

ASSIGNMENT-1

ONLINE FOOD DELIVERY SYSTEM

PROJECT REPORT

Submitted by

SRUTHI K S

(Admission No : 12580)

Submitted to

ANIT JAMES

Asst Professor in Computer Applications

DEPARTMENT OF COMPUTER APPLICATION

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY,KOTTAYAM

APJ Abdul Kalam Technological University

University in Thiruvananthapuram

Kerala

2021

TABLE OF CONTENTS

Sl.No. Topic

1. INTRODUCTION

2. SYSTEM ANALYSIS

2.1 Existing System

2.2 Proposed system

2.3 Feasibility Analysis

2.3.1 Technical Feasibility

2.3.2 Economic Feasibility

2.3.3 Operational Feasibility

2.4 System Specification

2.4.1 Software Specification

2.4.2 Hardware Specification

2.5 Software Requirements

2.5.1 Overview of PHP

2.5.2 Apache Web Server

2.5.3 MYSQL

2.5.4 HTML(Hyper Text Markup Language)

2.6 Data Flow Diagram(DFD)

2.7 Structure Chart

2.8 Menu Tree

2.9 System Flow Chart

3. SYSTEM DESIGN

3.1 Module Description

3.2 Input design

3.3 Output design

3.4 Database design

3.4.1 Normalization

3.4.1.1 First Normal Form(1NF)

3.4.1.2 Second Normal Form(2NF)

3.4.1.3 Third Normal Form(3NF)

4. **SYSTEM TESTING**
 - 4.1 Unit Testing
 - 4.2 Integration Testing
 - 4.3 System Testing
5. **SYSTEM IMPLEMENTATION**
6. **SYSTEM MAINTENANCE**
 - 6.1 Types Of System Maintenance
 - 6.1.1 Corrective Maintenance
 - 6.1.2 Adaptive Maintenance
 - 6.1.3 Perfective Maintenance
7. **CONCLUSION**
8. **FUTURE ENHANCEMENT**
 - APPENDIX A : SCREEN LAYOUTS**
 - APPENDIX B : TEST CASES**
9. **BIBLIOGRAPHY**

CHAPTER-1

INTRODUCTION

In real life situations, we may face some sort of difficulties while making food orders in a hotel. Waiting for the waiter to arrive is itself a time consuming process. Also the menu card may lack details like the chef who prepared it and its recipes. Updating the printed rates and details in the menu card is also an expensive work. Therefore most of the managements do not care to update it. This often results in conflicts between the customers and management. This project overcomes all these drawbacks.

Online Food Delivery system is a web application program through this the user can make food orders. By choosing the menu, the user gets a detailed picture of the various food items prepared in the hotel, the quantity left in the hotel and rate of each food items. The hotel receives the order given by the user and gives an acknowledgement that he has received. Then the users get the corresponding bill for the ordered food. The user can select the mode of payment, cash or card .The admin will view the food orders. Then the admin will allot the work to the concerned delivery boy. The delivery boy will get the details of allotted work. Then the delivery boy can add current status to the admin. The admin will receive the status of the deliveryboy. A customer can get discount and other benefits by registering as a Prime member. They get discounted on delivery. The user can make door-to-door deliveries from most restaurants that offer take-out. After you place an order, a Dasher picks it up and delivers it to your door. Doordash then displays a list of restaurants in the area that they can pick up from and deliver to your address.

The project is developed using PHP as designing tool and MySQL as database. PHP is a powerful tool for web programming From Microsoft and is the front end of this project with MySQL as backend.

CHAPTER-2

SYSTEM ANALYSIS

In order to develop a new system, a detailed study of the existing system called the System Analysis is necessary. Analysis is the first step in the system development life cycle.

Identifying the need for a new information system and launching an investigation for the required system which best satisfy the exact requirement of the user is the first part of the development activities of a system.

THE INITIAL INVESTIGATION

The first step in the system development life cycle is the identification of the need. This is a user's request to change, improve, or enhance an existing system. It is also important to determine whether the request is valid and feasible before we attempt to improve or modify the existing system or build a new one.

The user's request identifies the need for change and authorizes the initial investment. It may undergo several modifications before it becomes a written commitment.

Once the request is approved the following activities are carried out:

- ☐ Background Investigation
- ☐ Fact-finding and Analysis

At the end a detailed report is created which is called the "Project Proposal" or "Software Requirement Specification".

