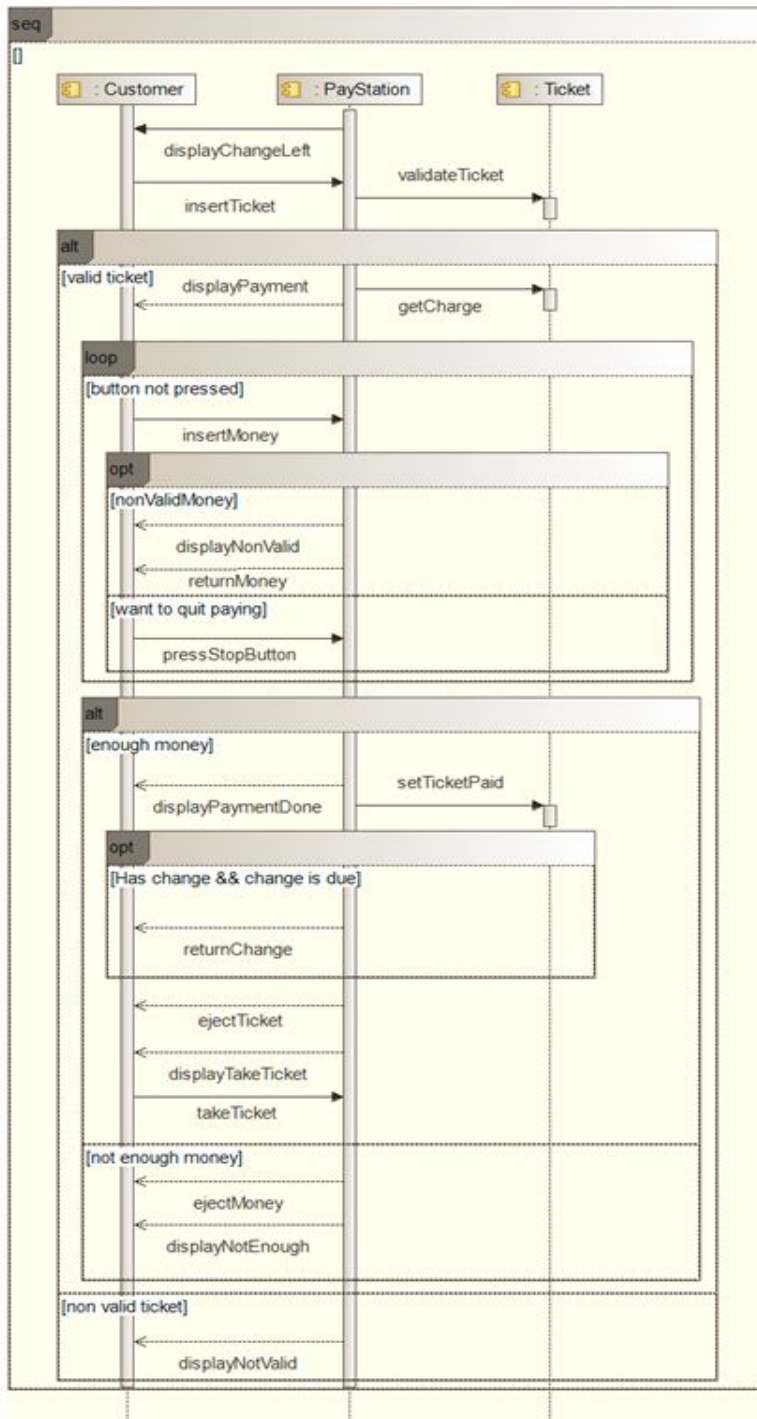
Leave car park

This sequence diagram shows the interaction of the customer with the