Use Case UC1: Customer authentication
Primary Actor: Customer
Stakeholders and Interests:
- Database Authentication: Wants to ensure that the user is registered and logged into the system
Preconditions:
The customer is a real person with a valid email address
Success Guarantee (Postconditions):
The customer is logged in to the system
Main Success Scenario (or Basic Flow):
1. The customer enters their username and password into the system
2. The database authenticates the login information
3. The customer is successfully logged in
Extensions (or Alternative Flows):
1a. If the customer has not registered yet, they will need to click the register button and enter information such as their first name, last name, and email address
2a. If the username and/or password is incorrect, the customer is requested to enter their login information again
Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the customer to enter information and to select the different options
- There is a separate button for login and register
- The system is robust and can detect false customer inputs
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc:
- What happens if a user forgets their username and/or password?

Use Case UC2: Book a parking space
Primary Actor: Customer
Stakeholders and Interests:
- Parking Enforcement Officer: Wants to ensure that a booked parking space has been paid for

- Database Authentication: Wants to ensure that the user is registered and logged into the system

- Payment Services: Wants to ensure that the payment has been authorized and completed

- System Administrator: Changes the status of the user booking from unpaid to paid when the user makes a payment

- Parking Space DB: Wants to ensure that the parking space is available for the customer to book
Preconditions:
The customer must be signed in. The customer is registered and logged in to the system
Success Guarantee (Postconditions):
The customer books an unoccupied parking space which is recorded in the system, and the payment is confirmed
Main Success Scenario (or Basic Flow):
1. The customer selects the parking space which they desire
2. The customer selects the amount of time they would like to park for (in minutes)
3. The customer adds the parking space to the cart and is able to add up to two more parking spaces if desired
4. The customer checks out the cart
5. The customer enters credit card information
6. The payment is confirmed
7. The parking space is successfully booked
Extensions (or Alternative Flows):
1a. If the parking space is already occupied, send an error message to choose another space
2a. If the parking time is invalid, send an error message to choose another time
3a. If the customer tries to add a fourth parking space, send an error message to proceed to checkout
6a. If the payment information is invalid, cancel the transaction and send a message to enter the payment information again
Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the customer to enter information and to select the different options
- The payment should take a few seconds to process
- The system is robust and can detect false customer inputs
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc:
- What are the City of Toronto Laws for parking?
- Are the parking spaces valid parking areas approved by the City of Toronto?

Use Case UC3: Booking cancellation
Primary Actor: Customer
Stakeholders and Interests:
- Parking Enforcement Officer: Wants to ensure that the cancelled space is now unoccupied and unpaid for

- Database Authentication: Wants to ensure that the user is registered and logged into the system

- Payment Services: Wants to ensure that the payment has been refunded to the user