SOFTWARE REQUIREMENT SPECIFICATION (SRS)

Software Requirement Specification is an important document prepared at the end of the analysis activities. It is appropriate to think that the SRS document as a contract between the development team and the customer, which can also be used to resolve any disagreements, which may arise in the future. Once the customer agrees to the SRS document, the development team proceeds to develop the product conforming to all the requirements mentioned in the SRS document.

A software requirement specification contains the following details.

- ❑ Functional requirements of the system.
- ❑ Non-functional requirements of the system.
- ❑ Constraints on the system.

2.1 EXISTING SYSTEM

These days a customer finds it very difficult to place food orders in a hotel. The existing system is time consuming on the whole. The customer has to find the table ,wait for the waiter. The menu card also lacks details like recipes of the food and the chef who prepared it. Updating the printed menu card is also an expensive work.

2.1.1 Limitations of the Existing System

1. Time consuming

Since all the works are done manually,all the process take a lot of time are stored in a file . so it takes a lot of space to store small information.

2. Inefficient

Searching manually for a particular record requires that files to be arranged in a proper way. But this happens seldom in the existing system.

3. Inflexibility

Editing the details of a particular entry is a tedious and time consuming job.

2.2 PROPOSED SYSTEM

The proposed software is a web-based application. The proposed system helps in many ways. The application is very helpful to a person to place food orders. The user gets a detailed view of the type, quantity and quality of the food items. The customer can also retrieve information about the chef who prepared the food item and its recipes.

Advantages

- It records all the details permanently.
- Less use of manual work
- Fast & easy application development
- User friendly etc.

- Any cancellations of parties are also updated.
- Also keep the record of food items prepared and the sales of food.
- All the food order details like order types are stored daily.

2.3 FEASIBILITY ANALYSIS

- Feasibility study done in three phases
 1. Operational feasibility
 2. Technical feasibility
 3. Economical feasibility

2.3.1 Operational Feasibility

Proposed systems are beneficial only if they can be turned into information systems. That is it will meet the organizations operating requirements and also checks that whether the system will work when it is developed and installed. The software **Online food Delivery** will support the operational feasibility to a great extends. The performance of this software is more accurate, more user friendly, effective, error free.

A feasibility study is not warranted systems, in which economic justification is obvious, technical risk is low, few legal problems are expected and no reasonable alternative exists. However, if any of the preceding conditions fail, a study of that area should be conducted. Economic justification includes a broad range of concerns that include cost-benefit analysis, long-term corporate income strategies, impact on other profit concerns or products, cost of resources needed for development and potential market growth.

2.3.2 Technical Feasibility

This is related to the technicality of the project. This evaluation determines whether the technology needed for the proposed system is available or not. It deals with hardware as well as software requirements. That is, type of hardware, software and the methods required for running the systems are analysed. This involves financial consideration to accommodate technical enhancement. If the budget is a serious constraint, then the project is judged not feasible.

The website online Grocery supports the technical feasibility to a great extends. That is, this software can be operated with the minimum technical support. It uses PHP as front end,

MYSQL as database at windows platform and Mozilla Firefox and Google Chrome as browser. And also it provides accuracy, reliability, ease of access and data security.

2.3.3 Operational Feasibility

Proposed projects are beneficial only if they can be turned into information systems that will meet the operating requirements of the organization. This test of feasibility asks if the system will work when it is developed and installed. This project satisfies all the operational conditions.

2.3.4 Economic Feasibility

Economical analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as cost/benefit analysis: the procedure is to determine the benefits and saving that are expected from a proposed system and compare them with cost. If benefits outweigh cost, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have a chance of being approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle.

Hence the engineer will not find any difficulty at the installation time and after installation user also newer find difficulty i.e., hang, slow speed or slow response time. One project is compulsory for each student this project is either dummy or lives. If I am developing a live project then it gives a lot of confidence. It is better for me and for company because, I am developing a system without any money. So, everything is in favour now, I can say the cost of this software is I think negligible. Hence the economic feasibility is very good.

2.4 HARDWARE SPECIFICATION

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application.

Minimum requirements are,

Processor: 2.6 GHz dual core

Intel processor RAM: 2GB

Monitor: LED Monitor

Hard Disk Drive: 240 GB

Keyboard: 110 Logitech multimedia keyboards

Mouse: PS/2 Logitech Scroll Mouse

Cell Phone: Any Android phones

2.5 SOFTWARE REQUIREMENTS

Front End: Codegniter and PHP

Back End: My SQL

Operating System: Windows 10, 8, 7

2.5.1 Overview of PHP

The original release of PHP was designed and created by Rasmus Lerdorf back in the middle of the 90s as a way of making various common web tasks easier and less repetitive. Back then, the main goal was to have the minimum amount of logic as was possible in order to achieve results, and this led to PHP being HTML-centric - that is, PHP code was embedded inside HTML.

