Alberto Gonzalez Klein

Miguel Arconada Manteca

alberto.gonzalezk@estudiante.uam.es
miguel.arconada@estudiante.uam.es

TEST PLAN AND TEST RESULTS

Application: <Hounting>

DATE: May 11th, 2018

1. TEST CASE *"PAY BOOKED OFFER"*

1.1 Use Case: Pay booked offer

<Copy here the use case description to be tested>
<Indicate clearly the path to be tested (main, alternative)>

1.2. Test case design (including expected inputs and outputs)**:**

Preconditions:

- The user has logged in with a guest account.
- The guest must have booked the offer previously.
- The user is currently in the *Personal Zone* window, in the *Guest* tab

Scenario:

1. The user presses the *Booked offers* button
2. A new window with one booked offer pops up
3. The user selects the offer
4. The guest presses the PAY button in their booked offers list.
5. The system checks if the guest that is trying to pay is the one who booked the offer.
6. The system calculates the amount of money for each part involved in this operation (deposit and money).
7. A new window pops up with a summary of the purchase
8. The guest confirms the operation.
9. The payment system checks if the credit card associated with that user is not fake.
10. The payment system accepts the operation.
11. The payment system distributes the money according to what it was calculated before.
12. A confirmation message appears
13. The user clicks on *OK* button
14. The offer disappears from the table

Extensions (Alternative paths):

2.1. The guest decides not to accept the operation.

- 2.1.1. A message of "Operation cancelled correctly" pops up
- 2.1.2. The offer is not removed from the table

2.2. The credit card associated with the guest is incorrect.

- 2.2.1. An error message pops up
- 2.2.2. The system cancels the operation.
- 2.2.3. The guest is banned from the system and automatically logs out
- 2.2.4. The window is closed
- 2.2.5. A new login window appears

2.3. The reservation has expired

- 2.3.1. The offer does not appear on the table
- 2.3.2. The system puts the user in the list of users that cannot book or purchase this offer
- 2.3.3. If the user tries to purchase the offer an error message appears pops up

2.4. The user has not selected any offers from the table

2.4.1. An error message pops up prompting the user to select one

1.3. Test execution result

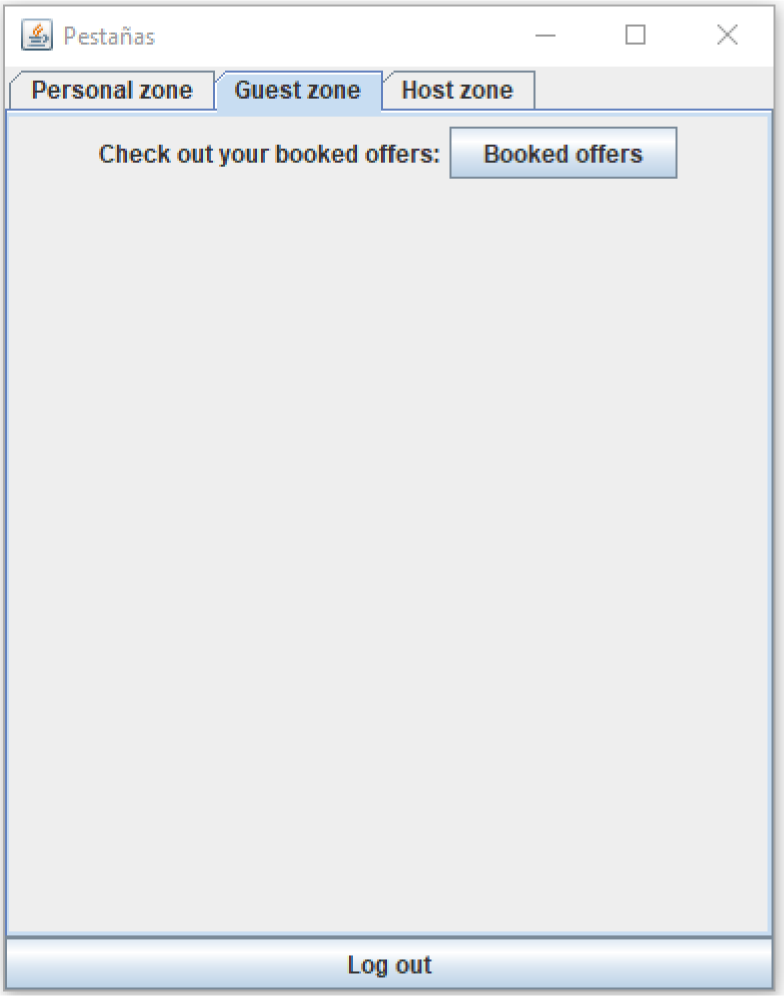
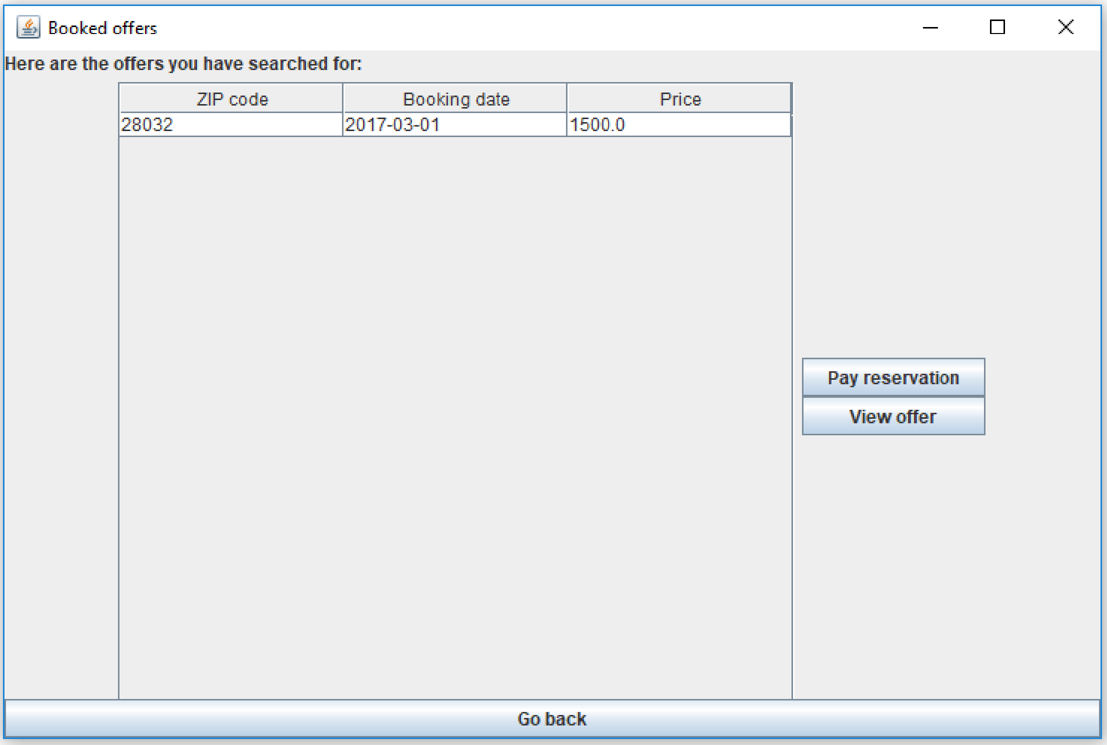


