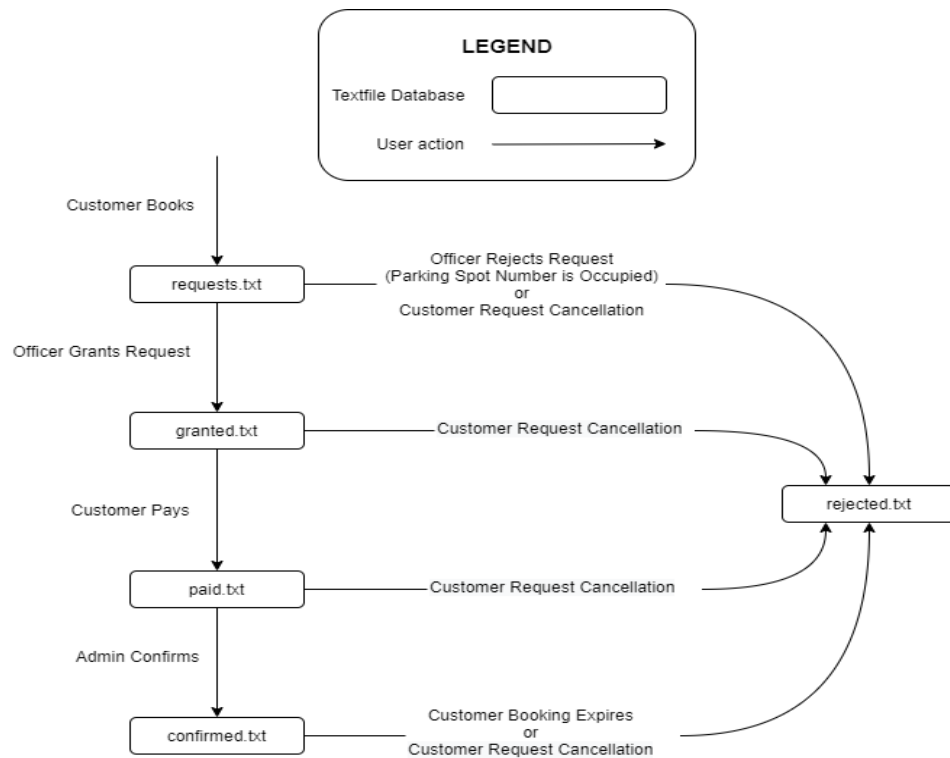


# EECS 3311 – PROJECT REPORT

## Mark Miguel 214588115

### Overview:

The diagram below illustrates the flow of Booking information in the database stored in text files.



### How to Run:

**NOTE:** Each type of user will have a default account and their account information will be stored in customers.txt, officers.txt and admin.txt. The default admin account is admin, admin (username, password). The default customer account is c, c (username, password). The default officer account is o, o (username, password). The default parking spots will consist of 10 vacant parking spots.

**NOTE:** You may have to change the file paths specified in the backend package.

**NOTE:** If eclipse asks to reconfigure the build path. All the jars used in the project is in the jar directory.

Step1: Run the Main class.

Step2: Have fun!

### Test Coverage:

**NOTE:** My application consists of 4 Non-GUI class, and they are all stored in the backend package.

**WARNING:** When running the JUnit test, the database will be wiped out and will reset to its default state. To learn about the details of the reset, please read the method wipeClean() in the Booking class. Please ignore the coverage result of org.eclipse.wb.swt since it contains the GUI classes and was not needed for test coverage. Only the non-GUI classes were tested for code coverage as told by Professor Wang. The image below shows that I have covered at least 80% percent of my non-GUI classes.

@ Javadoc Declaration Console Coverage				
testCoverage (Apr. 22, 2021 9:39:57 p.m.)				
Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
EECS3311-Project	24.0 %	4,756	15,021	19,777
src	24.0 %	4,756	15,021	19,777
org.eclipse.wb.swt	0.0 %	0	14,692	14,692
backend	90.6 %	3,158	329	3,487
Booking.java	87.5 %	2,155	308	2,463
Register.java	95.6 %	366	17	383
Authenticator.java	98.5 %	258	4	262
ParkingSpace.java	100.0 %	379	0	379
tests	100.0 %	1,598	0	1,598

### Changes:

There are a few changes I have made from the initial design in the midterm. Firstly, I have added several GUI classes. This is because I wanted to make my application user friendly and easier to understand. Moreover, the GUI classes in the initial design was too simple and so the new design needed some GUI classes to make the overall application user friendly. Secondly, I have removed the BookingsList class since we were told to use csv or texfiles to store our database. I have also removed the PayPal, CreditCard and DebitCard class since the Professor assumed that all forms of payments were valid. Thus, with the assumption of payment validity, I thought implementing the PayPal, CreditCard and DebitCard class were no longer necessary. I have also modified the manageParkingEnforcementOfficer class by merging it with Booking class. This is because the manageParkingEnforcementOfficer methods are identical with the methods that were implemented in the Booking class. In other words, it would have been redundant to implement methods with the same purpose on the two classes. Overall, the changes from the initial design were done to make the application user friendly and easy to understand. Moreover, I believe these changes were necessary since there were hurdles from the initial design, that would not make sense in a real-world application of the system.

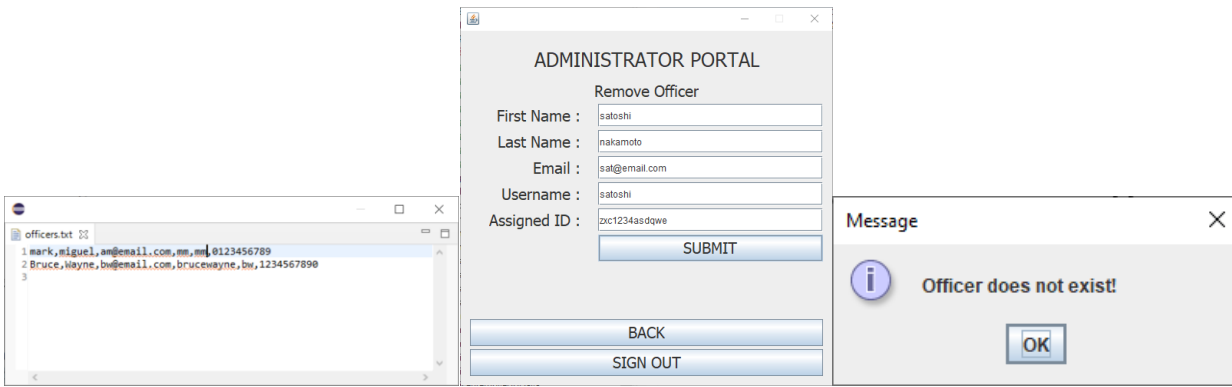
### Requirements:

The following are images consisting of data to broadly illustrate how each requirement is implemented. Please run the application as it is easier to understand how each requirement is implemented.