The first popular version of PHP was called PHP/FI 2.0, for Personal Home Page / Form Interpreter, and, despite its parsing inconsistencies, managed to attract a fair few converts, including myself. The main issue with this version was that the PHP/FI parser was largely hand-written, and so users often encountered scripting errors that were not technically errors - they were just the PHP/FI parser screwing up. Furthermore, the parser was absolutely tied to the Apache web server, and was hardly renowned for its speed.

Some of these issues were resolved in version 3, when Zeev Suraski and Andi Gutmans re-wrote PHP from the ground up using standard "compiler compiler" tools like Flex and Bison. This made the parser itself all but bulletproof, which in turn gave sanity back to many PHP users!

2.5.2 HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as `` and `<input />` directly introduce content into the page. Other tags such as `<p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

2.5.3 XAMPP

XAMPP is an easy to install Apache distribution containing MySQL, PHP and Perl. XAMPP is really very easy to install and to use - just download, extract and start.

XAMPP for Windows

The distribution for Windows 2000, 2003, XP, Vista, 7 and 8. This version contains: Apache, MySQL, PHP + PEAR, Perl, mod_php, mod_perl, mod_ssl, OpenSSL phpMyAdmin,

Webalizer, Mercury Mail Transport System for Win32 and NetWare Systems v3.32, Ming, FileZilla FTP Server, mcrypt, eAccelerator, SQLite, and WEB-DAV + mod_auth_mysql.

- Apache 2.4.4
- MySQL 5.5.32
- PHP 5.4.16
- phpMyAdmin 4.0.4
- FileZilla FTP Server 0.9.41
- Tomcat 7.0.41 (with mod_proxy_ajp as connector)
- Strawberry Perl 5.16.3.1 Portable

2.5.4 MYSQL

MySQL is the world's most popular open source database software, with over 100 million copies of its software downloaded or distributed throughout its history. With its superior speed, reliability, and ease of use, MySQL has become the preferred choice for Web, Web 2.0, SaaS, ISV, Telecom companies and forward-thinking corporate IT Managers because it eliminates the major problems associated with downtime, maintenance and administration for modern, online applications.

The MySQL database is owned, developed and supported by Sun Microsystems, one of the world's largest contributors to open source software. MySQL was originally founded and developed in Sweden by two Swedes and a Finn: David Axmark, Allan Larsson and Michael "Monty" Widenius, who had worked together since the 1980's. More historical information on MySQL is available on Wikipedia

- The best and the most-used database in the world for online applications
- Available and affordable for all
- Easy to use
- Continuously improved while remaining fast, secure and reliable
- Fun to use and improve
- Free from bugs

2.5.5 Apache Web Server

The three leading Web servers: Apache HTTP Server for UNIX and Windows, Microsoft Internet Information Server, and Netscape Enterprise Server. Apache Server is generally considered the most stable of major Web servers, With a reputation for enviable uptime percentage. Although it is neither the fastest nor the easiest to administer, once you get it set up, Apache HTTP Server seemingly never crashes. It also doesn't require server reboots every time a setting is changed. PHP inherits this reliability; plus, its own implementation is solid yet light weighted

A two-and-a-half-month head to head test conducted by the Network Computing labs in October 1999, Apache Server with PHP handily beat both IIS/Visual Studio and Netscape Enterprise Server for stability for environment. Apache is probably the Web server most commonly used with PHP and MYSQL-so common that the acronym LAMP has emerged to describe precisely this combo.

2.6 DATA FLOW DIAGRAM (DFD)

A graphical representation is used to describe and analyze the movement of data through a system manual or automated including the processes, storing of data and delays in the system. Data flow Diagrams are the central tool and the basis from which other components are developed.

The transformation of data from input and output through process may be described logically and independently of the physical components associated with the system. They are termed logical data flow diagrams, showing the actual implementation of data between people, departments and work stations. DFD is one of the most important modeling tools used in the system design. DFD shows the flow of data through different process of the system. Data Flow Diagrams are made up of number of symbols, which represent system components.

Context Diagram-Level 0

Analysis model helps as to understand the relationship between different components in the system. Analysis model shows the user clearly, how a system will function. This is the first technical representation of a system. The analysis modeling must achieve three primary objectives.