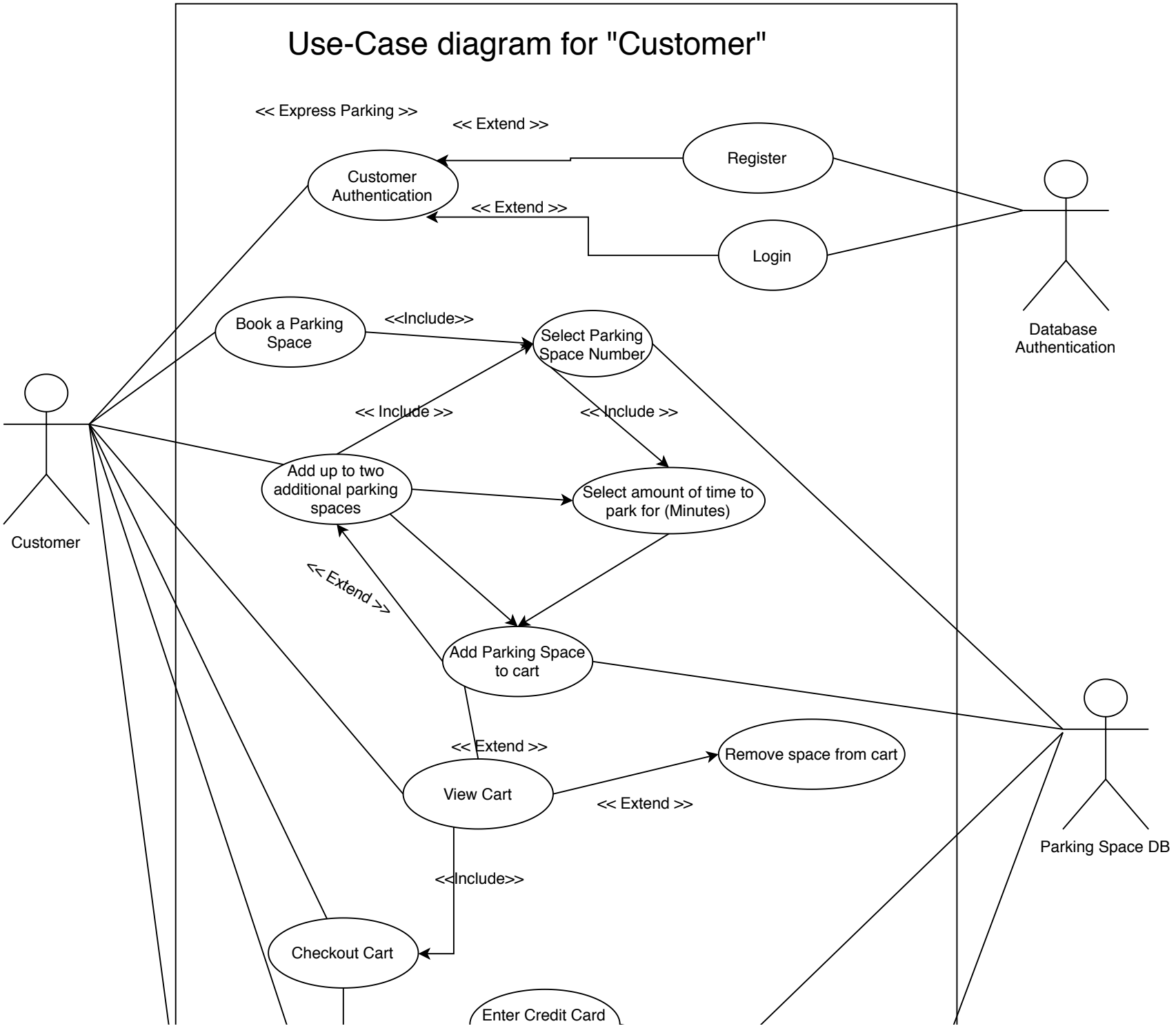
- System Administrator - Wants to ensure that the customer is cancelling before the time that they have to park