Figure 1:

Pressing the *Booked offers* button. The window containing the offers booked by the logged user will pop.

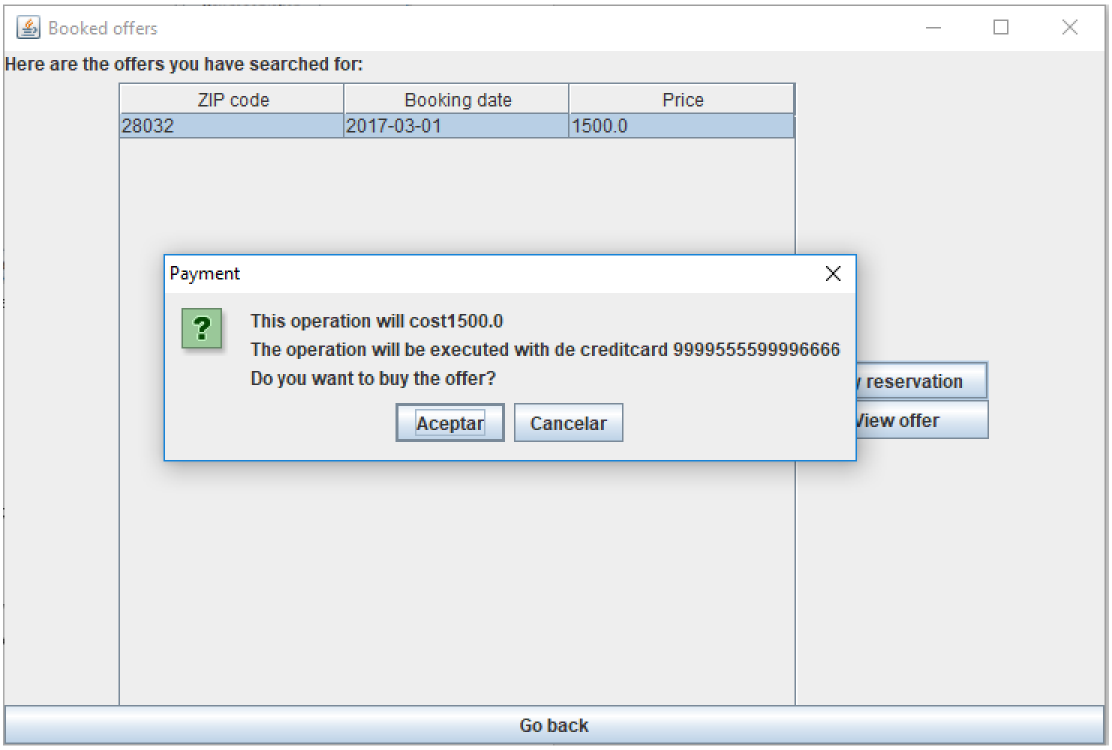


Here are the offers you have searched for:

ZIP code	Booking date	Price
28032	2017-03-01	1500.0

Figure 2:

I select the offer showed and press the *Pay reservation* button. This will lead me to a confirmation window showing the information of the payment.



