

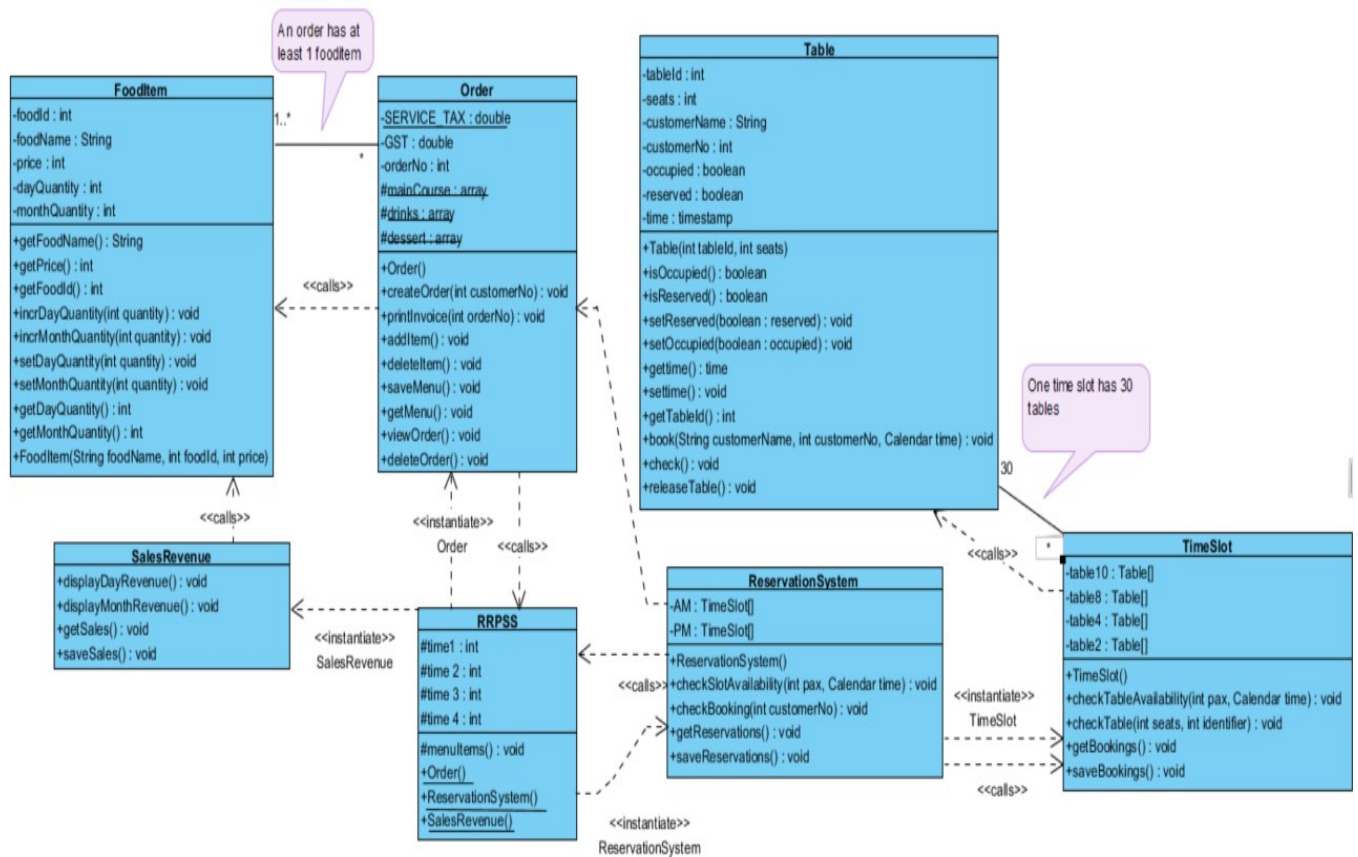


NANYANG
TECHNOLOGICAL
UNIVERSITY

CZ2002 Report

Marathe Ajinkya Avinash	(U1522716K)	(BCG2)
Brandon Ho Min Yang	(U1520888H)	(BCG2)
Rawal Akshay	(U1522163H)	(BCG2)
Zhang Yangshen	(U1420817G)	(SE2)
Voong Alan	(N1603834B)	(SE2)

