***** JAVA MINI PROJECT ******

TOPIC- FULLY FLEXIBLE SLOT SYSTEM FOR SHOPS

PROBLEM STATEMENT: In regard with the current pandemic situation we aim to design a slot system that would enable people to book their slots (each slot limited to a specific number of customers) and generate token numbers to refrain from social gathering .
We have considered a supermarket and assumed that there are different floors each dedicated to one specific item.

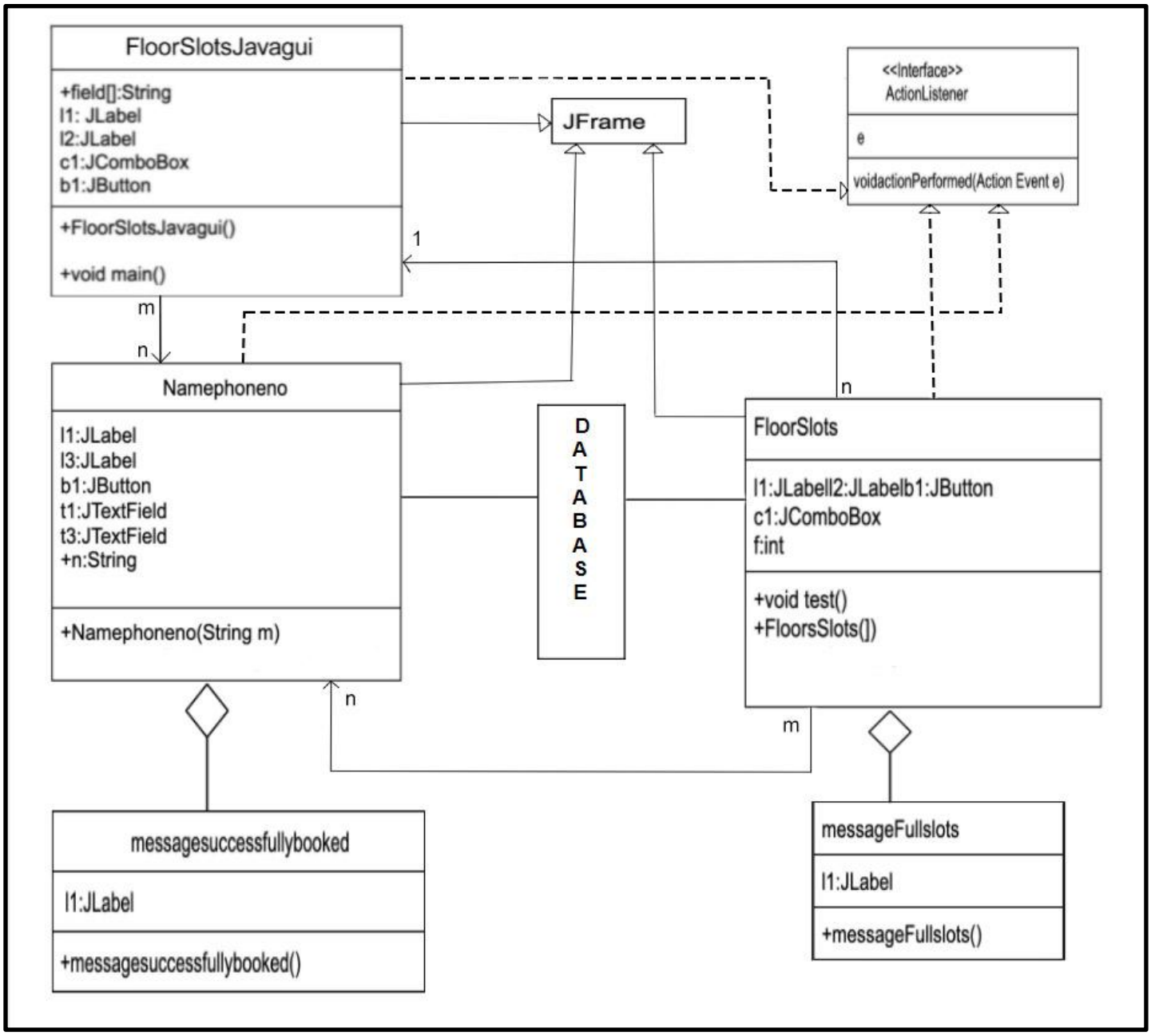
Eg: the first floor for grocery , second for clothes , third for accessories, fourth for daily essentials . We can restrict the number of people to 20(suppose) for each slot. Here we have restricted ourselves to 5 entries.