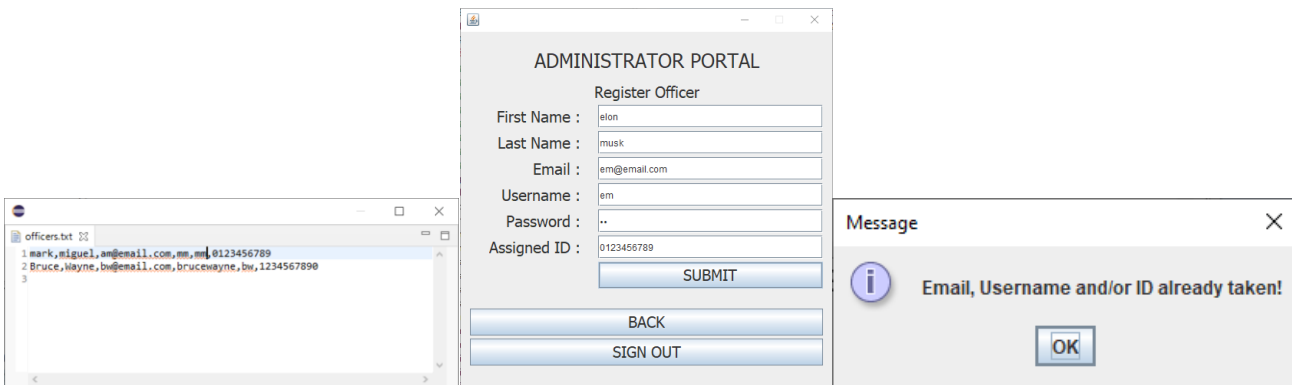
4.1.3-REQ-1: The system must assign a unique ID to each new parking enforcement officer added

The screenshots illustrate the implementation of requirement 4.1.3-REQ-1. The first window shows the Administrator login screen. The second window shows the Administrator Portal with options to register, remove, manage payment status, and view bookings. The third window shows the Register Officer form with fields for First Name, Last Name, Email, Username, Password, and Assigned ID. The fourth window shows the Administrator Portal with the registered officer's details. A message box at the bottom indicates 'Registration successful!'.

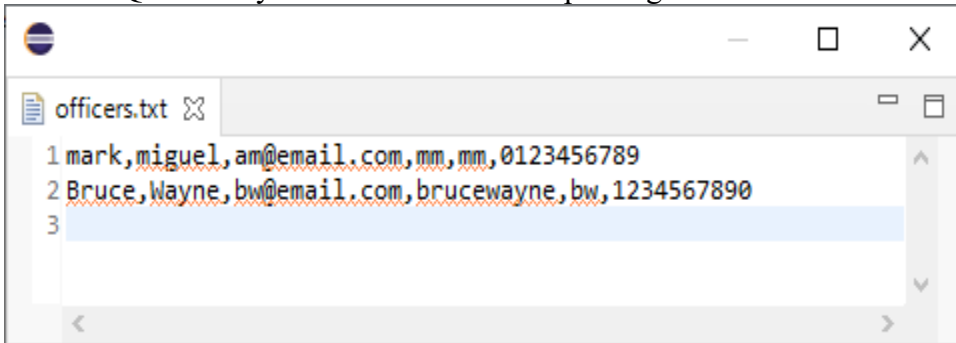
4.1.3-REQ-2: The system must verify the parking enforcement officer exists in the system before removing an officer



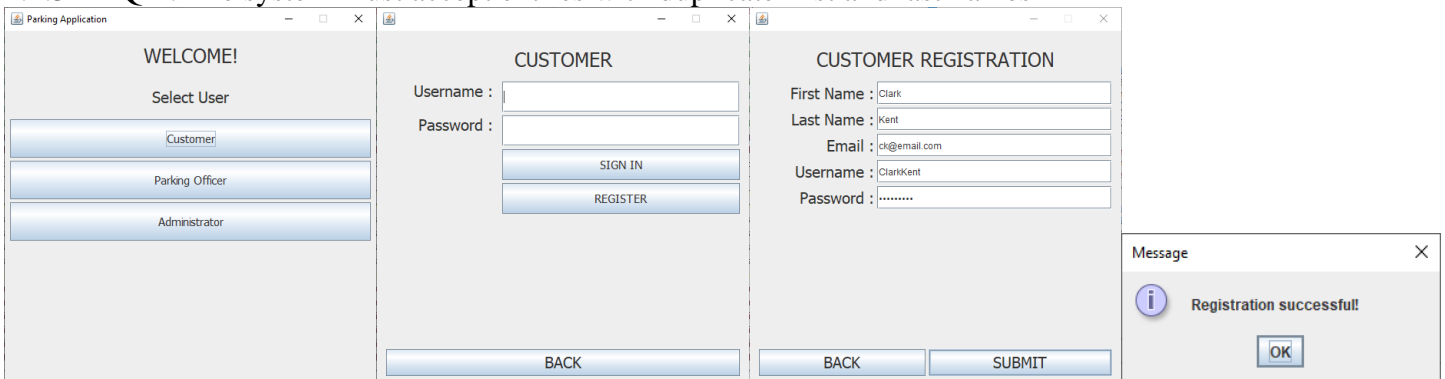
4.1.3-REQ-3: The system must verify a new parking enforcement officer's ID does not exist in the system already