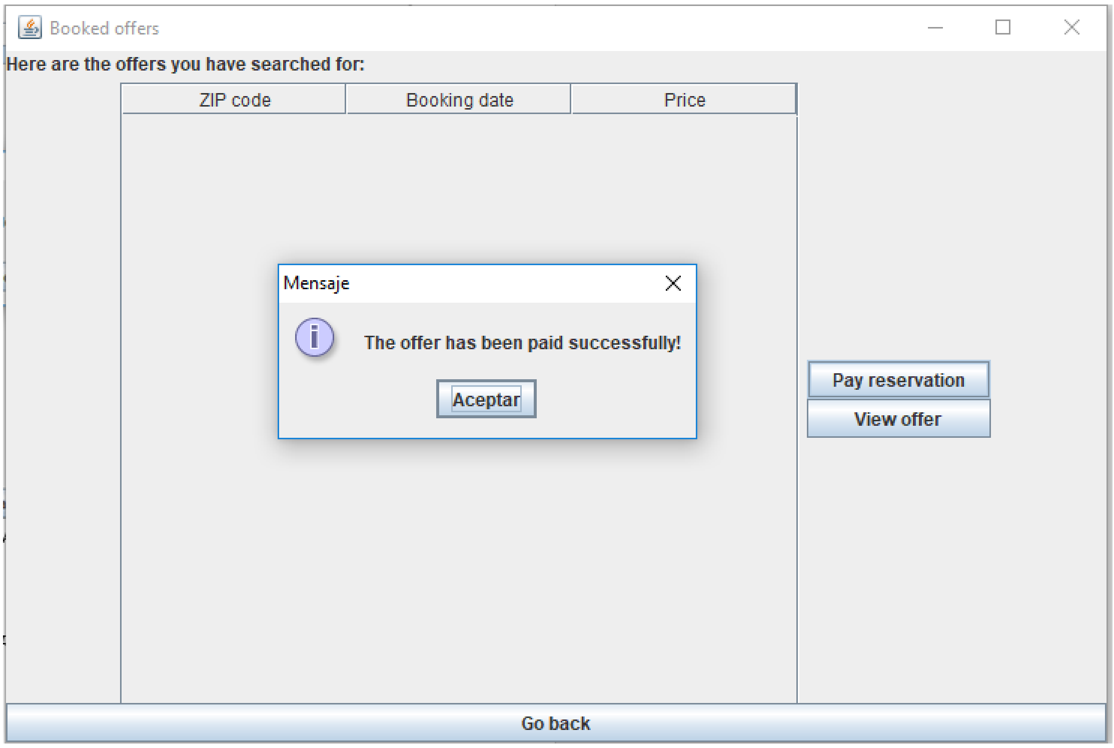
Here are the offers you have searched for:

ZIP code	Booking date	Price
28032	2017-03-01	1500.0

This operation will cost1500.0
The operation will be executed with de creditcard 9999555599996666
Do you want to buy the offer?

Figure 3:

I press *Aceptar*, concluding the payment and deleting the offer from the table. A confirmation message will show up



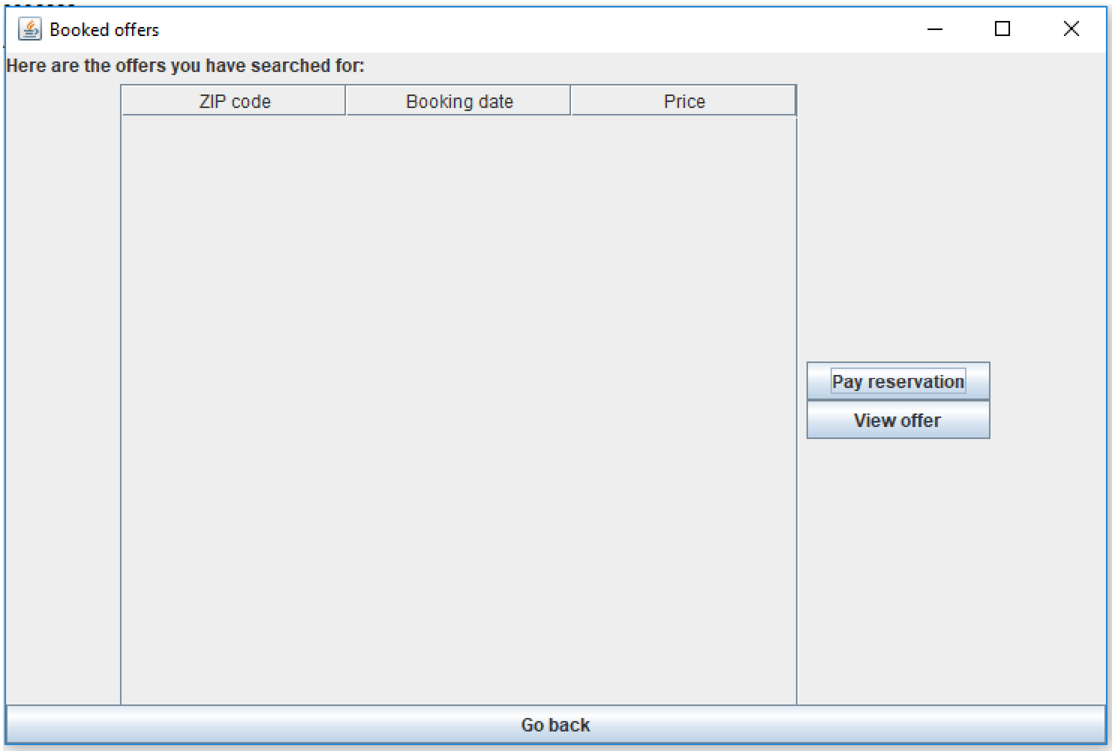
Here are the offers you have searched for:

ZIP code	Booking date	Price

The offer has been paid successfully!