The contents of the document include **CLASS DIAGRAM** and the **Snapshots** of the STEP by STEP execution.

CLASS DIAGRAM

SNAPS OF THE EXECUTION

The first dialog box that displays the floor details asks the user to choose for the desired floor.


FloorSlotsJavagui
+field[]:String
l1: JLabel
l2:JLabel
c1:JComboBox
b1:JButton
+FloorSlotsJavagui()
+void main()
JFrame
<<Interface>>
ActionListener
e
voidactionPerformed(Action Event e)
1
m
n
Namephoneno
l1:JLabel
l3:JLabel
b1:JButton
t1:JTextField
t3:JTextField
+n:String
+Namephoneno(String m)
DATABASE
n
FloorSlots
l1:JLabell2:JLabelb1:JButton
c1:JComboBox
f:int
+void test()
+FloorsSlots[])
n
m
messagesuccessfullybooked
l1:JLabel
+messagesuccessfullybooked()
messageFullslots
l1:JLabel
+messageFullslots()

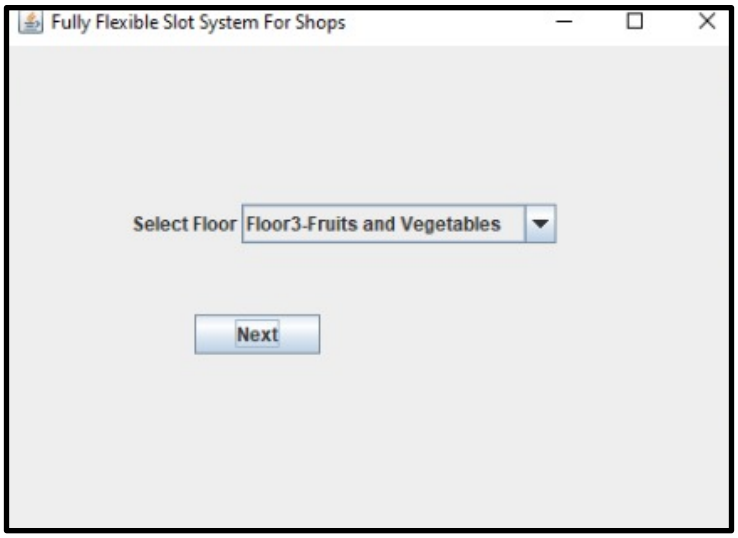


Fig 1: Select the Floor

Next we have the allotted slots on display and the user has to choose the slot as per his convenience.

