Laboratory 3

Data flow modelling with CASE tools – High Level Design

1. **Introduction and Purpose of Experiment**

   Students will apply data flow modelling to develop the high level design for given scenario

2. **Aim and Objectives**

   Aim: To develop software architecture for a given requirements specification using Structured analysis and Design Technique

   Objectives: At the end of this lab, the student will be able to

   - Identify context of the software
   - Identify Inputs, Outputs and Data Stores for a given software
   - Identify modules in a software and their dependencies
   - Create design document for a given SRS

3. **Experimental Procedure**

   - Work in teams of 4 students
   - Each team should read the problem statement and identify requirements as a group
   - Each team will then confirm the requirements and document the requirements in an high level design document
   - Each individual will then write their lab manual, documenting their observations

4. **Presentation of Results**

USE CSE TABLE:

5. Use Case	Choose Language
Actors	Customer
Description	A Customer has to order food from the tablet been provided. Customers come from a variety of language backgrounds. To address people speaking different languages, a 'select

	language' option is provided for the customer to choose, upon which the tablet functions strictly under the chosen language.

Use Case	Order Food
Actors	Customer, Chef
Description	The customer has to order food from the menu in the tablet. The customer confirms the order upon which the order is been updated to the chef and the cashier.

Use Case	Notify Serve Team
Actors	Chef, Serving Team
Description	Once the orders received are prepared to be served, the chef notifies the Serving team to collect the food.

| **Use Case**	Serve Food
Actors	Customer, Serving Team
Description	On getting notified to collect food from the Chef, the serving team collects the food and serves the Customer.

Use Case	Feedback
Actors	Customer, Chef, Manager
Description	The feedbacks are given about the food and other services on the tablet by the customer. The feedbacks related to food and other services are reported to Chef and the Manager respectively.

Use Case	Bill Payment
Actors	Customer, Cashier
Description	The Customer has to pay on having food either through cash or cashless means. Either of these choices is notified to the Cashier by the Customer through the tablet. Paying through cash involves interaction between Cashier-Customer. Cashless mode requires the Customer to pay through the tablet itself.

Use Case	Connect to LAN

Actors	Cashier, Manager
Description	A report on customers' orders, their feedbacks and other data including taxes, profit are maintained in a Local Area Network which can be accessed only by the Cashier and the Manager. Cashier updates the customers' orders and other data and Manager can get these reports just by logging in.

Use Case	Validate User
Actors	Cashier, Manager
Description	For the Cashier to update reports and the Manager to get these reports, they have to login to the LAN, upon which they are given access.

