

**Alexandria University**  
**Faculty of Engineering**  
**Computer and Communication Department**



**Subject**

**Project Report**

<b>Name</b>	<b>Id</b>	<b>Group</b>	<b>Section</b>
<b>Omar Hossam EIDin Ebraheem</b>	<b>6667</b>	<b>3</b>	<b>1</b>
<b>Mohamed Abdelaleem Talaat</b>	<b>6183</b>	<b>3</b>	<b>2</b>

**Supervisor**

**Dr. Mervat Mikhail**

**2019 - 2020**

## 1. Introduction

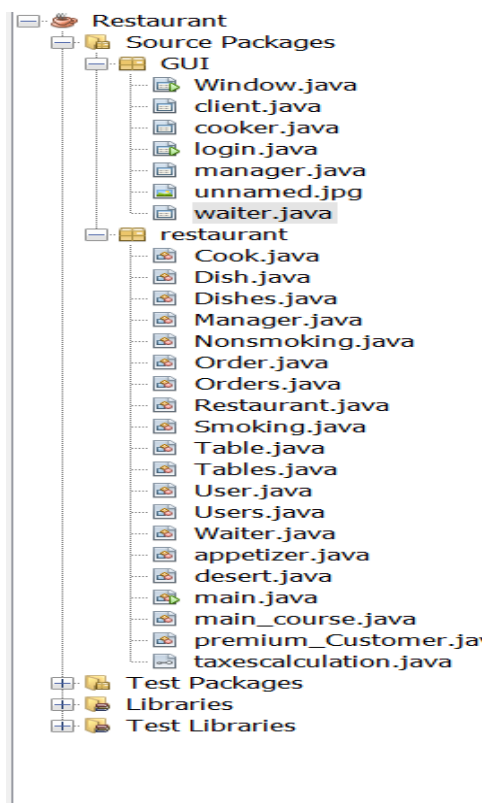
Firstly , it gives us the greatest pleasure to work in collaboration with each others in such an amazing and effortfully project ( Restaurant Reservation System ) .

We have been worked together in each part of the project equally , hoping from god to gain the admire from the Supervisor , teaching assistants and all the instructors .

Also , we have been applied all the Studied OOP concepts , the required rules and the instructions sent in the pdf of assignment 2 that assigned by the teaching assistants at date of 19/3/2020 as much as we can .

## 2. Discussion

We have divided the main idea of the project , during the problem analysis , design and even in the program coding to two main packages which are the : restaurant package and the qui package to be able to apply and implement the project successfully as required as shown in ( *figure 1.1* ) .



**Figure 2.1**

The restaurant package contains more than 90 % of the logic and the OOP applied concepts , while the GUI package contains the remaining 10% percentage in addition to containing all the GUI classes and the codes that helps us in the implementation of the GUI .

In the restaurant package we have applied and implemented the concepts of the inheritance , polymorphism and the interface in addition to the classes and methods which are more than 20 class that helps us in reading from the xml file , and writing on it and implementation of the project using all the studied OOP concepts in addition to some videos and references we will attach them bellow helps us in reading from and writing in the XML file , all the detailed OOP concepts and the mapping of the project is shown briefly in the attached UML class diagram files.

While in the GUI package we have calling and using all the suitable methods and classes that help us in reading and writing in the file , in the GUI package we have implemented more than 5 classes using the JFrame classes in order to maintain and implement the GUI , one for welcome screen , one for login screen , one for waiter , one for manager , one for cooker and the last one is for the client .

We tried as much as we can to reach to the more better design to be more easy and clear in reading ,tracing and organizing , although we have to keep it as much comfortable and identical to the criteria and instructions assigned in the pdf .

Also , to make the code more organizable , we have not wrote nor implement large codes inside the buttons of the GUI , we have organizing the large and more complicated codes in accessible methods , to be called easily inside the buttons to perform their specified task .

Notes :

- We have been implementing the code to read from ( file.xml ) file , while writing and receiving the orders in the ( order.xml ) file .
- We have been printing the data of the file to the running console while reading and writing from the file to check and make sure that the read or the wrote data have been implemented right and clearly .
- We have been calling the reading from file methods after calling the GUI and after opening the login screen and storing it in lists of objects using the list and array lists principles .
- We have been writing the orders taken and ordered by the the client in the (order.xml ) file in order to be cooked and served to the user , and to be displayed by authorized users
- We have been driven the UML class diagram as in extensions of ( JPG , PNG ) photos , as we could not deliver it in extension of ( PDF ) because the diagram was so large and detailed to all the implemented classes , OOP concepts , notes using arrows and even the GUI classes we have included it in the class diagram , so in this case the ( PDF ) extension could not clearly support that detailed UML class diagram , so we forced to deliver it in the ( JPG , PNG ) extensions in order to be delivered and navigated comfortably .
- We have delivered and attached all the needed and necessary files to make the program executed clearly , and free of defects .

### **3. Conclusion**

Finally , despite of the big effort we have done and the more stress we faced during analysing , design , coding the program , debugging , handling the extra cases and the continuous work during the the seven days ago staring to the screen of the computer more than 13 hours a day during the last 7 days , but really we were enjoyed and excited and have a honour to work on a such exiting and amazing project .

Hopping from the god to gain the admire of you as supervisors , teaching assistants and all the instructors of the project .

## References

1. <https://www.youtube.com/watch?v=MLbOb1CVuxs>
2. <https://www.youtube.com/watch?v=ia1ljw7UnU>
3. <https://www.youtube.com/watch?v=4iFHtnYXBtw&t=206s>
4. <https://www.youtube.com/watch?v=Twew1JXOONI&t=25s>
5. <https://www.youtube.com/watch?v=GZ9MT2myBf8&t=5186s>
6. <https://www.youtube.com/watch?v=AVVbuVn0N98&t=12s>
7. <https://www.youtube.com/watch?v=KhWTfIERQo&t=14s>
8. <https://www.googleadservices.com/pagead/aclk?sa=L&ai=DChcSEwjvzImotM3oAhWNse0KHV4KB24YABAAGgJkZw&ohost=www.google.com&cid=CAESP-D2F1QR3IYBdWKPun-GPiVcv1PrssrUTLwTlrbnLSdY65KnpOlbuoyq81gvLAJ5tQS0Rg0aJAIBm6WpIT8vA&sig=AOD641bTjHlcMVuXXMH0aR4eTNu5JVQ&q=&ved=2ahUKEwj6rIKotM3oAhWqAmMBHVbrBOqQ0Qx6BAgVEAE&adurl=>
9. [https://www.w3schools.com/java/java\\_interface.asp](https://www.w3schools.com/java/java_interface.asp)
10. <https://www.javatpoint.com/interface-in-java>
11. <https://www.codeproject.com/Questions/371230/Return-ArrayList-from-method-in-Java>
12. <https://www.vogella.com/tutorials/JAXB/article.html>
13. <https://howtodoinjava.com/jaxb/jaxb-exmample-marshalling-and-unmarshalling-list-or-set-of-objects/>
14. <https://stackoverflow.com/questions/5258159/how-to-make-an-executable-jar-file>
15. <https://www.youtube.com/watch?v=mE3rbtKm-pk>