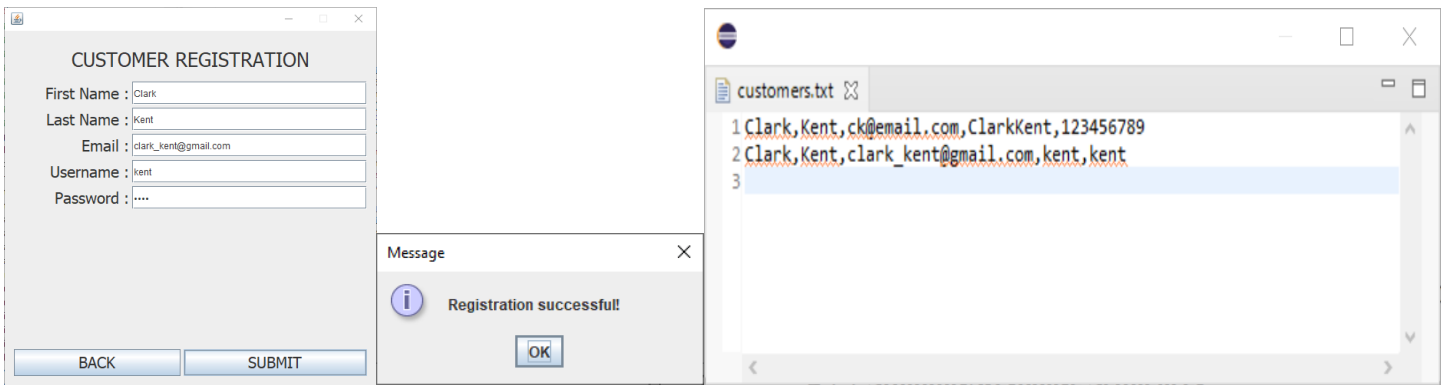


4.1.3-REQ-4: The system must store a new parking enforcement officer's registration information

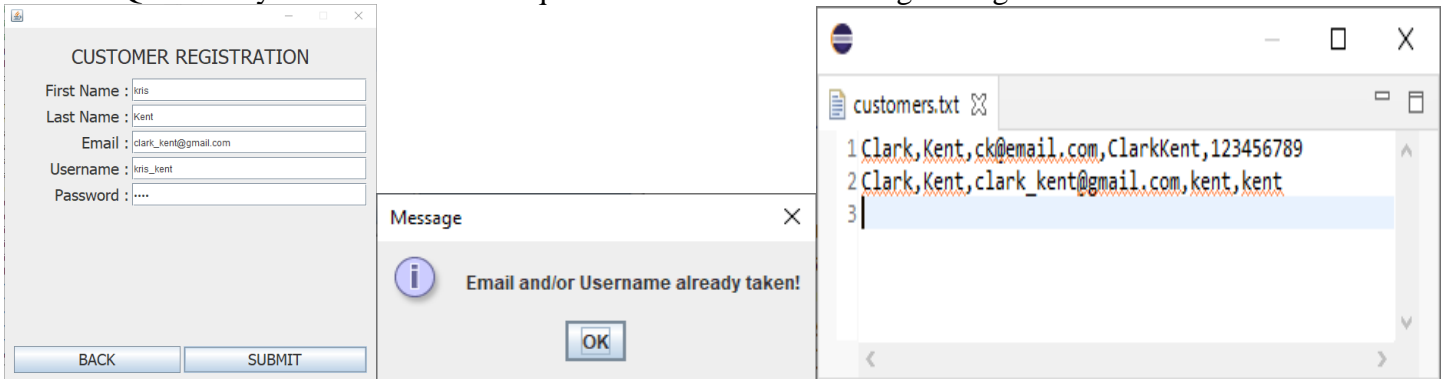


4.2.3-REQ-1: The system must accept entries with duplicate first and last names

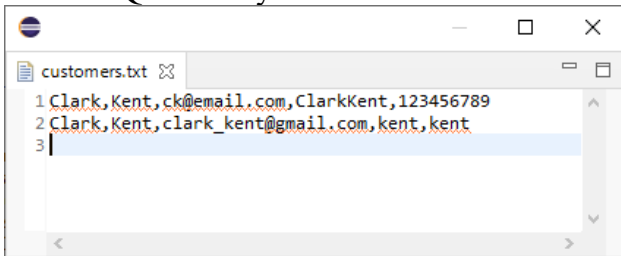




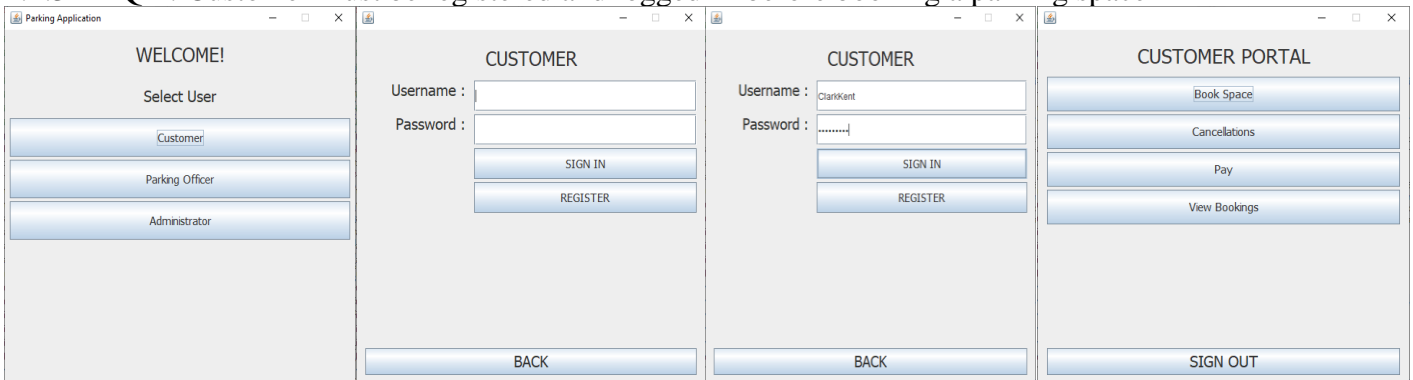
4.2.3-REQ-2: The system must allow unique email addresses when registering



4.2.3-REQ-3: The system must store a new user's login information for future authentication purposes



4.4.3-REQ-1: Customer must be registered and logged-in before booking a parking space



4.4.3-REQ-2: Customer must select which parking space they are booking

**CUSTOMER PORTAL**  
Book Space

Parking Space Number :

Expiry Date & Time : yyyy-MM-dd HH:mm:ss

License Plate Number :

4.4.3-REQ-3: If the customer selects a parking space which is occupied, they are presented with an error message and must select a new space

parkingspots.txt

- 1 1,vacant
- 2 2,vacant
- 3 3,vacant
- 4 4,vacant
- 5 5,vacant
- 6 6,occupied
- 7

**CUSTOMER PORTAL**  
Book Space

Parking Space Number : 6

Expiry Date & Time : 2021-04-24 16:00:00

License Plate Number : 60THAM

**Message**

Space successfully requested! Please wait for an Officer to grant your request!