- To establish a basis for creation of software design.
- To describe what the user requires.
- To define set of requirements that can be validated once the software is built.

A data flow diagram is a graphical technique that depicts information flow and transforms that are applied as data move from input to output. The DFD is used to represent increasing information flow and functional details. A level 0 DFD also called fundamental system model represents the entire software elements as a single bible with input and output indicated by incoming and outgoing arrows respectively.

DFD illustrates how data is processed by a system in terms of inputs and outputs. It is a picture of system processing and flow without excessive concern for details.

The DFD showing the top level of the system is called “Context Diagram”. It should be an overview including basic inputs, processes and outputs. Then it is exploded in to more detailed lower level diagram that shows additional features of the system.

The purpose of DFD is to provide a semantic bridge between users and system developers. The diagrams are graphical, eliminating thousands of words, logical

representations, modeling what system does; hierarchical, showing system at any level of details; and Jargon less, allowing user interaction and reviewing.

The goal of data flow diagramming is to have a commonly understood model of a system. The diagram is the basis of structured system analysis. The Data flow diagram, also known as “Bubble Chart” has the purpose of clarifying system requirements and identifying major transformations that will become program in system design. The bubble represents the data transformations and the lines represent data flows in the system.

2.6.1. Data Flow Notations

External Entity

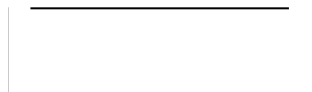


A square defines a source or destination of system data.

Data flow

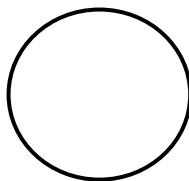


An arrow identifies data flow. It is pipeline through which information flow.



Data store

An open rectangle is data store, data at rest, or a temporary repository of data.

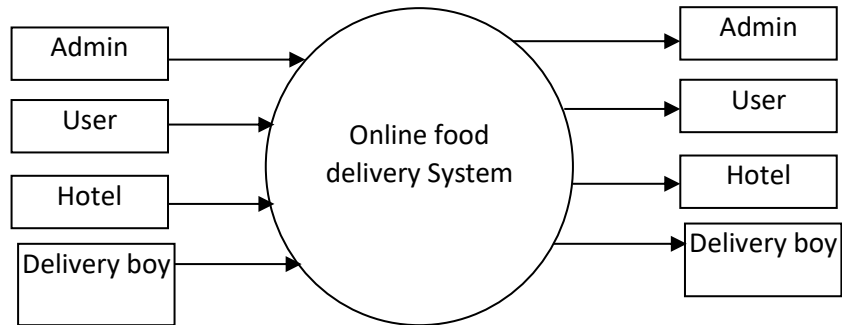


Process

A circle or a bubble represents a process that transforms incoming data flows into outgoing data flows.