Figure 4:

Pressing *Aceptar* will quit the confirmation message, allowing me to navigate through the system.



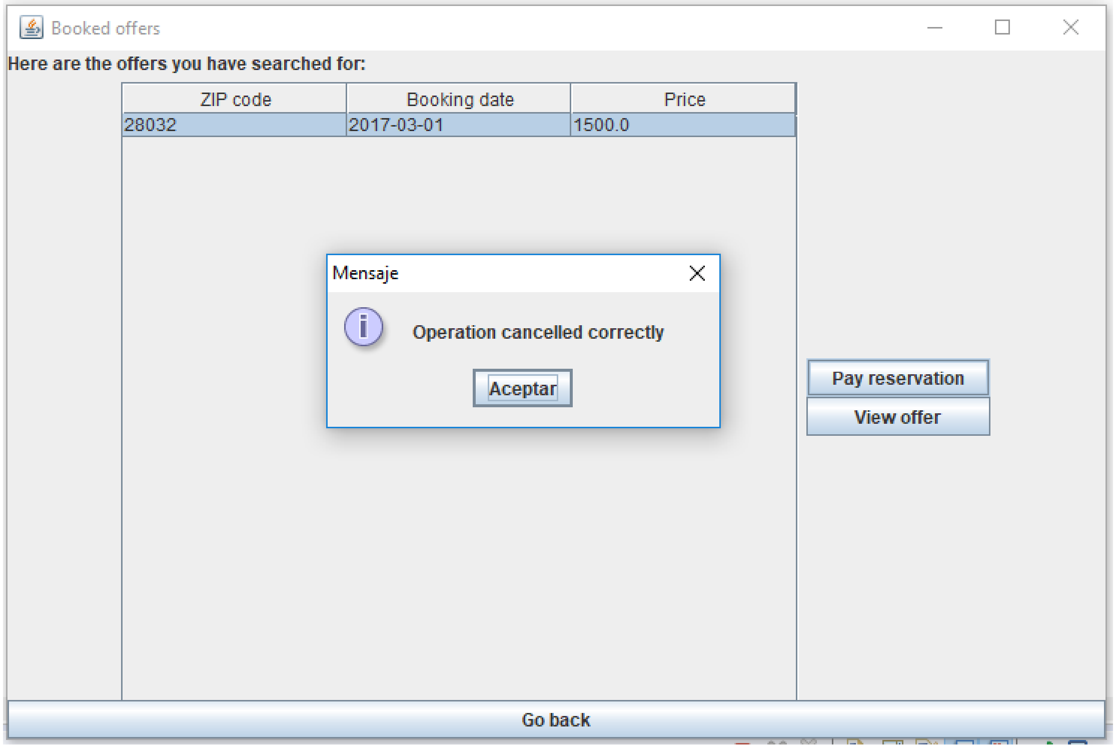
Here are the offers you have searched for:

ZIP code	Booking date	Price

Figure 5:

This is the result of the use case.

Alternative paths:



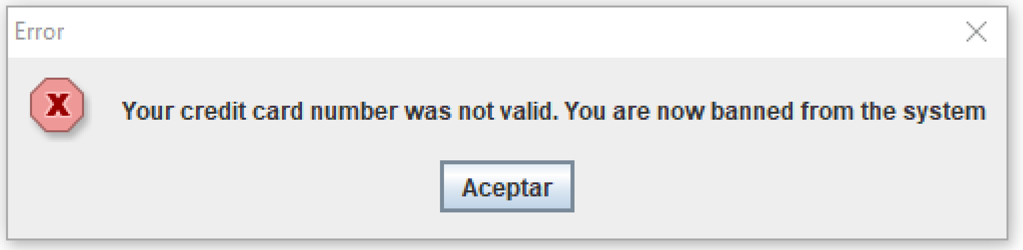
Here are the offers you have searched for:

ZIP code	Booking date	Price
28032	2017-03-01	1500.0

Operation cancelled correctly

Figure 6:

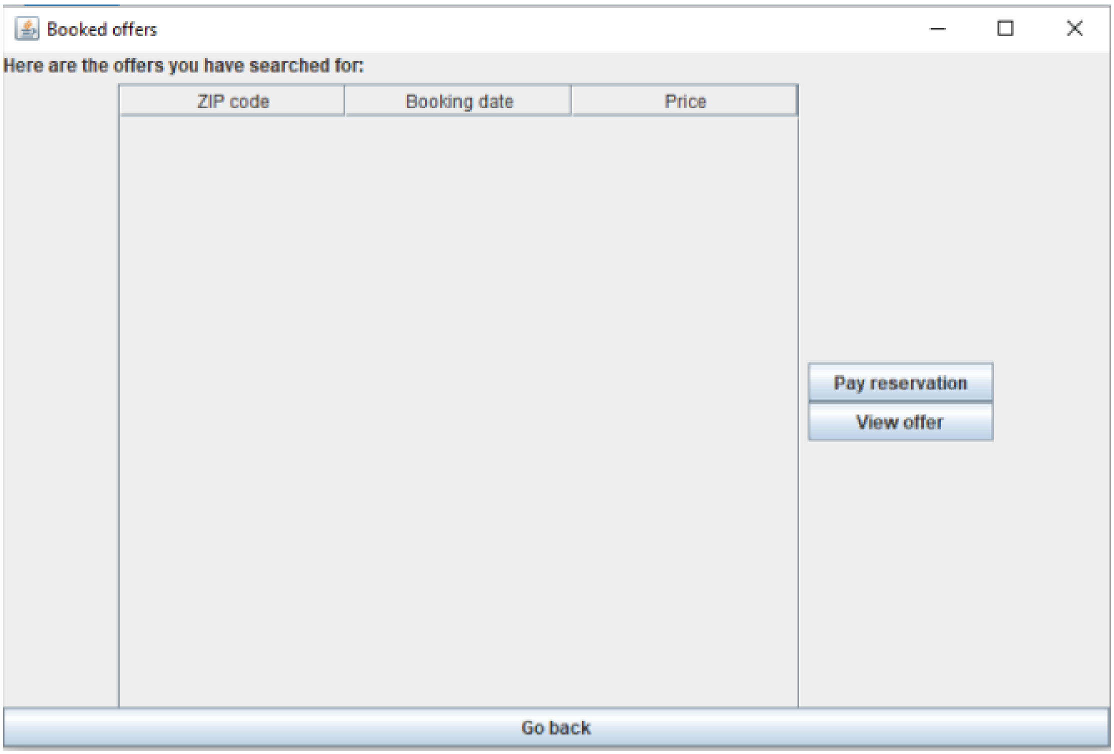
In the figure 3, if we press the *Cancelar* button instead of *Aceptar*, this informative message will show up. As you can appreciate, the offer has not been removed from the table.


Error

Your credit card number was not valid. You are now banned from the system

Figure 7:

In the figure 3, if we press the *Aceptar* button and we have an invalid credit card number, this message shows up. Then, if we press *Aceptar,* we will go to the login window.

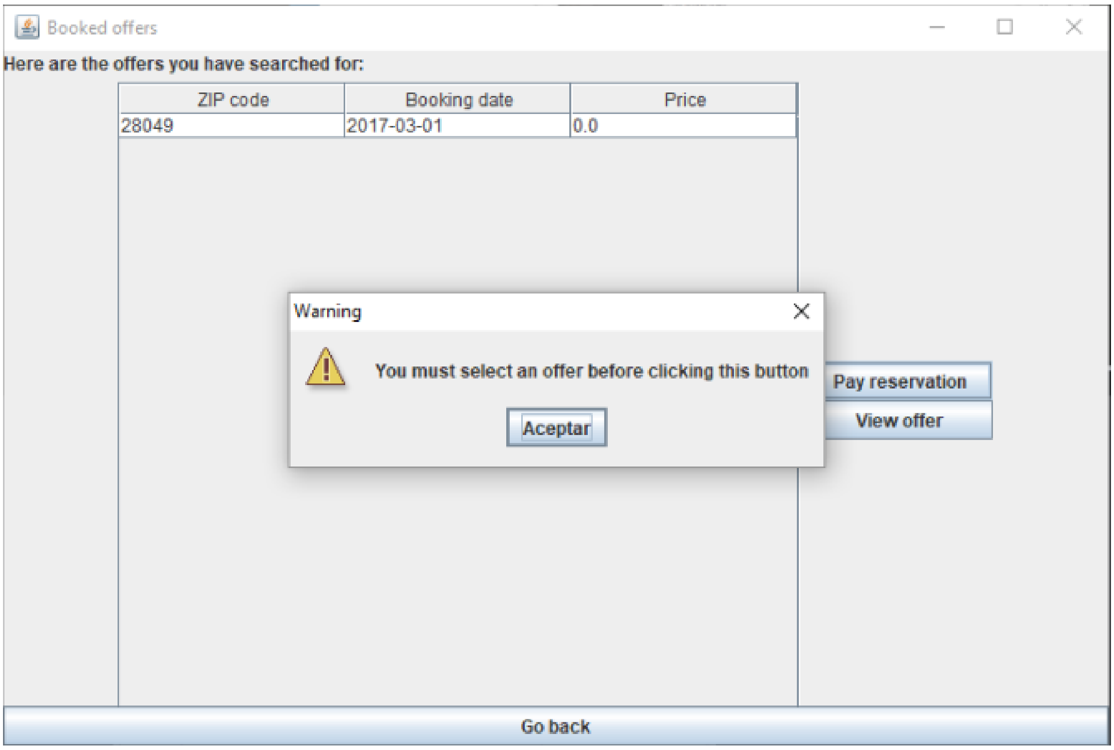

Booked offers

Here are the offers you have searched for:

ZIP code	Booking date	Price

Figure 8:

In Figure 1, if you press the *Booked offers* button and your reservation has expired, no offer will show up.



Here are the offers you have searched for:

ZIP code	Booking date	Price
28049	2017-03-01	0.0

You must select an offer before clicking this button

Figure 9:

In Figure 2, if we press the *Pay reservation* button and we do not select an offer before, this warning will show up.