**PARKING OFFICER**

Username : mm

Password : -

**OFFICER PORTAL**

**OFFICER PORTAL**  
Manage Parking

**OFFICER PORTAL**  
Manage Parking  
Grant Request

Booking ID : 81114e92

**Message**

Parking spot number is Occupied! Customer request rejected!

**CUSTOMER**

Username : ClerkKent

Password : .....

**CUSTOMER PORTAL**

**CUSTOMER PORTAL**  
View Bookings

Current Bookings :





REMOVED REQUESTS (Check The Reason for Details):

Parking Number: 6    Expiry Time: 2021-04-24 16:00:00    License Plate: 60THAM    Booking ID: 81114e92    Reason: Parking Spot Number is Occupied!

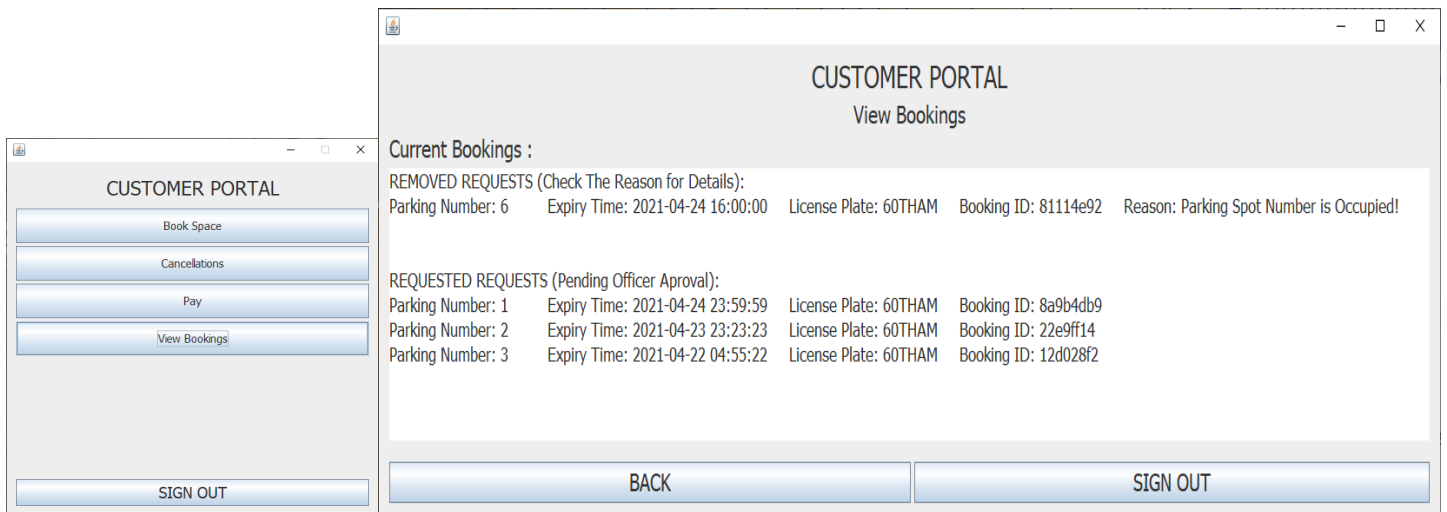
4.4.3-REQ-4: Customer must enter how long they want to book said parking space for

CUSTOMER PORTAL		
Book Space		
Parking Space Number :	<input type="text"/>	
Expiry Date & Time :	<input type="text" value="yyyy-MM-dd HH:mm:ss"/>	
License Plate Number :	<input type="text"/>	
<input type="button" value="SUBMIT"/>		
<input type="button" value="BACK"/>		<input type="button" value="SIGN OUT"/>

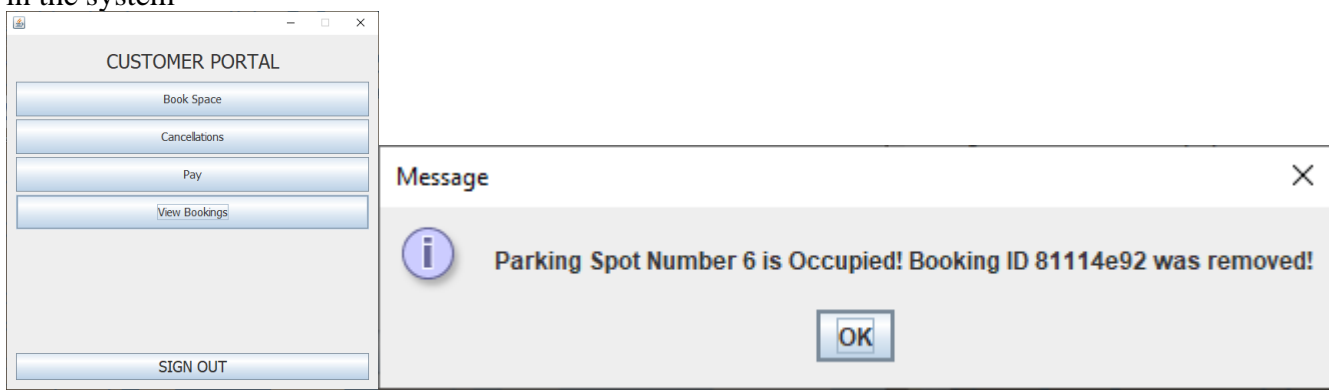
4.4.3-REQ-5: The system allows a customer to book up to three parking spaces