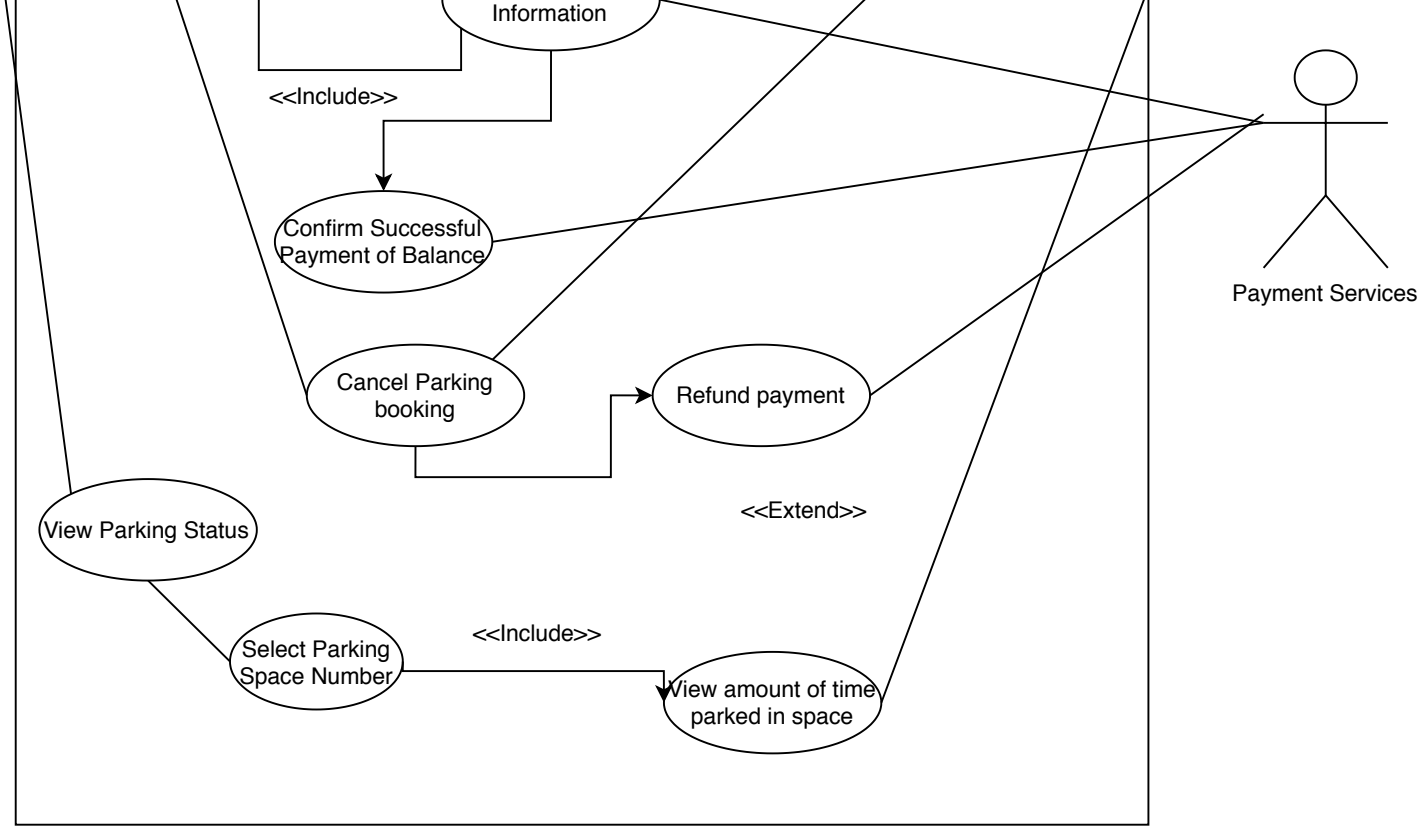
- Parking Space DB: Wants to ensure that the parking space is occupied by the correct customer

Preconditions:
The customer must be signed in. The parking space that the customer wishes to cancel is under their name, and the cancellation is before the time of the parking
Success Guarantee (Postconditions):
The parking space booking is cancelled and customer is refunded
Main Success Scenario (or Basic Flow):
1. The customer enters their booking ID in the cancellation window
2. The System Administrator cancels the booking
3. The customer receives a refund on the same card they used for payment
Extensions (or Alternative Flows):
1a. If the booking ID does not exist, the customer is requested to enter the booking ID again
1b. If the booking ID is not under the customer's name, the customer is requested to enter the booking ID again
2a. If the booking time is past the current time the customer wants to cancel, the cancellation is not processed
Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the customer to enter information and to select the different options
- There is a separate button on the main menu for cancellation
- The system is robust and can detect false customer inputs
- Each space has its own booking ID, in case the customer has multiple spaces booked
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc:
- What happens if a user forgets their username and/or password?

Use Case UC4: View parking status
Primary Actor: Customer
Stakeholders and Interests:
- Database Authentication: Wants to ensure that the user is registered and logged into the system

Parking Space DB: Wants to ensure that the correct parking information is taken for the customer to view
Preconditions:
The customer must be signed in. The customer must have at least one parking space booked.
Success Guarantee (Postconditions):
The parking space booking is viewed by the customer
Main Success Scenario (or Basic Flow):
1. The customer enters their booking ID in the view parking space window
2. The customer is able to see their booking, showing the location and amount of time parked in space.
Extensions (or Alternative Flows):
1a. If the booking ID does not exist, the customer is requested to enter the booking ID again
Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the customer to enter information and to select the different options
- There is a separate button on the main menu for view booking.
- The system is robust and can detect false customer inputs
- Each space has its own booking ID, in case the customer has multiple spaces booked
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc:
NA