system when leaving the carpark. It goed through the steps of the system when leaving the carpark. It has an option for the office attendent to help the customer.
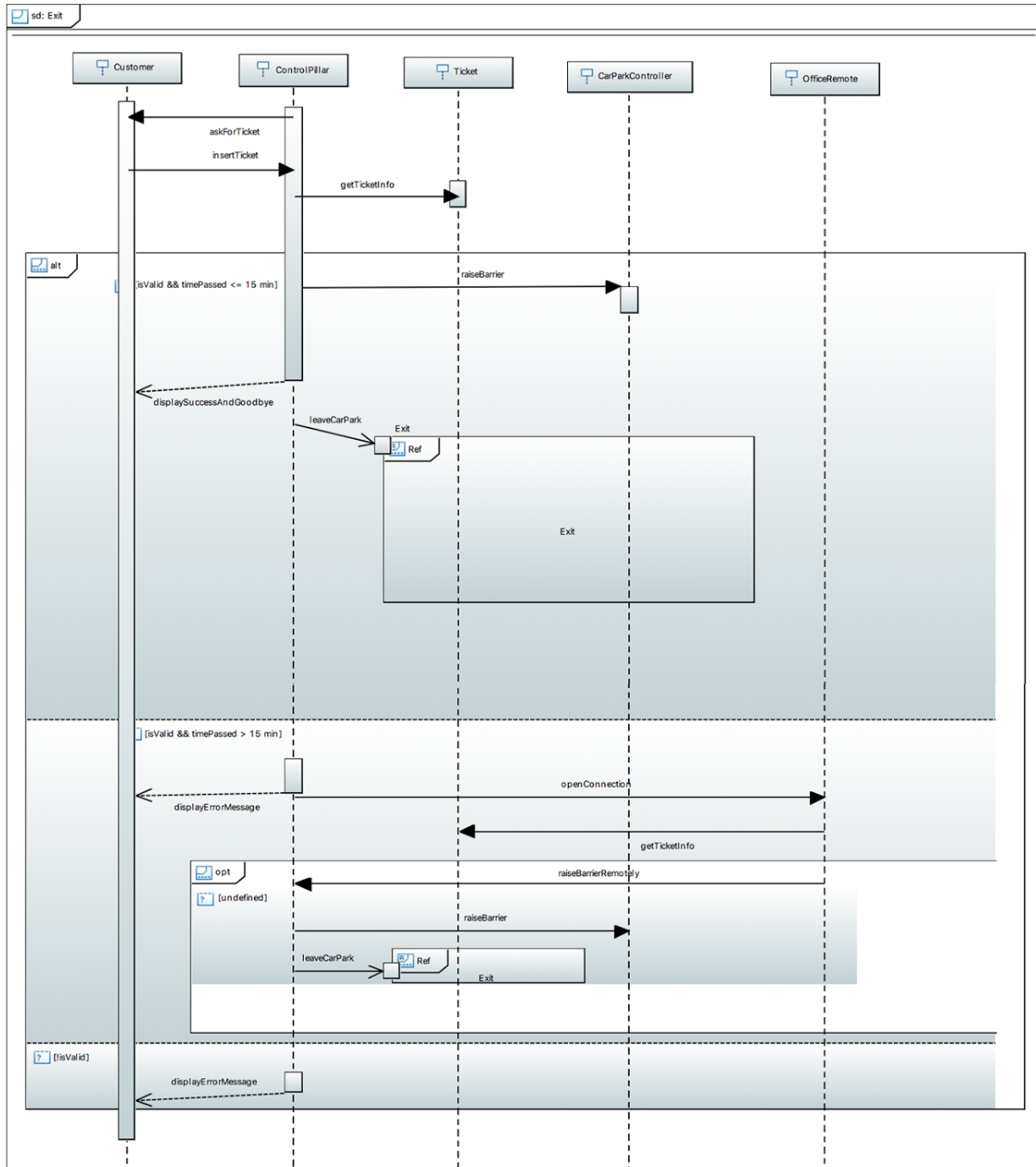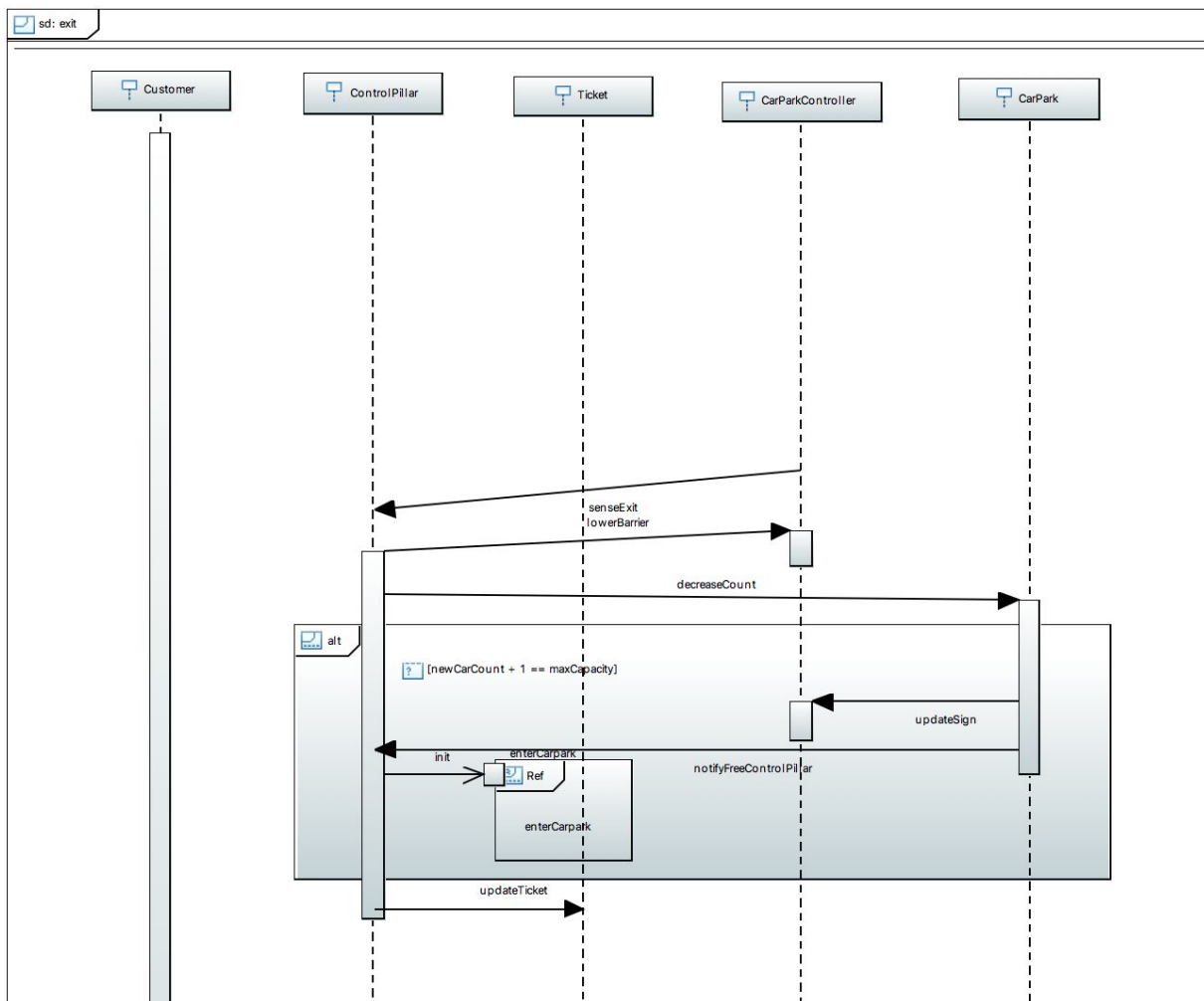
Management