<div> <div>CUSTOMER PORTAL</div> <div>Book Space</div> <div>Parking Space Number : <input type="text" value="1"/></div> <div>Expiry Date &amp; Time : <input type="text" value="2021-04-24 23:59:59"/></div> <div>License Plate Number : <input type="text" value="60THAM"/></div> <div><input type="button" value="SUBMIT"/></div> <div><input type="button" value="BACK"/></div> <div><input type="button" value="SIGN OUT"/></div> </div>	<div> <div>Message</div> <div> <b>Space successfully requested! Please wait for an Officer to grant your request!</b></div> <div><input type="button" value="OK"/></div> </div>
<div> <div>CUSTOMER PORTAL</div> <div>Book Space</div> <div>Parking Space Number : <input type="text" value="2"/></div> <div>Expiry Date &amp; Time : <input type="text" value="2021-04-23 23:23:23"/></div> <div>License Plate Number : <input type="text" value="60THAM"/></div> <div><input type="button" value="SUBMIT"/></div> <div><input type="button" value="BACK"/></div> <div><input type="button" value="SIGN OUT"/></div> </div>	<div> <div>Message</div> <div> <b>Space successfully requested! Please wait for an Officer to grant your request!</b></div> <div><input type="button" value="OK"/></div> </div>
<div> <div>CUSTOMER PORTAL</div> <div>Book Space</div> <div>Parking Space Number : <input type="text" value="3"/></div> <div>Expiry Date &amp; Time : <input type="text" value="2021-04-22 04:55:22"/></div> <div>License Plate Number : <input type="text" value="60THAM"/></div> <div><input type="button" value="SUBMIT"/></div> <div><input type="button" value="BACK"/></div> <div><input type="button" value="SIGN OUT"/></div> </div>	<div> <div>Message</div> <div> <b>Space successfully requested! Please wait for an Officer to grant your request!</b></div> <div><input type="button" value="OK"/></div> </div>
<div> <div>CUSTOMER PORTAL</div> <div>Book Space</div> <div>Parking Space Number : <input type="text" value="4"/></div> <div>Expiry Date &amp; Time : <input type="text" value="2021-04-21 23:59:59"/></div> <div>License Plate Number : <input type="text" value="60THAM"/></div> <div><input type="button" value="SUBMIT"/></div> <div><input type="button" value="BACK"/></div> <div><input type="button" value="SIGN OUT"/></div> </div>	<div> <div>Message</div> <div> <b>Customers can only book up to 3 parking spaces!</b></div> <div><input type="button" value="OK"/></div> </div>

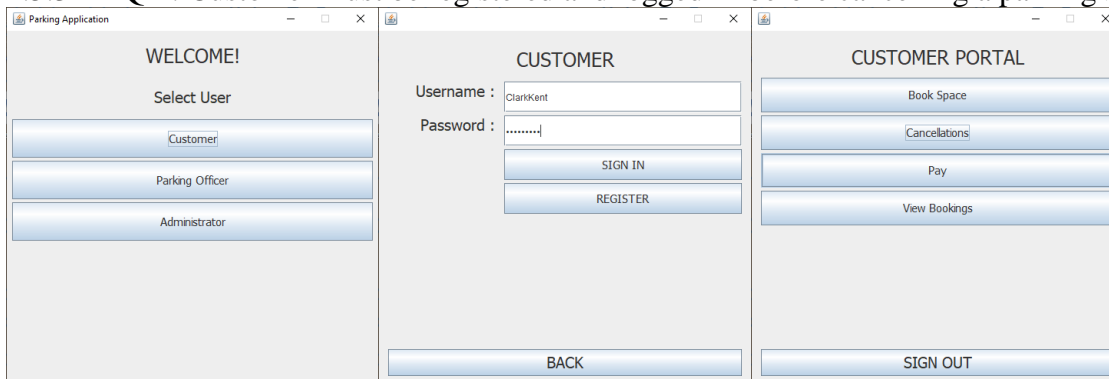
4.4.3-REQ-6: Each booked parking space receives a unique booking ID



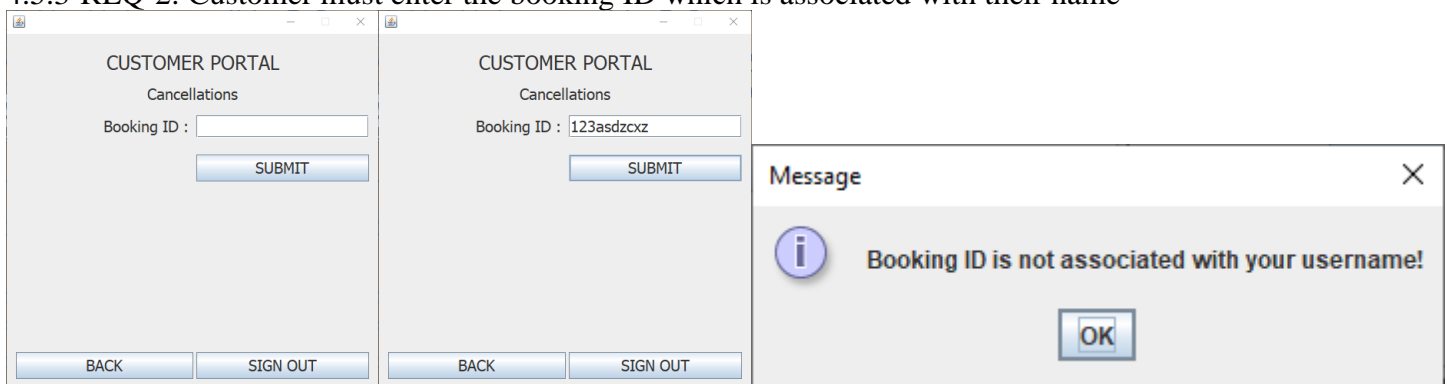
4.4.3-REQ-7: The system must display an error message if the parking space the customer booked is occupied in the system