Use-Case diagram for "Customer"
<< Express Parking >>
<< Extend >>
Register
Customer Authentication
<< Extend >>
Login
Database Authentication
Book a Parking Space
<<Include>>
Select Parking Space Number
Customer
<< Include >>
<< Include >>
Add up to two additional parking spaces
Select amount of time to park for (Minutes)
<< Extend >>
Add Parking Space to cart
<< Extend >>
Remove space from cart
View Cart
<< Extend >>
Parking Space DB
<<Include>>
Checkout Cart
Enter Credit Card


Information
<<Include>>
Confirm Successful Payment of Balance
Payment Services
Cancel Parking booking
Refund payment
<<Extend>>
View Parking Status
Select Parking Space Number
<<Include>>
View amount of time parked in space

Use Case UC1: Manage Parking Spaces
Primary Actor: Parking enforcement officer
Stakeholders and Interests:
- Database Authentication: Wants to ensure that database has correct amount of parking spaces, and only vacant spaces are being booked by customers
Preconditions:
- Parking enforcement officer is a real employee of the Toronto Parking Authority
- Parking enforcement officer is added in the system and logged in
- A parking space must be vacant before being granted to a customer
- A parking space must be already occupied before being rejected to a customer
Success Guarantee (Postconditions):
- A new parking space is added to the database (addition)
- A parking space is no longer in the database (removal)
- A parking space request is granted or denied
Main Success Scenario (or Basic Flow):
Adding a new parking space
1. The parking enforcement officer verifies the new parking space does not already exist in the database
2. The parking enforcement officer enters relevant information such as parking space number
3. The database verifies new space is added

Removing a parking space:
1. The parking enforcement officer verifies the parking space exists in the database
2. The parking enforcement officer enters relevant information, such as parking space number
3. The parking space is removed from the database

View a parking space's information
1. The parking enforcement officer enters relevant information such as parking space number
2. Officer now views details such as if the space is occupied or not, by which user, etc in a window

Extensions (or Alternative Flows):
Adding a new parking space:
1. If the parking space already exists in the database, the parking enforcement officer will be asked to enter correct information and try again

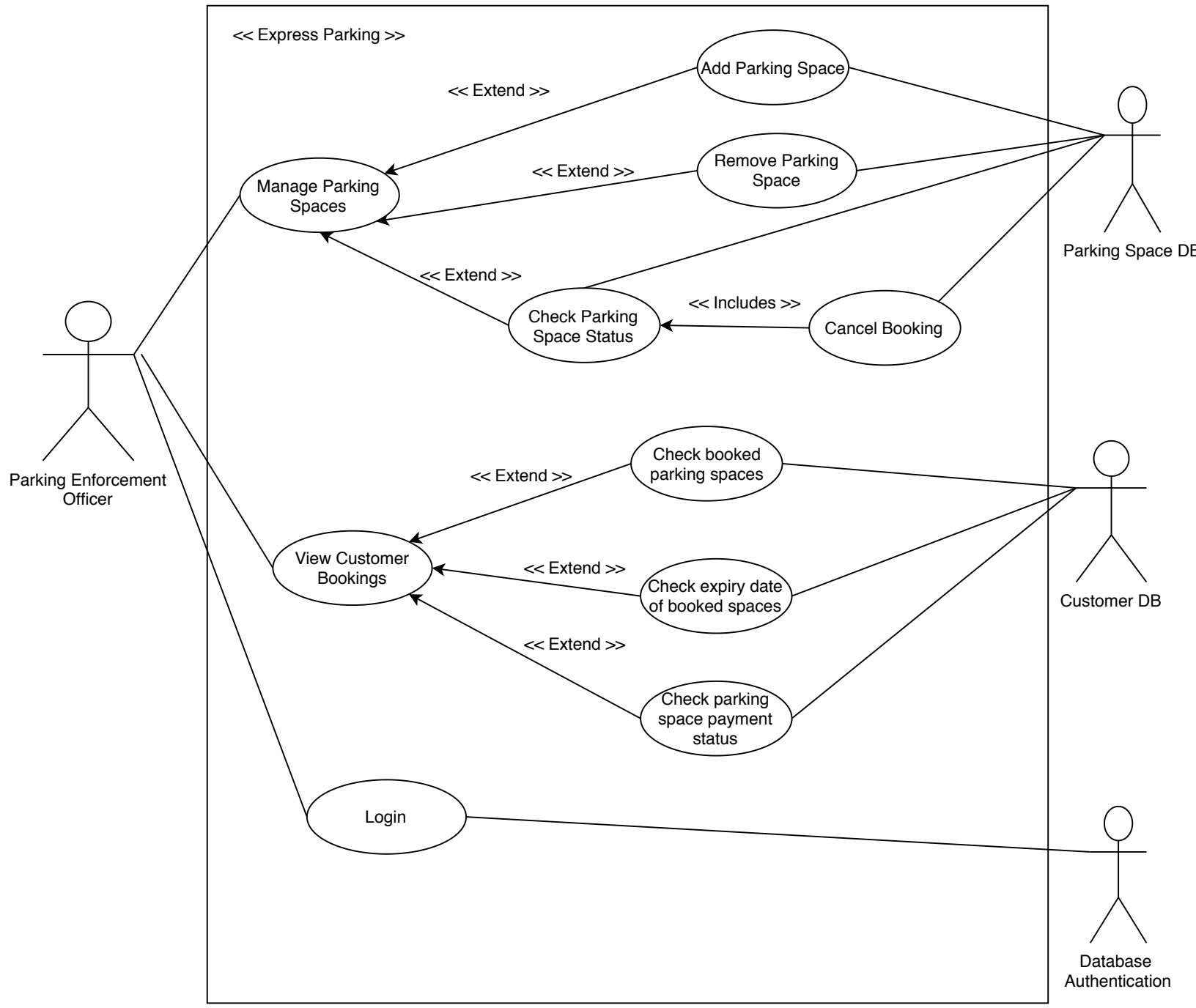
Removing a parking space:
2. If the parking space already does not exist in the database, the parking enforcement officer will be asked to enter correct information and try again