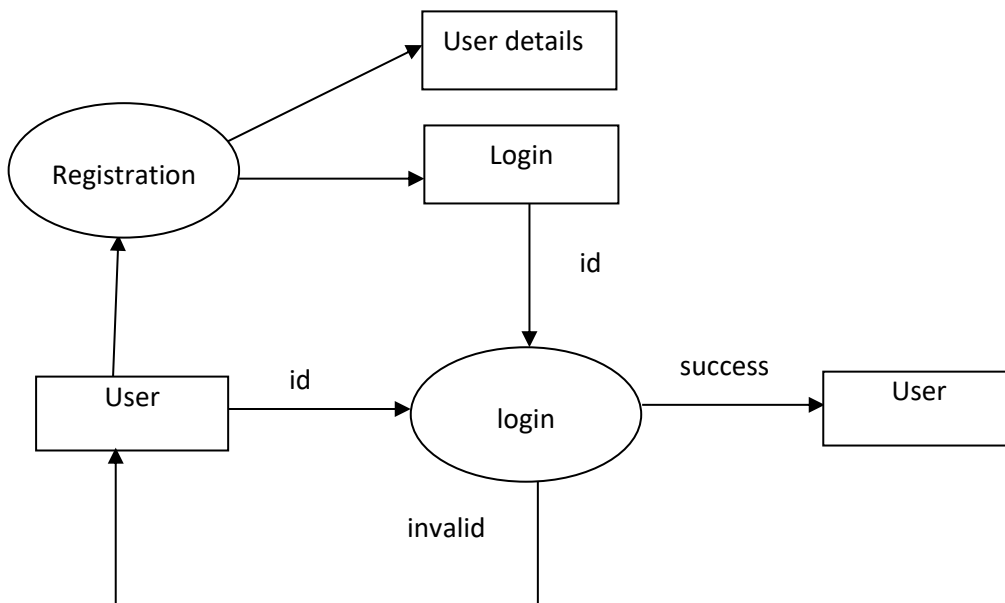
2.6.2 Context Diagram

Level 0 DFD

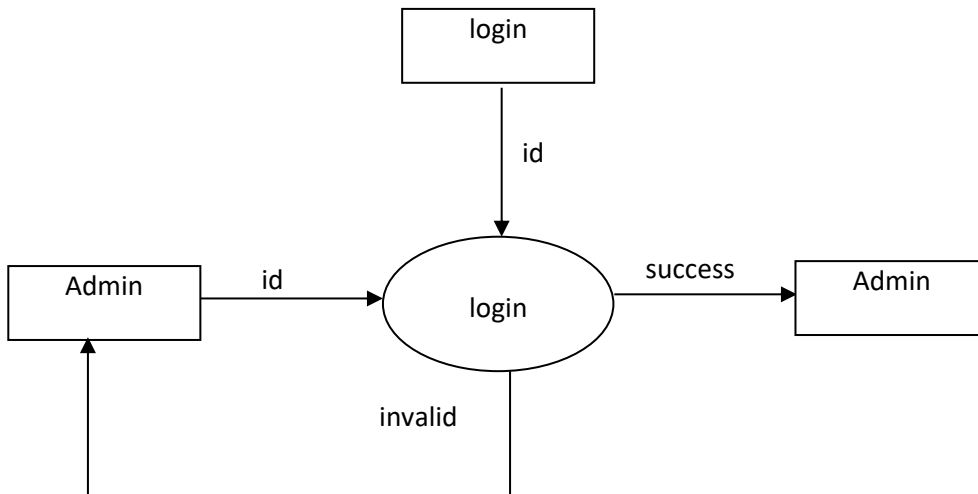


Level 1

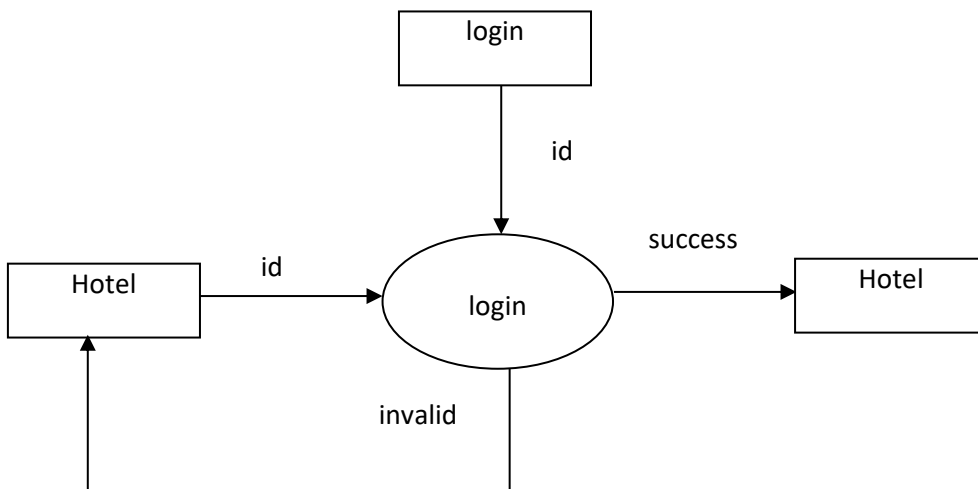