This sequence diagram shows the interaction of the manager with the system. The start is a overview generated by the carpark class, which is related to all relevant management information sources. Afterwards the manager can choose to view more detailed overviews of specific information sources.
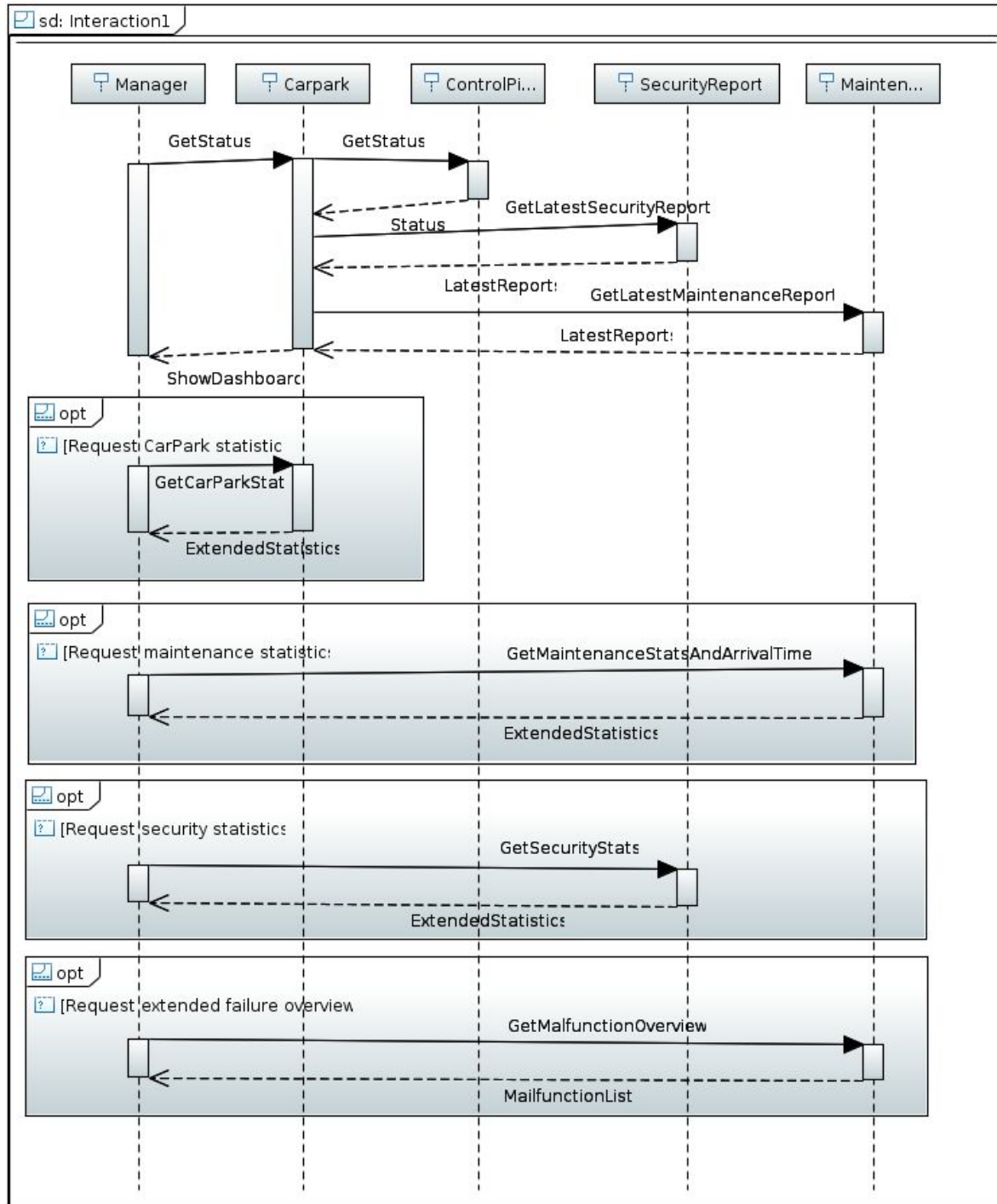
Class diagram