View a parking space's information:
1. If information is entered incorrectly, parking enforcement officer will be asked to enter it again and retry
Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the parking enforcement officer to enter information and to select the different options
- There is a separate button for add, remove, and view information
- The system is robust and can detect false inputs
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc:

Use Case UC3: View Customer Booking
Primary Actor: Parking enforcement officer
Stakeholders and Interests:
- Database Authentication: Wants to ensure that the user is registered and logged into the system
-Toronto Parking Authority: Wants to ensure build statistics such as how many occupied spaces at a time, etc
Preconditions:
- The parking enforcement officer is a real person with a valid ID
- The parking enforcement officer is an employee of the Toronto Parking Authority
Success Guarantee (Postconditions):
The parking enforcement officer can view any customer's booking information
Main Success Scenario (or Basic Flow):
1. The parking enforcement officer enters a customer's information such as name and username
2. The database verifies customer exists
3. The parking enforcement officer can successfully see the customer's bookings and their details such as expiry time, parking space number, etc
Extensions (or Alternative Flows):
2a. If the customer does not exist in the database, the parking enforcement officer will be requested to re-enter the information
Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the customer to enter information and to select the different options
- The system is robust and can detect false customer inputs
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc:

Use Case UC2: Parking Enforcement Officer Authentication
Primary Actor: Parking enforcement officer
Stakeholders and Interests:
- Database Authentication: Wants to ensure that the user is registered and logged into the system
Preconditions:
- The parking enforcement officer is a real person with a valid ID
- The parking enforcement officer is an employee of the Toronto Parking Authority
Success Guarantee (Postconditions):
The parking enforcement officer is logged in to the system
Main Success Scenario (or Basic Flow):
1. The parking enforcement officer enters their ID and password into the system
2. The database authenticates the login information
3. The parking enforcement officer is successfully logged in
Extensions (or Alternative Flows):
1a. If the parking enforcement officer has not been added to the system yet, they will need to click the register button and enter information such as their first name, last name, and email address
2a. If the ID and/or password is incorrect, the parking enforcement officer is requested to enter their login information again
Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the customer to enter information and to select the different options
- The system is robust and can detect false customer inputs
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc:
- What happens if a parking enforcement officer forgets their ID and/or password?