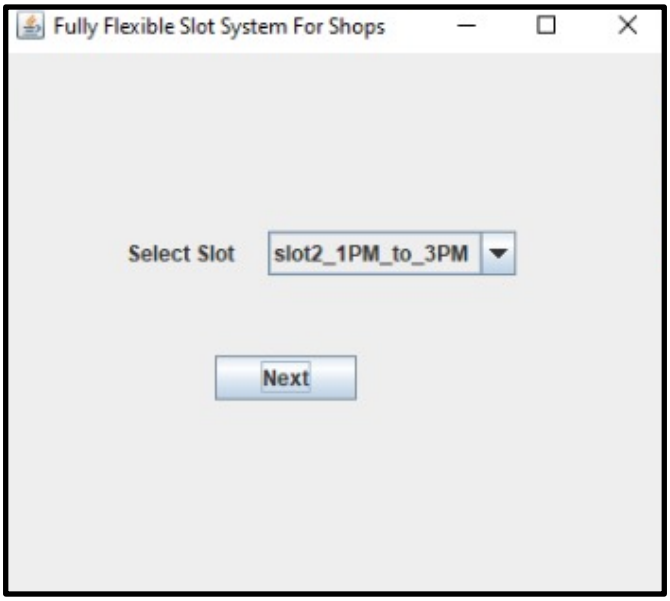


Fig 2: Select the slot time

In the next step we ask the user to input his details : Name and Phone number

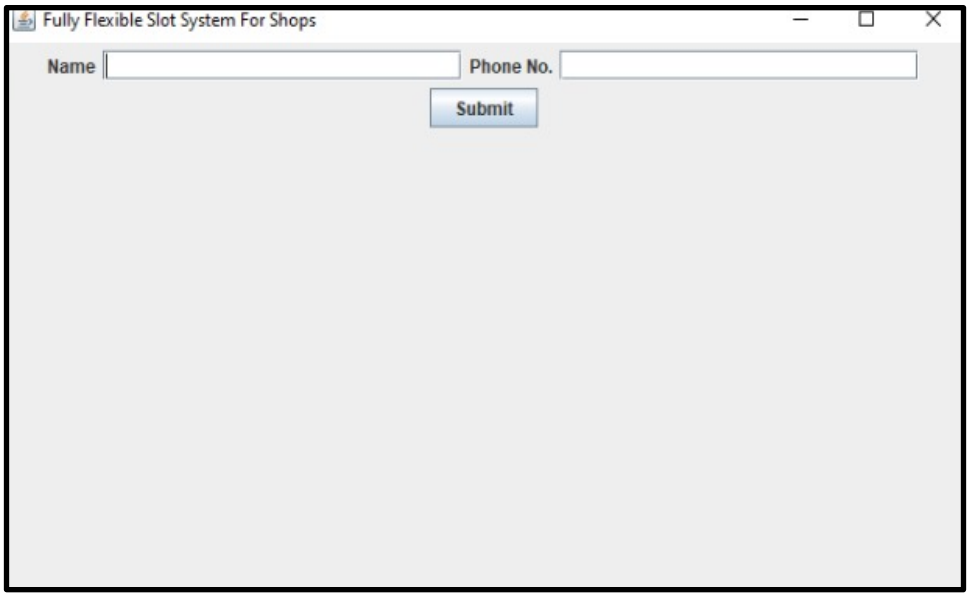


Fig 3: Enter Name and Phone number of the customer

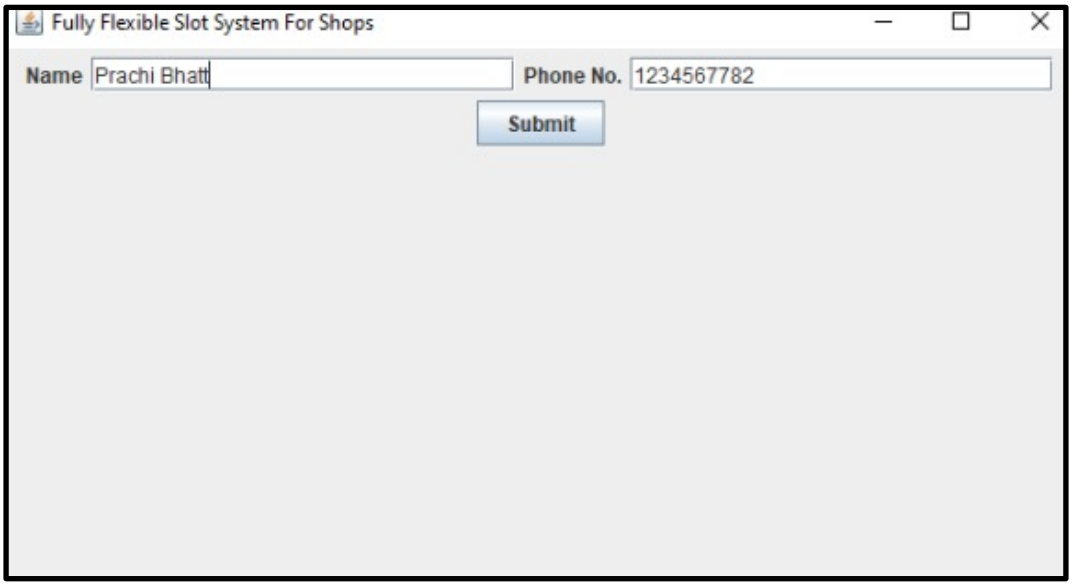


FIg 4: Values entered by the customer

The entries are fed into the database and then checked if the number of customers visiting the particular floor are not exceeding the desired number of customers we want to allow (here it is 5).