Login Level for User



Level 1
Login Level for Admin

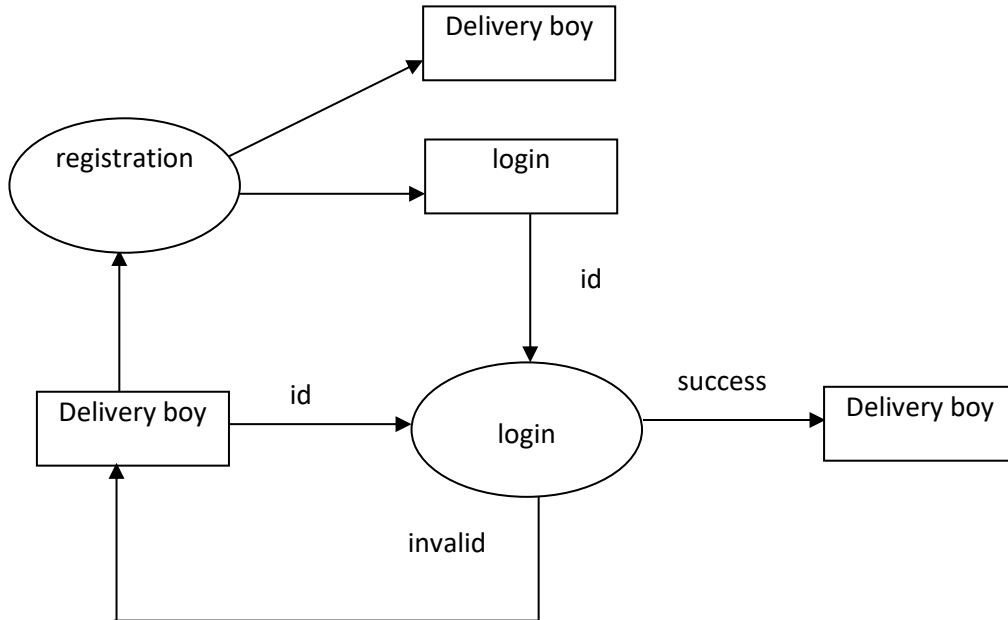


Level 1
Login Level for Hotel