UML Class Diagram



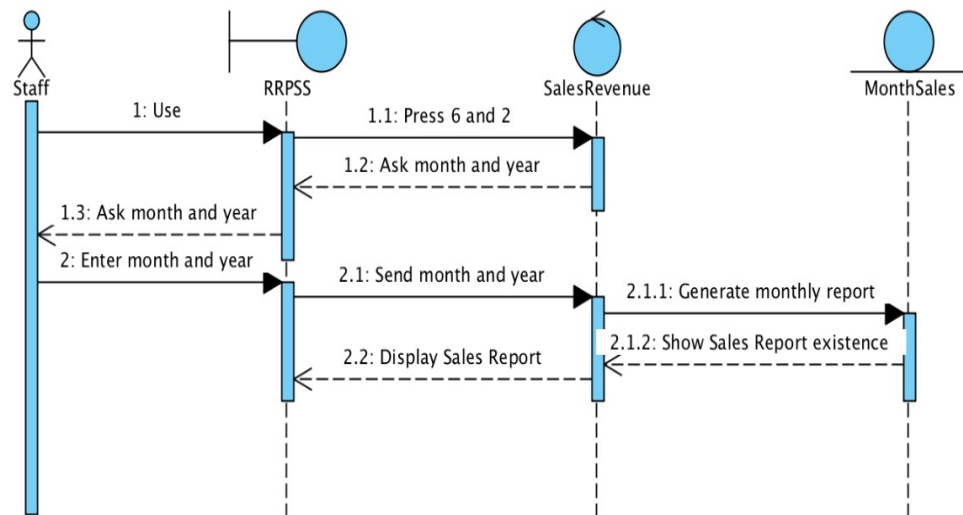
**The full size UML Class Diagram and UML Sequence Diagram have been saved in the CD as separate files for reference.

UML Sequence Diagram

****The full size UML Class diagram and UML Sequence Diagrams have been saved in the CD as separate files for reference.**

	Staff		RRPSS		SalesRevenue		MonthSales	
--	-------	--	-------	--	--------------	--	------------	--

sd Sequence Diagram



OOP Design

- **Single Responsibility Principle-**

According to computer scientist Robert Martin, “A class should have only one reason to change”. In our application, every class encapsulates only any one part of the application functionality. Change in one part of the program, for example, the Reservation system will not affect the food ordering functionality. Our application also supports the single responsibility on a method level. Each method in our program has only one defined function with separate functions for tasks like viewing the order and printing the invoice. For example, change in the format of the invoice would change only that function.

- **Open-Closed Principle-**

Our application supports this principle as this can be extended without changing the source code. Insertion and deletion of menu items is a good example of how our application supports the open closed principle

- **Don't Repeat Yourself-**

Our application does not repeat any functionality. We only have one function which has been reused whenever that functionality needed.

- **Composition over Inheritance:**

Our application works on composition of objects of classes into another classes rather than inheriting classes. This is the reservation system of our application supporting object composition.

Enhancements:

1.Graphical User Interface:

A graphical user interface would make the application user friendly. The current application makes the user enter numbers of their choices to operate the system. The application would be much more interactive with GUI. We must consider making a web application or a mobile application using the java swing GUI.

2.Databases:

The data in the program is lost every time we exit the program. So the application makes use of the text files to store the data while retrieving while starting each program.

Using text files for such applications is very unprofessional and unstructured. When we deal with large data, text files can be very troublesome to store such data. We can make use of databases where we create tables and store data in Access or phpMyAdmin. Using MySQL, we can do incredible dealings with data stored in local computer or internet server. However, using these technology can bring more security to the date.

Reusability in the enhanced application

- Though we make use of a Graphical User Interface, the code for all the systems like reservations, order placing and revenue report can always be reused at the back end and thus it supports the OOP principle of reusability.
- Similarly, the same functions and classes can be reused with just the addition of code for storing and retrieving data in databases.