USE CASE DIAGRAM:
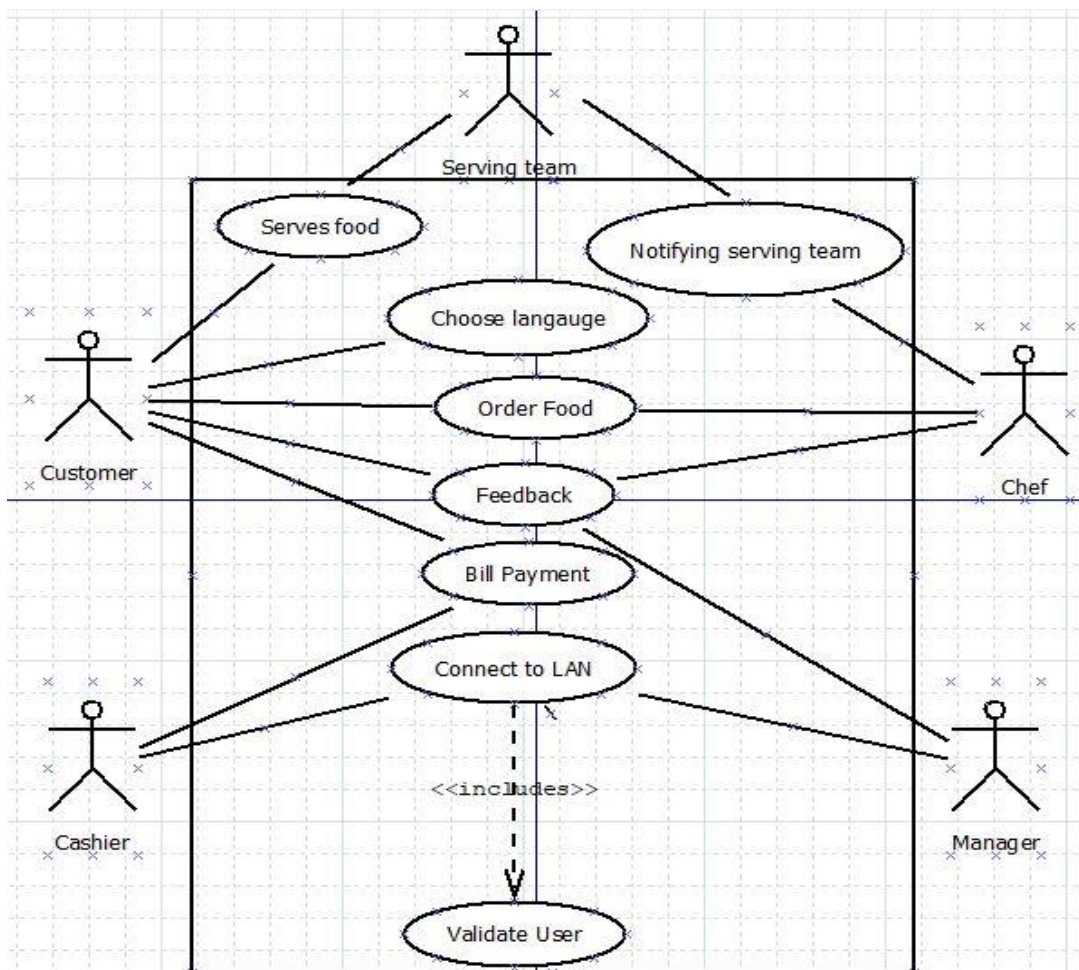
6. **Analysis and Discussions**

One of the hardest parts of any project is agreeing on precisely **what one is going to build.** A use case diagram serves for this purpose. It helps the development team for a software application **look at how the system will be used**. As the name suggests, a use case is literally a case for a potential usage of the system.

Eliciting all the use cases for a software application gives the development team a good handle on what **goals the system** is supposed to achieve from its **user's perspective**. E.g. the online ticket booking system shall enable travellers to book tickets for a selected airline on a selected flight, date and time.

Each use case describes one way the system is used, but one of the big benefits of use case diagrams is that it also describes all of the **things that might go wrong**. Identifying exceptions to a successful scenario early in the project saves a lot of time by finding subtle requirements.

7. **Conclusions**

Use case diagrams help in viewing the system as a black box which is helpful in explaining the system to developers as well as a design plan to customers. This diagram helps also in debugging some errors that might occur in planning as well.

8. **Comments**

1. **Limitations of Experiments**

Use cases are high level view of the system as whole. This view just gives us the idea of how the system woks as end user's point of view. To understand the inner working of the system, one must go much deeper to Sequence Diagrams to understand the sequence of events that occur.

2. **Limitations of Results**

The ordering of food system is connected to the local network a.k.a LAN. The orders are then updated to Cashier, Chef and Manager based upon the Hotel Management Algorithm. This operation is not shown in the current Use Case Design.

3. **Learning happened**

In the current Lab, one could learn:

i. The importance of SRS and other document in the software engineering process

ii. Importance of viewing the system in an abstract view and analysing its operation and doing the same in the testing phase i.e black box testing

4. **Recommendations**

Instead of DIA, "Gliffy" an online tool can be used in the lab. It is far superior than the software which is currently in use and also very flexible.

Component	Max Marks	Marks Obtained
Viva	6	
Results	7	
Documentation	7	
Total	20	