Every path of the use case has been considered and implemented. If no errors are made, the requirement is satisfied. On the other hand, if an error is made, the program will stop it from causing major trouble.

2. TEST CASE *"CREATE A HOLIDAY OFFER"*

1.2 Use Case: Create a holiday offer

<Copy here the use case description to be tested>
<Indicate clearly the path to be tested (main, alternative)>

1.2. Test case design (including expected inputs and outputs)**:**

Preconditions:

- The user must be logged in as a host or multirole user
- A house must have been previously created

Scenario:

1. The host presses the CREATE OFFER button.
2. A message pops up prompting the user to select the type of offer
3. The user selects "HOLIDAY" and clicks OK
4. A new message pops up asking the user for the offer data
5. Selects the ZIP code of a house that has been created previously.
6. The user fills the required data:
    a. Writes a simple description.
    b. Selects a starting date.
    c. Writes an amount of money as price

d. Chooses an amount of money to be paid as deposit.
e. Enters the duration in days of the offer
f. Selects the ZIP code of an existing house

7. The host presses the CREATE OFFER button.
8. The system saves the offer and marks it as pending for changes, for the administrator to decide whether to approve it, deny it or suggest some changes.
9. A confirmation window pops up

Extensions (Alternative paths):

1.1. The host decides to press the CANCEL button in the first message
- 1.1.1. A message of "Operation cancelled successfully" pops up
- 1.1.2. The system returns to its previous state

1.2. The host presses the cancel button in the second pop up
- 1.2.1. A message of "Operation cancelled successfully" pops up
- 1.2.2. The system returns to its previous state

1.3. The user does not previously own any houses
- 1.3.1. An error message pops up
- 1.3.2. The operation is cancelled

1.4. The user enters an incorrect starting date
- 1.4.1. An error message pops up
- 1.4.2. The operation starts again

1.5. The user enters either a deposit, price or duration which is not numerical
- 1.5.1. An error message pops up
- 1.5.2. The operation restarts

1.6. The user leaves one field blank
- 1.6.1. A warning message pops up

1.7. The user selects "LIVING"
- 1.7.1. A window with different information pops up.
- 1.7.2.

1.3. Test execution result

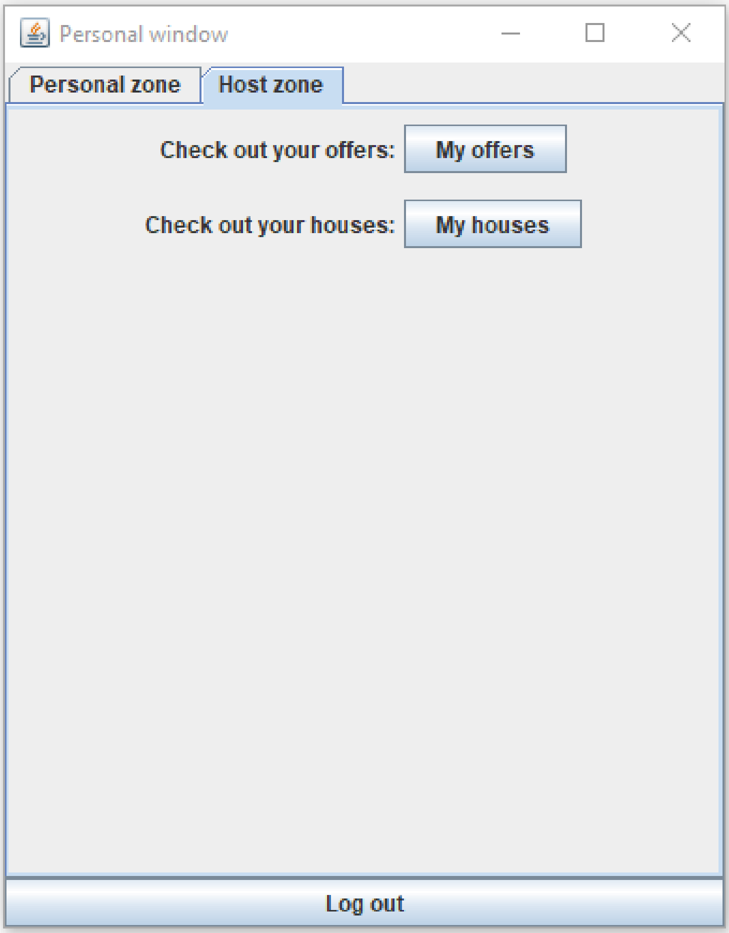
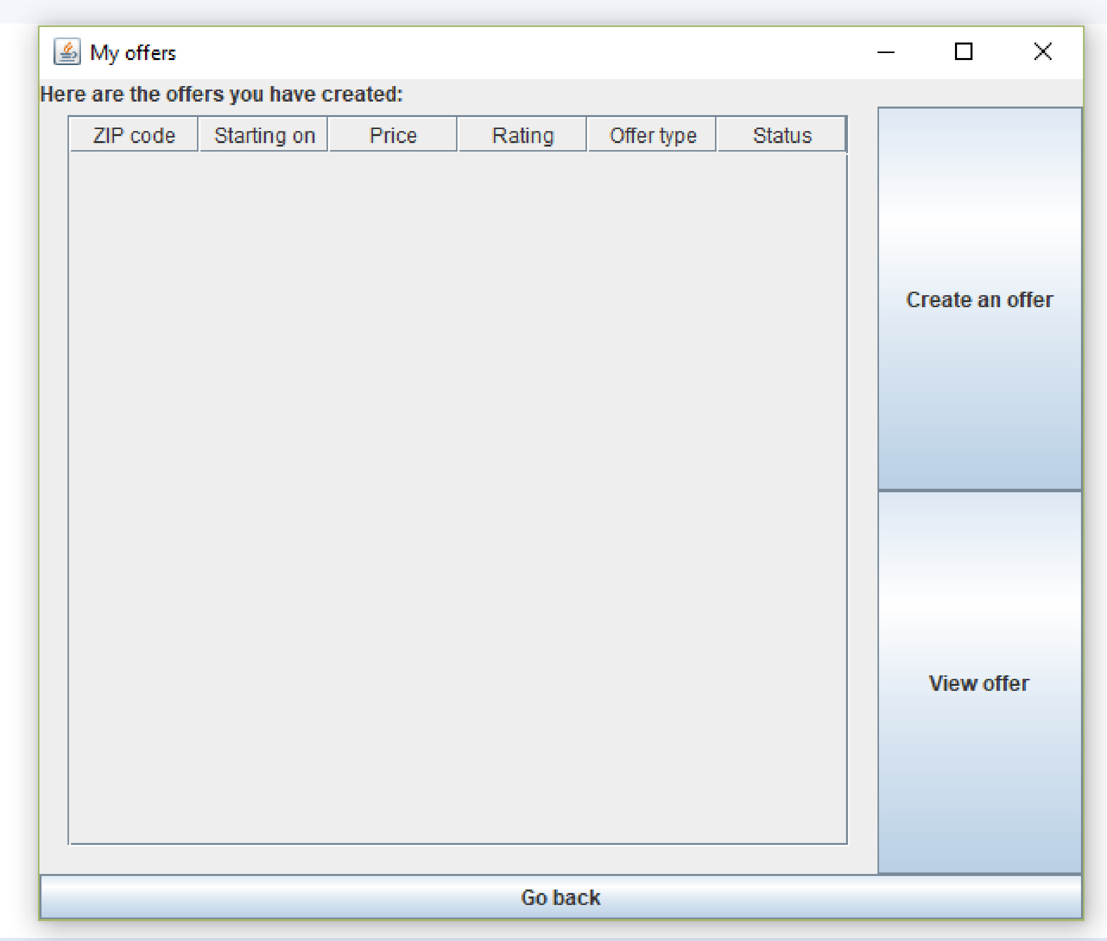