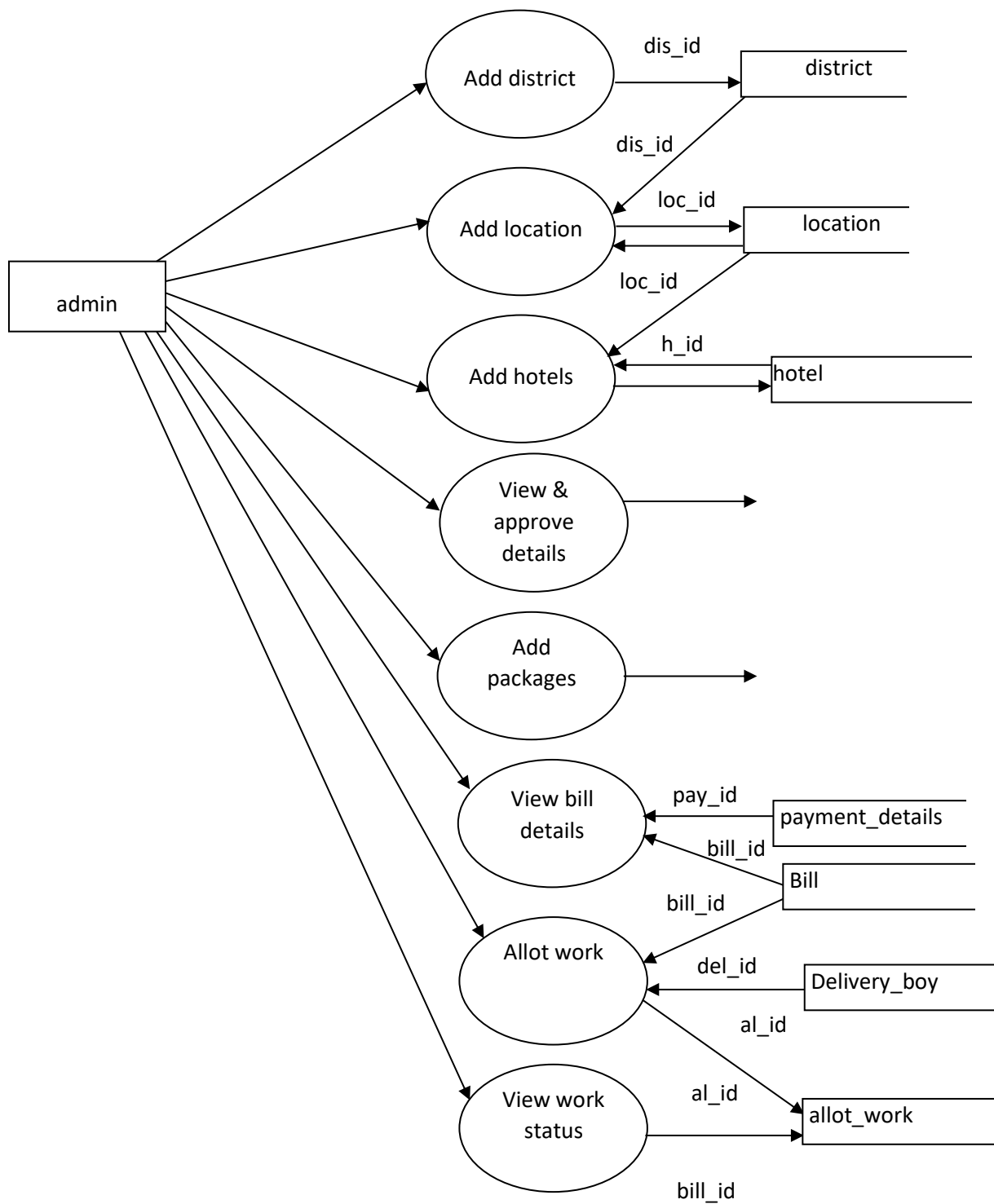
Level 1

Login Level for Delivery boy



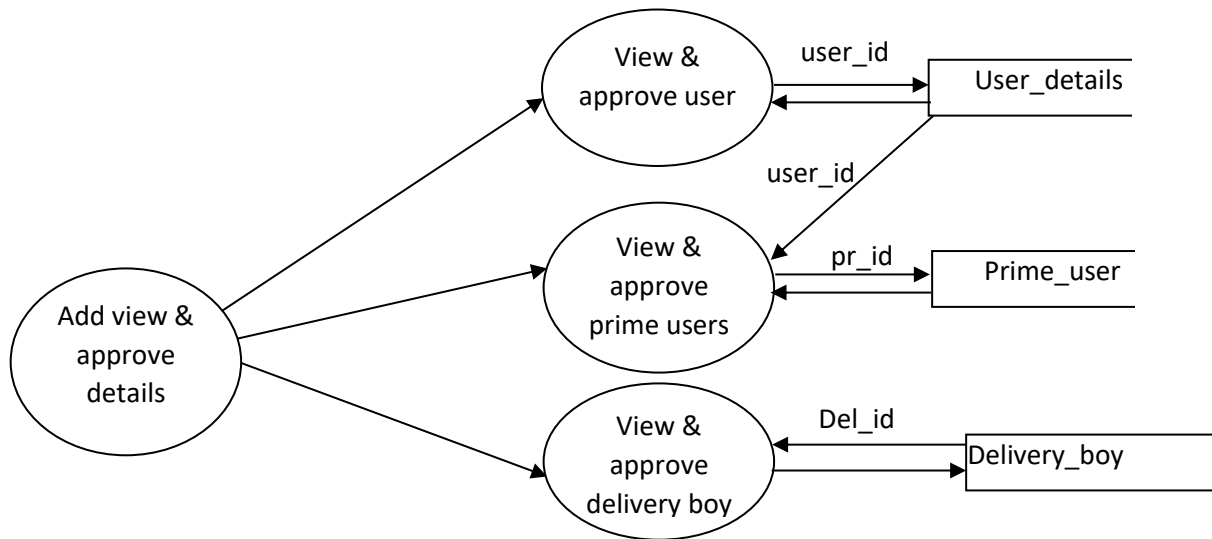
Level 2

Admin



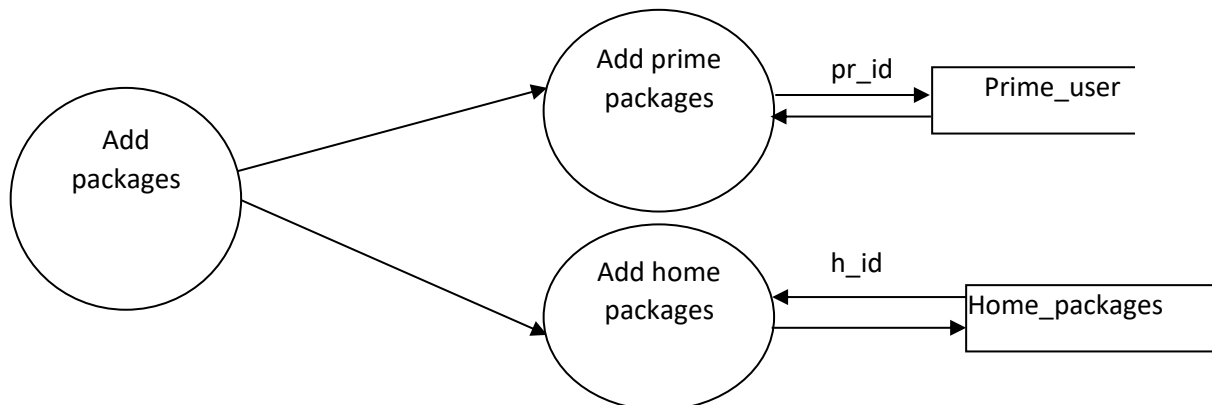