The class diagram gives an overview of the objects that exists in the system. Besides the objects that are obvious considering a car park, the carpark and the customer, more objects are included. Because the service of the security and the maintenance needs to be evaluated, the relation between the carpark and these companies have an association class with the details about the contract. The quality of the services of both companies will be registered and can give a historical view of the service during a specific period.
This diagram also holds the possibility that a carpark can contract multiple security- and maintanance companies with corresponding SLA's.

MaintenenceSLA
+ maxResolveTime: int [1]

Customer
+ name: String [1]
+ email: String [1]
+ phoneNumber: int [1]

MaintenenceCompany
+ name: String [1]
+ address: String [1]
+ tel.no: int [1]

SecurityCompany
+ name: String [1]
+ address: String [1]
+ tel.no: int [1]

has a
1
*

SeasonPass
+ validFrom: EDate [1]
+ validTill: EDate [1]

made by
1

MaintenenceReport
+ malfunctionTime: EDate [1]
+ resolveTime: EDate [1]
+ equipment: String [1]
+ comments: String [1]

made by
1

SecurityReport
+ incidentTime: EDate [1]
+ location: String [1]
+ report: String [1]

works for

belongs to
1
*

SeasonTicket

GeneralReport

securitySLA
+ minimumVisits: int [1]
+ minVisitDuration: int [1]

works for

done by

GeneralTicket
+ entryTime: EDate [1]
+ exitTime: EDate [1]

is for
*
1

CarPark
+ name: String [1]
+ address: String [1]
+ tel.no: int [1]
+ type: EEnumerator [1]
+ parkingspots: int [1]

is for
1
*

SecurityLog
+ arrivalTime: EDate [1]
+ departureTime: EDate [1]

1

Ticket
+ state: EEnumerator [1]

ControlPillar
+ type: EEnumerator [1]

State machine diagram

CarPark

Freespace

[weekend]

FreeSpaceNoSeason

[weekday]

OutOfPublicCapacity

FreePublicCapacity

FreeCapacity [weekend]

FreePublicCapacity [weekday]

OutOfCapacity

OutOfCapacity

FreeSeasonspace

FreeSeasonCapacity [weekday]

Full

OutOfCapacity

Planning

To get an overview of the different tasks and the deadlines a planning is made. This is also part of the course "Academic skills".

Date	5-11 dec	12-18 dec
Monday		Use case diagrams
Tuesday		Brief use case descriptions
Wednesday		Class diagram Extended use case diagrams
Thursday	Requirements Interview Report	State machine diagram Sequence diagrams
Friday	Activity diagrams	Writing brief report Glossary
Saturday		Finalising report
Sunday		Design Deadline

Who did what

Load balancing was done on a voluntarily basis. Everyone just asked for things to do and tasks were decided upon by one's own suggestion. In the following schema it can be seen which team member did what tasks.

Task	Who
Activity Diagrams	
Entry	Jan Reinder
Pay	Jan Reinder
Exit	Jan Reinder
Management	Jan Reinder
Use Case Diagrams	
Entry	Janwillem te Voortwis
Pay	Janwillem te Voortwis
Exit	Janwillem te Voortwis
Management	Janwillem te Voortwis
Glossary	Jan Reinder
Requirements	Janwillem te Voortwis
Brief Use Case Descriptions	Janwillem te Voortwis
Extended Use Case Descriptions	
Enter	Maximilian
Pay	Jan Reinder
Leave	Janwillem te Voortwis
Management	Maximilian
Sequence Diagrams	
Enter	Jan Reinder
Pay	Jan Reinder
Leave	Janwillem te Voortwis
Management	Maximilian

Class Diagram	Janwillem te Voortwis
State Machine Diagram	Maximilian
Interview Minutes & Report	Maximilian