In this case the slot is empty and thus we receive a message that slot is successfully booked.

Fig 5: Message after successful registration of slot for customer

Let's consider another case of the same process described above but with slight changes. The same steps are repeated as we see:

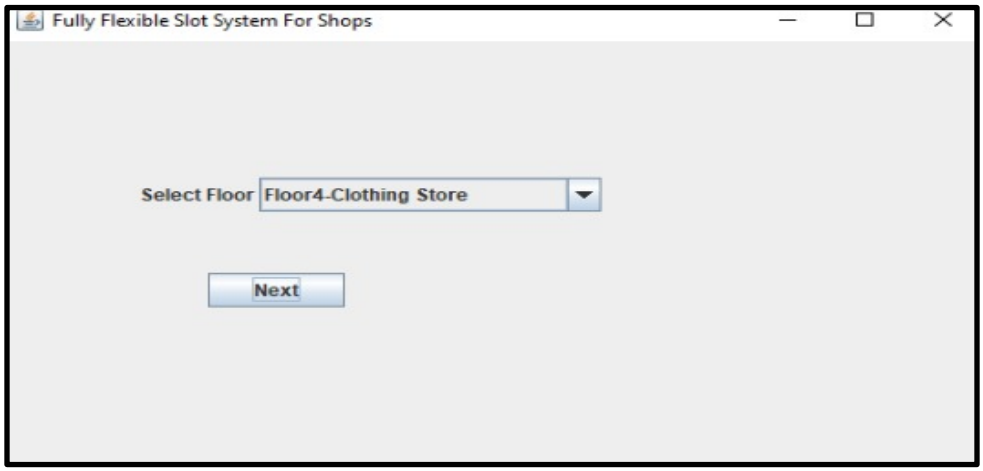


Fig 6: Registration for another visit

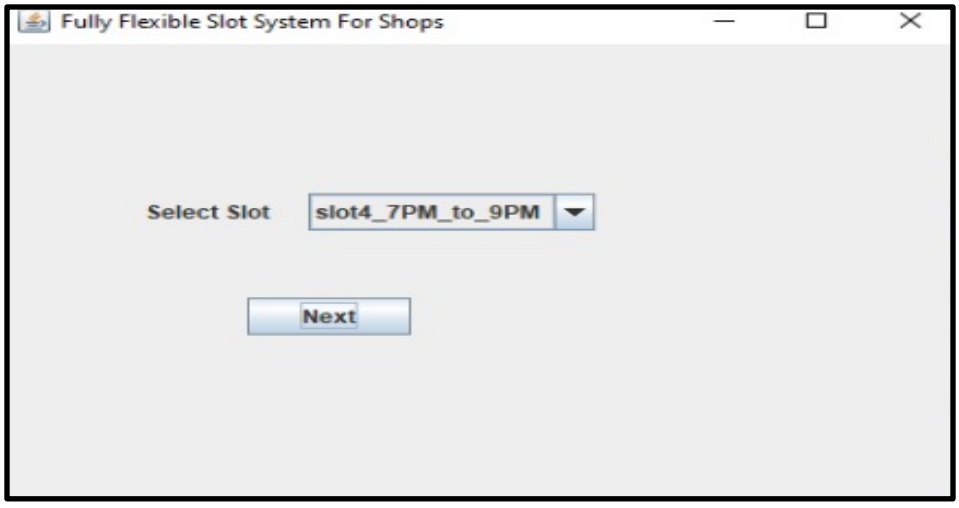


Fig 7: Select Time for visit

The difference is in this step where we have already 5 entries in our database for the ame slt and thus a message is generated to choose another slot.