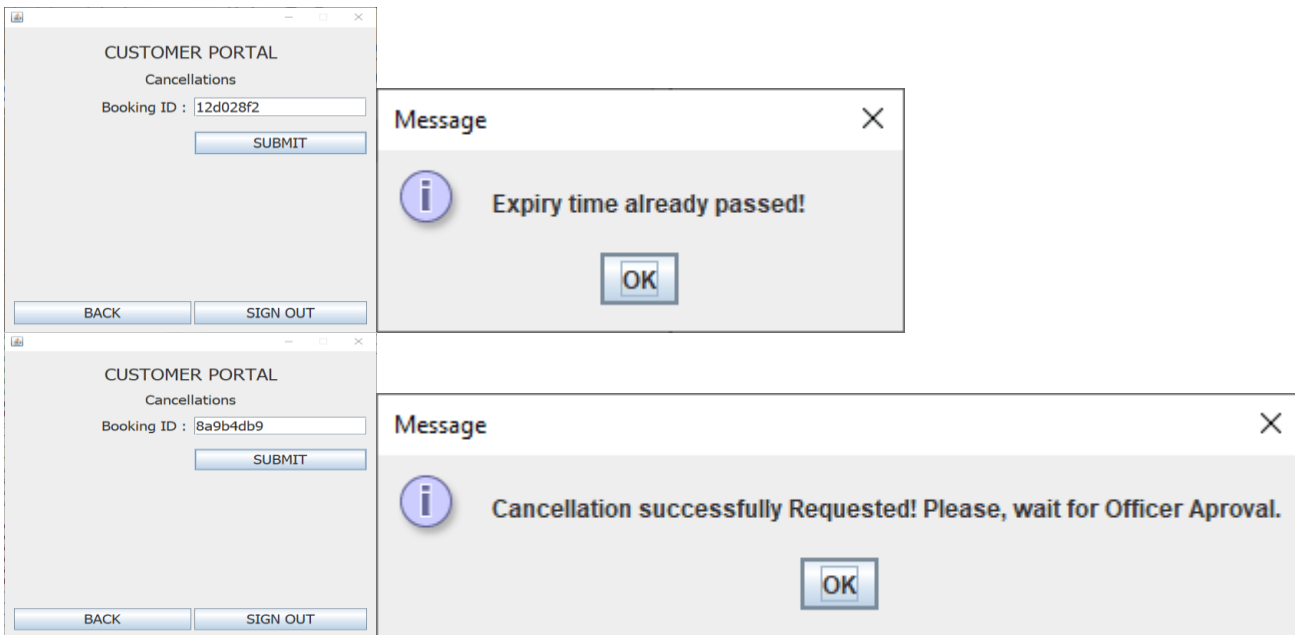
4.5.3-REQ-1: Customer must be registered and logged-in before cancelling a parking space



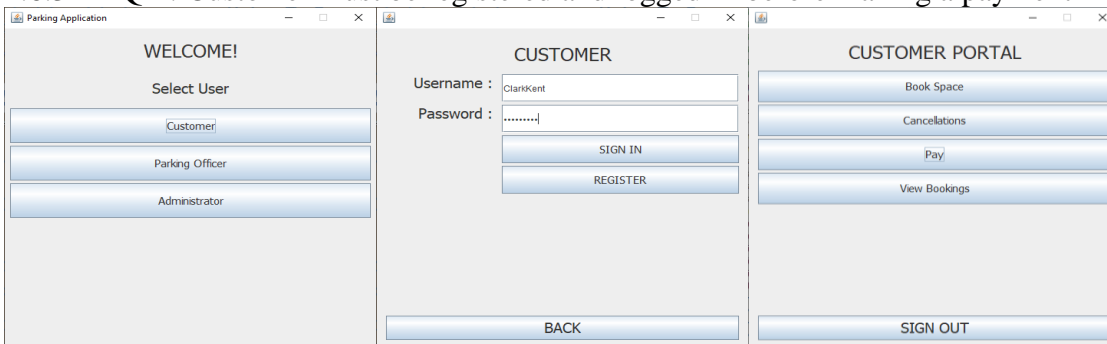
4.5.3-REQ-2: Customer must enter the booking ID which is associated with their name



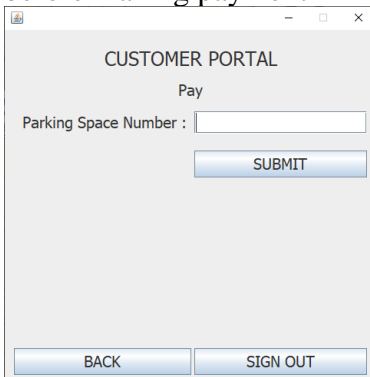
4.5.3-REQ-3: The cancellation can only go through if the time of cancellation is before the booking expiry time



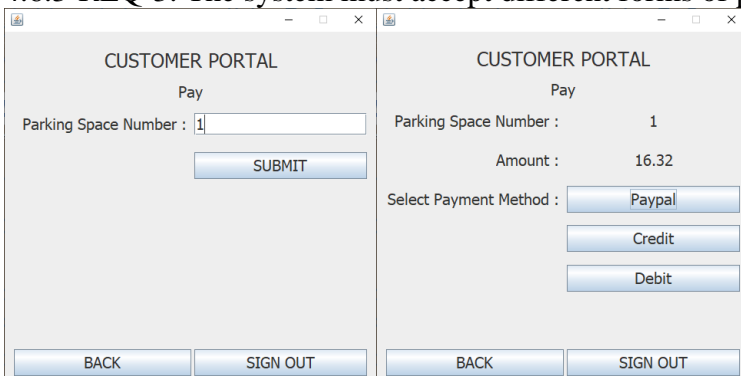
4.6.3-REQ-1: Customer must be registered and logged-in before making a payment



4.6.3-REQ-2: Customer must have entered additional information such as which parking space they are booking before making payment



4.6.3-REQ-3: The system must accept different forms of payment (ex: Paypal, credit, debit, etc.)



4.6.3-REQ-4: The system must automatically timestamp each payment confirmation



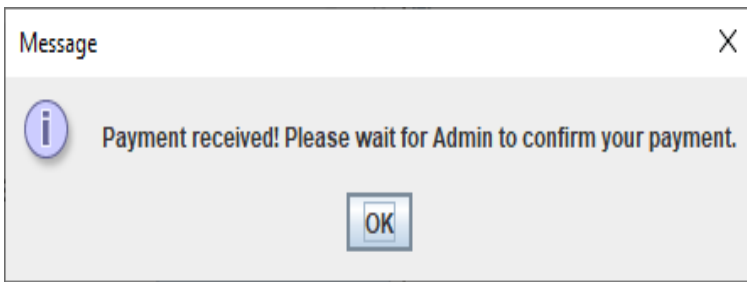
4.6.3-REQ-5: The system automatically starts a countdown till expiry once payment is confirmed  
4.6.3-REQ-4 and 4.6.3-REQ-5 requirement is already automatically met and implemented by my software design.

Instead of users inputting the length of time they want for a specific parking spot, my application system instead asks for the expiry date and time (format: yyyy-MM-dd HH:mm:ss) they want for the particular parking spot. Therefore, the system does not need to timestamp each booking after payment confirmation and does not need to start a countdown till expiry. Consequently, my design system consistently tracks the current time in comparison to the expiry time inputted by the customer. More specifically, the methods beforeExpiry(String booking) and afterExpiry(String booking) in the class Booking.java tracks and checks if the time in the customer’s booking is before or after the current time. My system design then allows customers to book however long they want, provided the expiry time inputted is a valid future time.