Figure 1:
We click on *My offers.* This will show a new window containing the offers created by the logged user and the option to create a new offer.

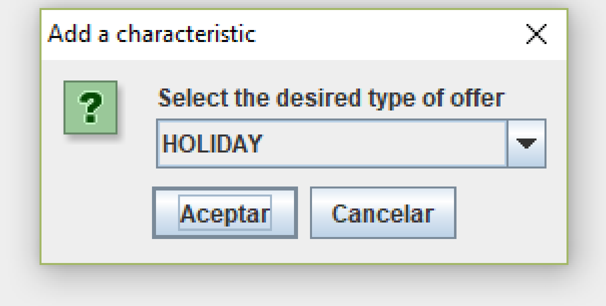


Here are the offers you have created:

ZIP code	Starting on	Price	Rating	Offer type	Status

Figure 2:

We click *Create an offer.* This will ask us for the type of offer we want to create.

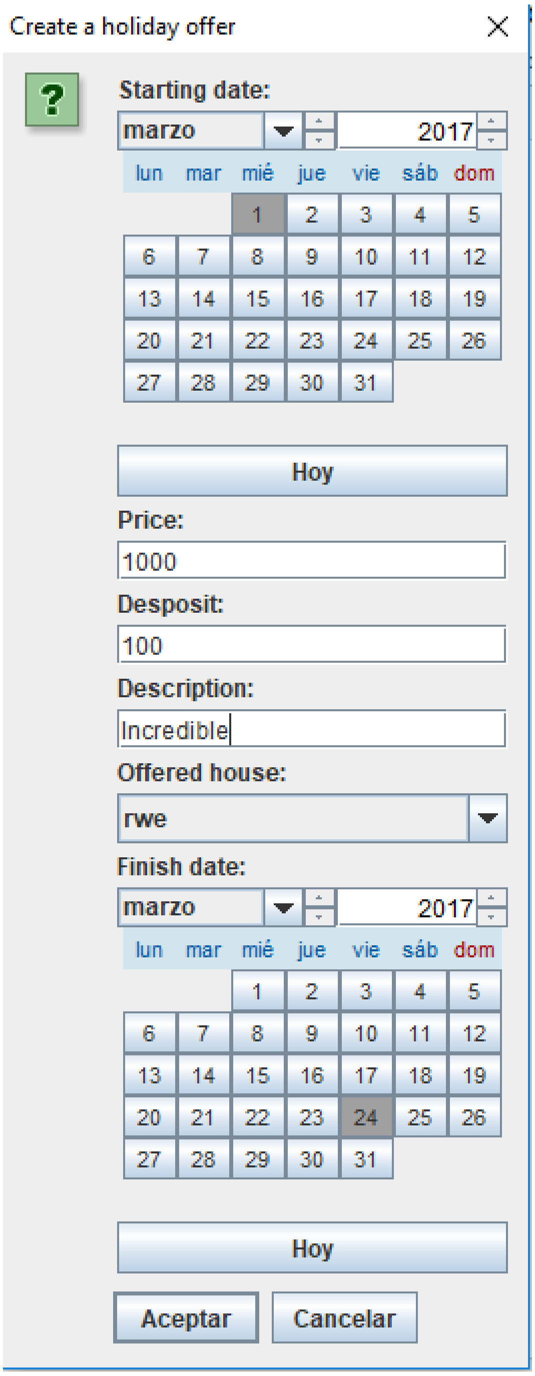


Select the desired type of offer

HOLIDAY

Figure 3:

We select the option *HOLIDAY*. This will show a new window asking us for the information about the house.



Starting date:
marzo 2017

lun	mar	mié	jue	vie	sáb	dom
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Price:
1000

Desposit:
100

Description:
Incredible

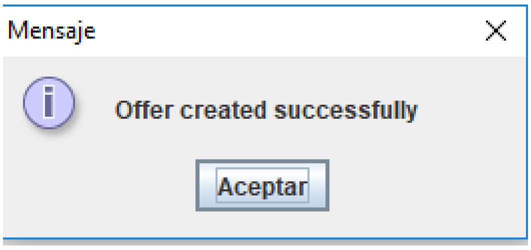
Offered house:
rwe

Finish date:
marzo 2017

lun	mar	mié	jue	vie	sáb	dom
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Figure 4:

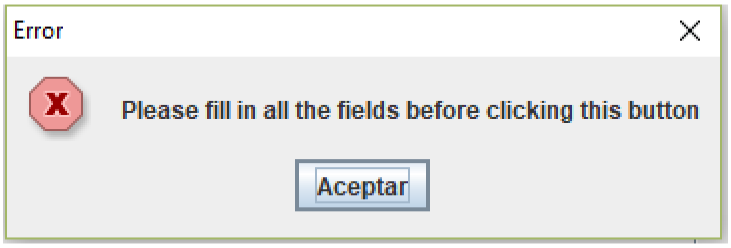
We fill the required information and then click the *Aceptar* button. An informative message will show up.



Offer created successfully

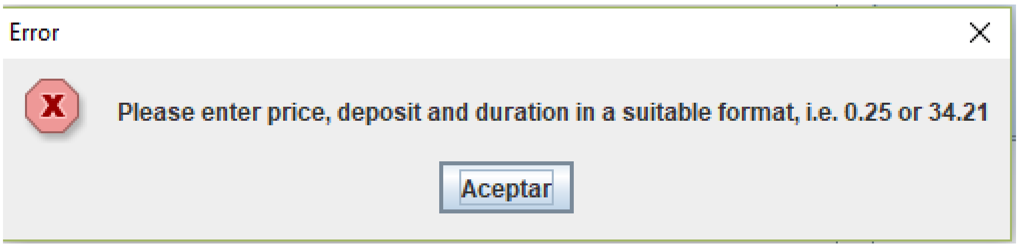