Fig 8: Slot already full

The database entries are as follows

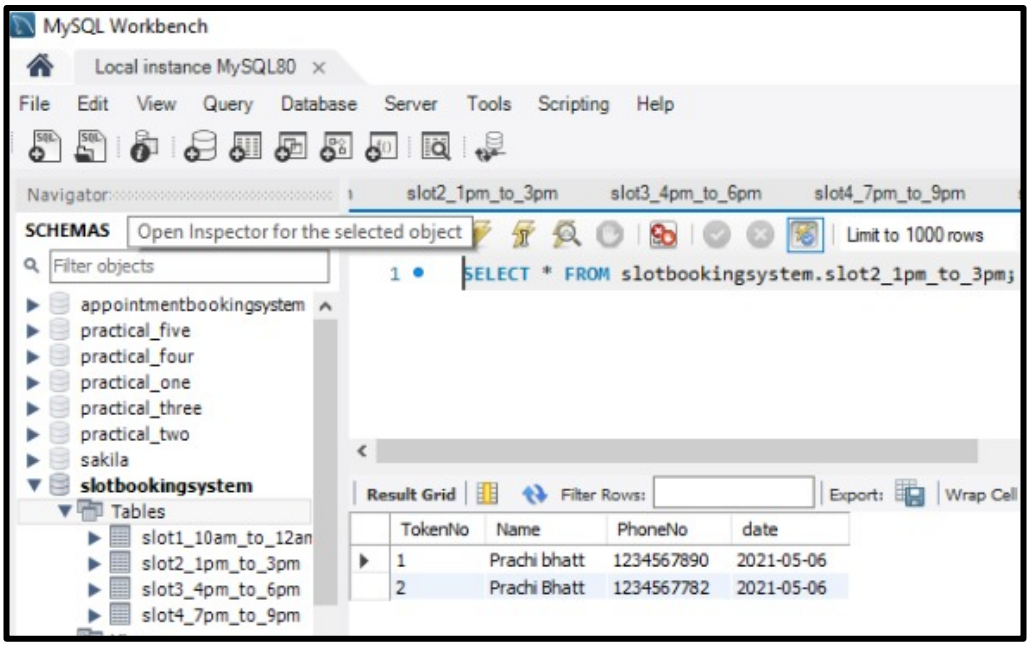


Fig 9: Database visualization 1

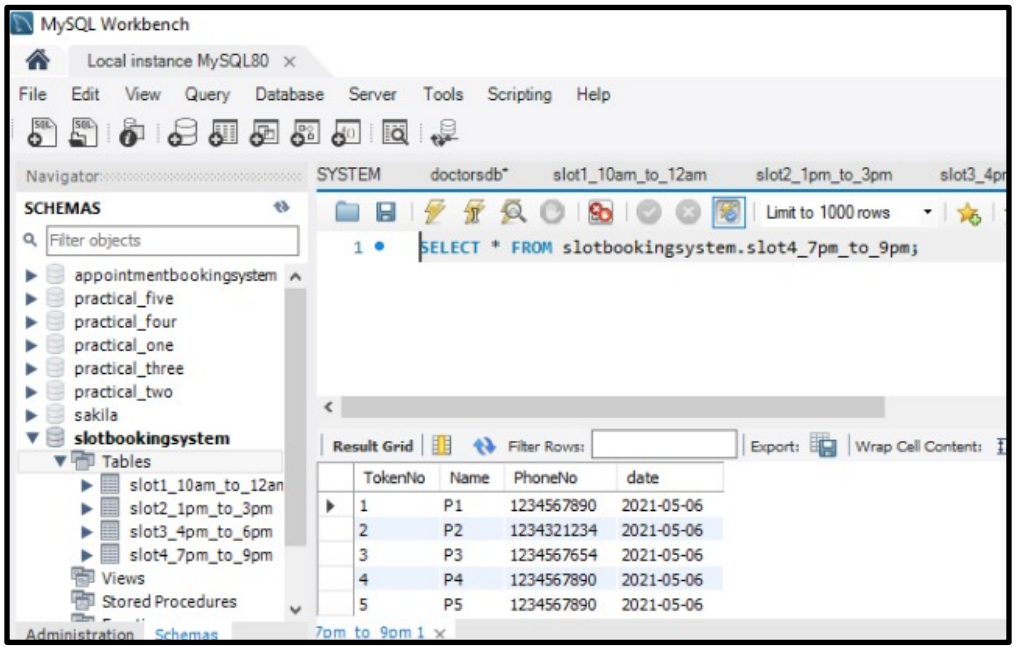


Fig10: Database visualization 2