4.6.3-REQ-6: The system allows a customer to pay for multiple parking space bookings (if any) cumulatively

Customer Portal - Pay	Customer Portal - Pay	Customer Portal - Paying With PayPal	Message
<p>Parking Space Number : 1</p> <p>SUBMIT</p> <p>BACK SIGN OUT</p>	<p>Parking Space Number : 1</p> <p>Amount : 15.86</p> <p>Select Payment Method :  Paypal  Credit  Debit</p> <p>BACK SIGN OUT</p>	<p>Parking Spot Number : 1</p> <p>Amount : 15.86</p> <p>Email : assuming_valid</p> <p>Password : *****</p> <p>SIGN IN</p> <p>BACK SIGN OUT</p>	<p>Payment received! Please wait for Admin to confirm your payment.</p> <p>OK</p>
<p>Parking Space Number : 2</p> <p>SUBMIT</p> <p>BACK SIGN OUT</p>	<p>Parking Space Number : 2</p> <p>Amount : 14.56</p> <p>Select Payment Method :  Paypal  Credit  Debit</p> <p>BACK SIGN OUT</p>	<p>Parking Space Number : 2</p> <p>Amount : 14.56</p> <p>Name on card : Professor Said</p> <p>Card number : assume valid</p> <p>Expiry date (MM/YY) : 0530</p> <p>Security code (CVC) : 999</p> <p>SUBMIT</p> <p>BACK SIGN OUT</p>	<p>Payment received! Please wait for Admin to confirm your payment.</p> <p>OK</p>
<p>Parking Space Number : 3</p> <p>SUBMIT</p> <p>BACK SIGN OUT</p>	<p>Parking Space Number : 3</p> <p>Amount : 11.51</p> <p>Select Payment Method :  Paypal  Credit  Debit</p> <p>BACK SIGN OUT</p>	<p>Parking Space Number : 3</p> <p>Amount : 11.54</p> <p>Name on card : assume valid</p> <p>Card number : assume valid</p> <p>Expiry date (MM/YY) : 1225</p> <p>Security code (CVC) : 999</p> <p>SUBMIT</p> <p>BACK SIGN OUT</p>	<p>Payment received! Please wait for Admin to confirm your payment.</p> <p>OK</p>

4.6.3-REQ-7: The system must authenticate customer payment information before proceeding with confirmation of parking space



#### 4.7.3-REQ-1: Only authorized users, such as parking enforcement officers , can view any customer's booking details

The screenshot displays the Parking Application interface. It features a 'WELCOME! Select User' dialog with buttons for 'Customer', 'Parking Officer', and 'Administrator'. The 'Parking Officer' window shows a login form with 'Username' (brucewayne) and 'Password' fields, and 'SIGN IN' and 'BACK' buttons. The 'OFFICER PORTAL' window has 'Manage Parking' and 'View Bookings' buttons, with 'SIGN OUT' and 'BACK' buttons at the bottom. The 'OFFICER PORTAL View Bookings' window shows 'Current Bookings' and 'PAID REQUESTS (Pending Admin Confirmation)' sections. The 'OFFICER PORTAL View Bookings' window also has 'View Current Bookings' and 'View Specific Customer' buttons, with 'SIGN OUT' and 'BACK' buttons at the bottom. The 'CUSTOMER PORTAL View Bookings' window shows 'ClarkKent's Current Bookings' and 'PAID REQUESTS (Pending Admin Confirmation)' sections, with 'SIGN OUT' and 'BACK' buttons at the bottom.

#### 4.7.3-REQ-2: Customers can view their parking space booking information, including expiry time

**WELCOME!**  
Select User

Customer  
Parking Officer  
Administrator

**CUSTOMER**  
Username : ClarkKent  
Password : .....  
SIGN IN  
REGISTER  
BACK

**CUSTOMER PORTAL**  
Book Space  
Cancellations  
Pay  
View Bookings  
SIGN OUT

**CUSTOMER PORTAL**  
View Bookings

Current Bookings :

REMOVED REQUESTS (Check The Reason for Details):  
Parking Number: 6    Expiry Time: 2021-04-24 16:00:00    License Plate: 60THAM    Booking ID: 81114e92    Reason: Parking Spot Number is Occupied!

PAID REQUESTS (Pending Admin Confirmation):  
Parking Number: 1    Expiry Time: 2021-04-18 23:59:59    License Plate: 60THAM    Booking ID: 8a9b4db9  
Parking Number: 2    Expiry Time: 2021-04-18 23:23:23    License Plate: 60THAM    Booking ID: 22e9ff14  
Parking Number: 3    Expiry Time: 2021-04-18 21:55:22    License Plate: 60THAM    Booking ID: 12d028f2

BACK    SIGN OUT

4.7.3-REQ-3: The system must notify the customer when their parking space booking is expired

**Message**

Booking ID 12d028f2 has expired! Please vacate Parking Spot Number 3 Thank You!

OK

4.8.3-REQ-1: The user must be an authenticated parking enforcement officer.

**WELCOME!**  
Select User

Customer  
Parking Officer  
Administrator

**PARKING OFFICER**  
Username : brucewayne  
Password : ..  
SIGN IN  
BACK

**OFFICER PORTAL**  
Manage Parking  
View Bookings  
SIGN OUT

**OFFICER PORTAL**  
Manage Parking  
Add Space  
Remove Space  
Cancel Request  
Grant Request  
BACK    SIGN OUT

4.8.3-REQ-2: Parking enforcement officers must verify a parking space is vacant before removing it from the system

**OFFICER PORTAL**  
Manage Parking  
Add Space  
Remove Space  
Cancel Request  
Grant Request  
BACK    SIGN OUT

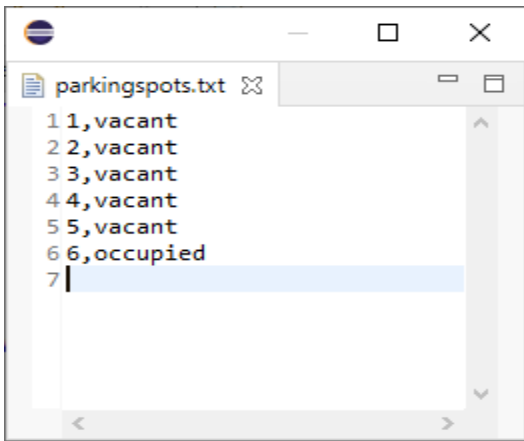
**OFFICER PORTAL**  
Manage Parking  
Remove Space  
Parking Space Number : 6  
SUBMIT  
BACK    SIGN OUT