Figure 5:
The offer has been created.

Alternative paths:



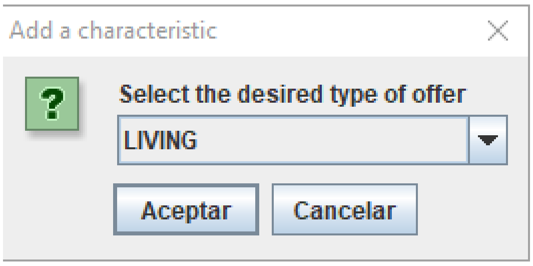
Please fill in all the fields before clicking this button

Figure 6:
In figure 4, if we click *Aceptar* before filling all the fields, this error will show up.



Please enter price, deposit and duration in a suitable format, i.e. 0.25 or 34.21

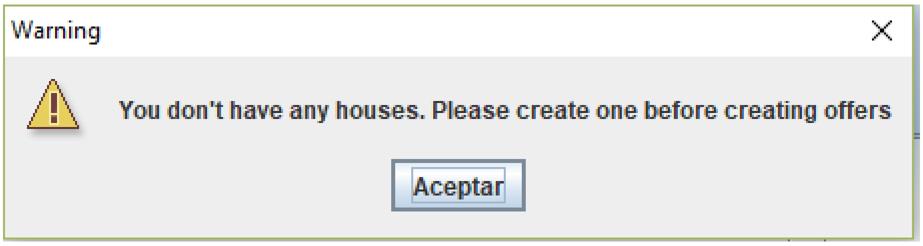
Figure 7:
In figure 4, if we click *Aceptar* when price, duration or deposit have invalid characters (like letters), this error will show up.



Select the desired type of offer

LIVING

Figure 8:
In figure 3, we can select the *LIVING* option. If we do so, we will create a living offer instead of a holiday offer.



Warning

You don't have any houses. Please create one before creating offers

Figure 9:
In figure 2, if we click *Create an offer* before having any houses created this warning will show up.

As we can appreciate, the functionality of the use case is satisfied. After coding, we have discovered new alternative paths that we did not consider at first. All the errors are indicated by an informative message.