Use-Case diagram for "Parking Enforcement Officer"



Use Case UC1: Manage parking enforcement officer

Primary Actor: System Admin

Stakeholders and Interests:

- System: ensuring all the correct parking enforcement officers have administrative privileges when logging into the system

Preconditions:

For Add:

- Parking enforcement officer is registered as an employee of the Toronto Parking Authority
- Parking enforcement officer is not yet added in the system

For Remove:

-User has parking enforcement officer privileges when logged into system
-User is not registered as a parking enforcement officer (could have been registered as parking enforcement in the past, could be mistakenly given privileges)

Success Guarantee (Postconditions):

- A new parking space is added to the database (addition)
- A parking space is no longer in the database (removal)
- A parking space request is granted or denied

For Add:

- A new parking enforcement officer is added to the database
- The new parking enforcement officer has all the parking enforcement officer privileges including viewing customer bookings and managing parking spaces

For Remove:

-The parking enforcement officer has been removed from the database of parking enforcement officers
- The new parking enforcement officer no longer has all the parking enforcement officer privileges including viewing customer bookings and managing parking spaces

Main Success Scenario (or Basic Flow):

Adding a new parking enforcement officer:

1. The system verifies the new parking enforcement officer does not already exist in the database
2. The system enters relevant information such as employee number (key), name, date of birth etc.

3. The database verifies new space is added

Removing a parking enforcement officer:

1. The system verifies the parking enforcement officer exists in the database
2. The system enters relevant information, such as employee number (key in DB)
3. The parking space is removed from the database
4. The database verifies new space has been removed

Extensions (or Alternative Flows):

Adding a new parking enforcement officer:

1. If the parking enforcement officer already exists in the database, the parking system will say so and will prompt the System Admin to either try again or cancel

Removing a parking enforcement officer:

2. If the parking enforcement officer queried does not exist in the database, the system will say so and will prompt the systems admin to try again or cancel

Special Requirements:

- The system uses a GUI interface with buttons and text boxes for the Systems Admin to enter information and to select the different options
- There is a separate button for add and remove Parking Enforcement Officer
- The system is robust and can detect false inputs

Technology and Data Variations List:

The system operates through user inputs and buttons for each task

Frequency of Occurrence:

Continuous

Misc: N/A

Use Case UC2: Change Payment Status
Primary Actor: System
Stakeholders and Interests:
- System: Displaying which parking spots have been paid for and which parking spots have not been paid for
-Parking Enforcement Officers: Can see if a parking space has been paid for

Preconditions:

-Payment service has verified that parking space(s) x has been paid for for y amount(s) of time
-Parking space x status is "unpaid"
-x is a valid parking space
-y is a valid amount of time

Success Guarantee (Postconditions):
- Parking space x is marked as "paid" for the time-period that the Customer has specified (y)
-Countdown timer on parking space begins from y to 0
-If Countdown timer hits zero, Parking space x is marked as "unpaid" (unless new customer pays for the same spot, in which case timer is reset by the new user)

Main Success Scenario (or Basic Flow):
Adding a new parking enforcement officer:

1. Customer Enters Parking spot(s) they have paid for, their information (eg. name, licence plate), and their credit card info in application
2. The system verifies the customer in the customer database
3. The system verifies the payment information and amount through Payment Services
4. The Transaction takes place(ie. money is transferred from customer bank account to city parking account)
5. The Parking spot entered by the Customer is updated to "paid" for an amount of time that corresponds to the amount the customer paid for the parking
6. The Countdown timer for the spot starts counting down

7. -If No new customer pays for the same parking space before timer hits zero, parking space is marked as unpaid
-If a new customer pays for the same parking space before timer hits zero, process starts all over again

Extensions (or Alternative Flows):
Adding a new parking enforcement officer:
1. If the customer is not in database: customer is prompted to register a new account and login

2. If the payment information is not verified the customer is asked to try again or cancel the transaction
3. If the transaction is not able to take place, the customer is asked to try again or cancel the transaction

4. If a new customer pays for the same parking space before timer hits zero, process starts all over again.

Special Requirements:
- The system uses a GUI interface with buttons and text boxes for the Customer to enter information and to select the different options
- The system is robust and can detect false inputs
Technology and Data Variations List:
The system operates through user inputs and buttons for each task
Frequency of Occurrence:
Continuous
Misc: N/A

Use-Case diagram for "System Administrator"