**Message**

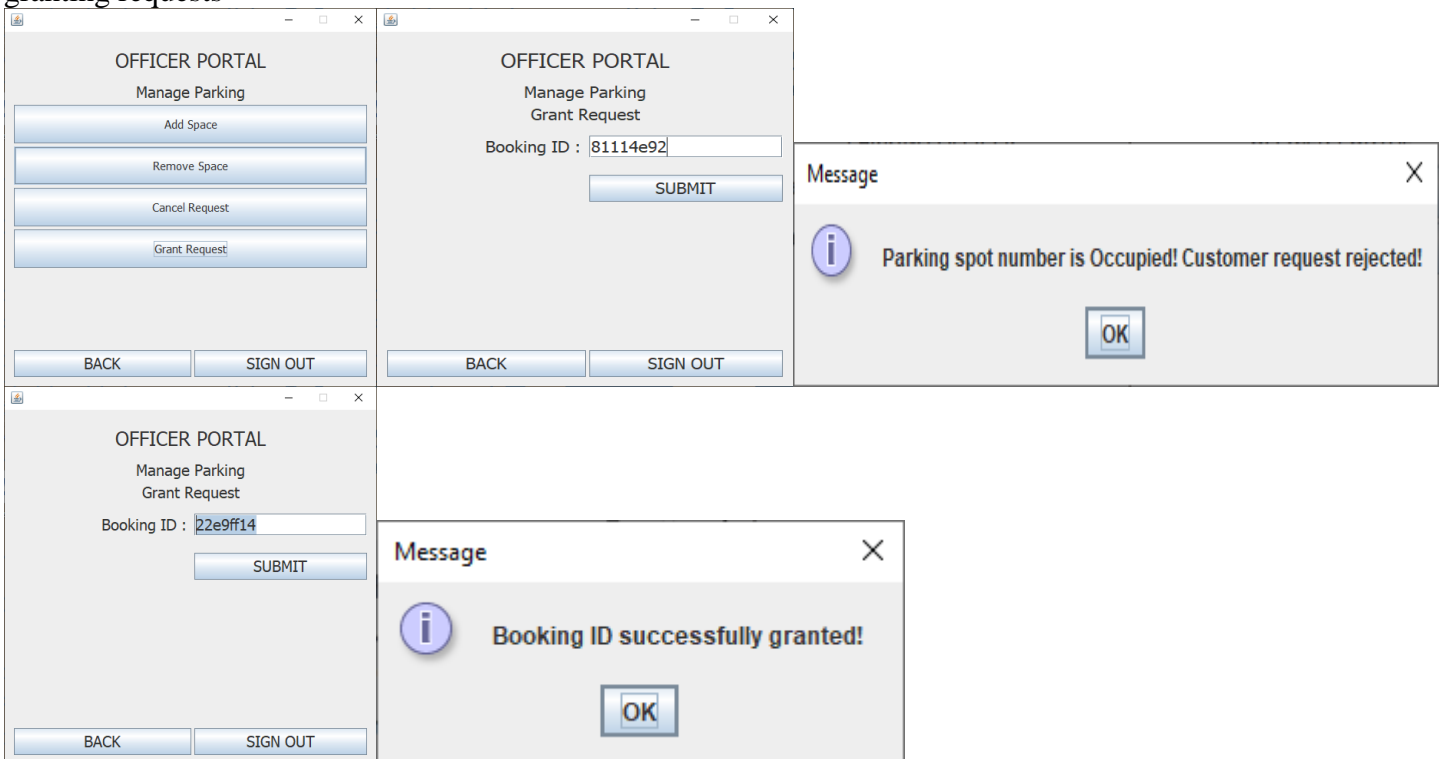
Parking Space is Occupied!

OK

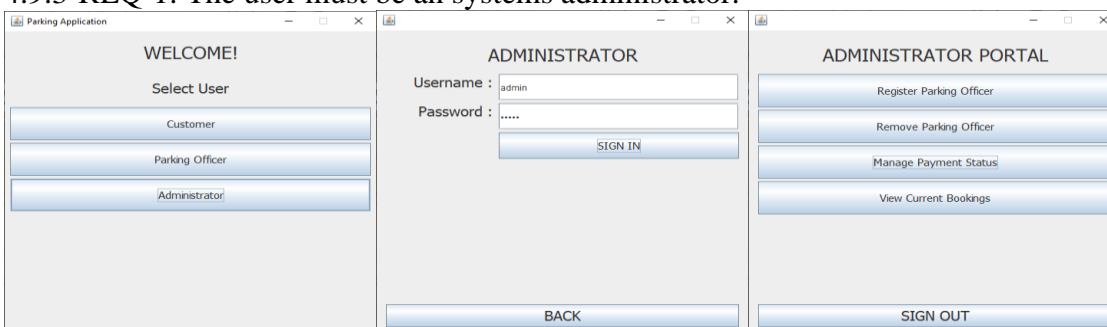
4.8.3-REQ-3: The system must have a minimum of one parking space



4.8.3-REQ-4: Parking enforcement officers must verify requested parking space is vacant in the system before granting requests



4.9.3-REQ-1: The user must be an systems administrator.



4.9.3-REQ-2: The system must verify the customer's existence before changing their payment status.

ADMINISTRATOR PORTAL  
Manage Payment Status  
First Name :  
Last Name :  
Email :  
Parking Space Number :  
SUBMIT  
BACK  
SIGN OUT

ADMINISTRATOR PORTAL  
Manage Payment Status  
First Name :  
Last Name :  
Email :  
Parking Space Number :  
SUBMIT  
BACK  
SIGN OUT

Message  
Invalid Customer info or Request does not exist or Customer has not paid yet!  
OK

4.9.3-REQ-4: The system must verify the customer’s payment of the parking space before changing their payment status

ADMINISTRATOR PORTAL  
Manage Payment Status  
First Name :  
Last Name :  
Email :  
Parking Space Number :  
SUBMIT  
BACK  
SIGN OUT

ADMINISTRATOR PORTAL  
Manage Payment Status  
First Name :  
Last Name :  
Email :  
Parking Space Number :  
SUBMIT  
BACK  
SIGN OUT

Message  
Request's payment successfully confirmed!  
OK