Level 2.0.1

Admin



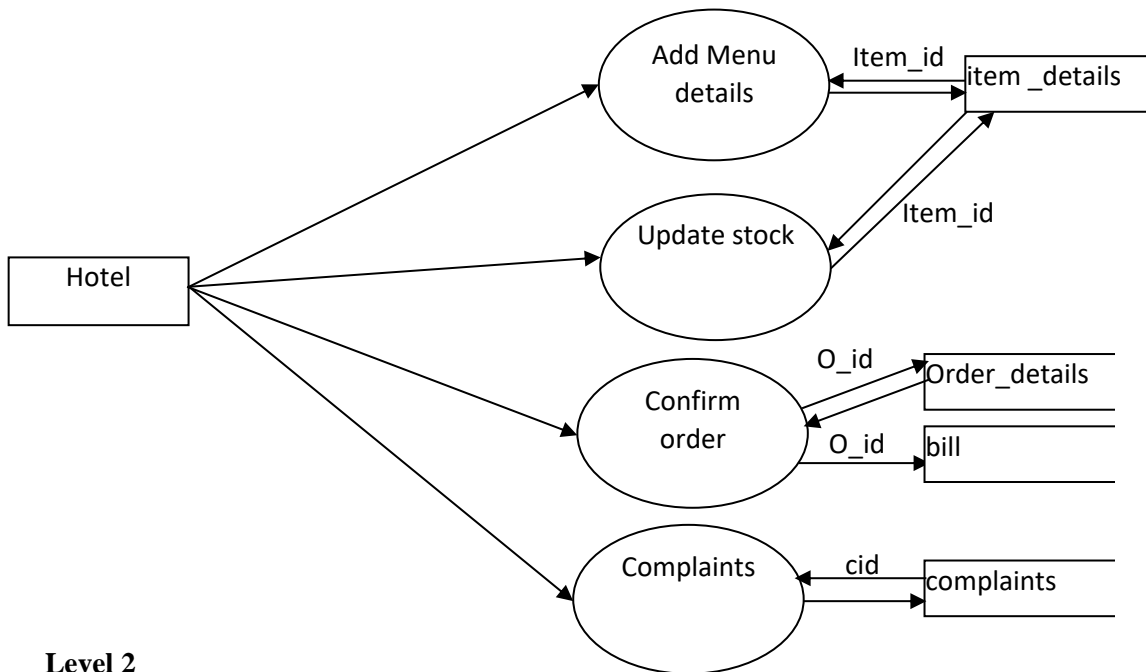
Level 2.0.2

Admin



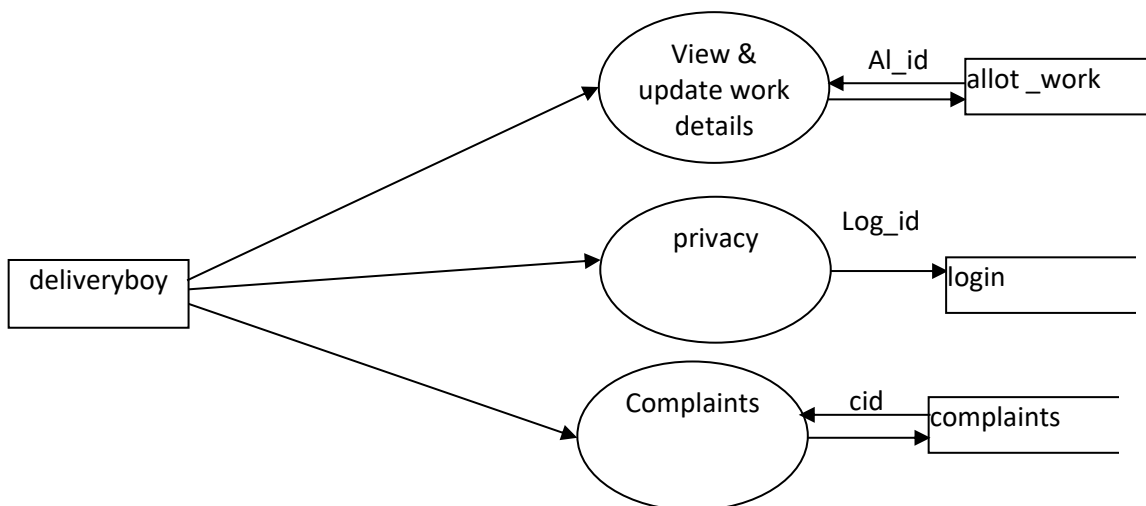
Level 2

HOTEL



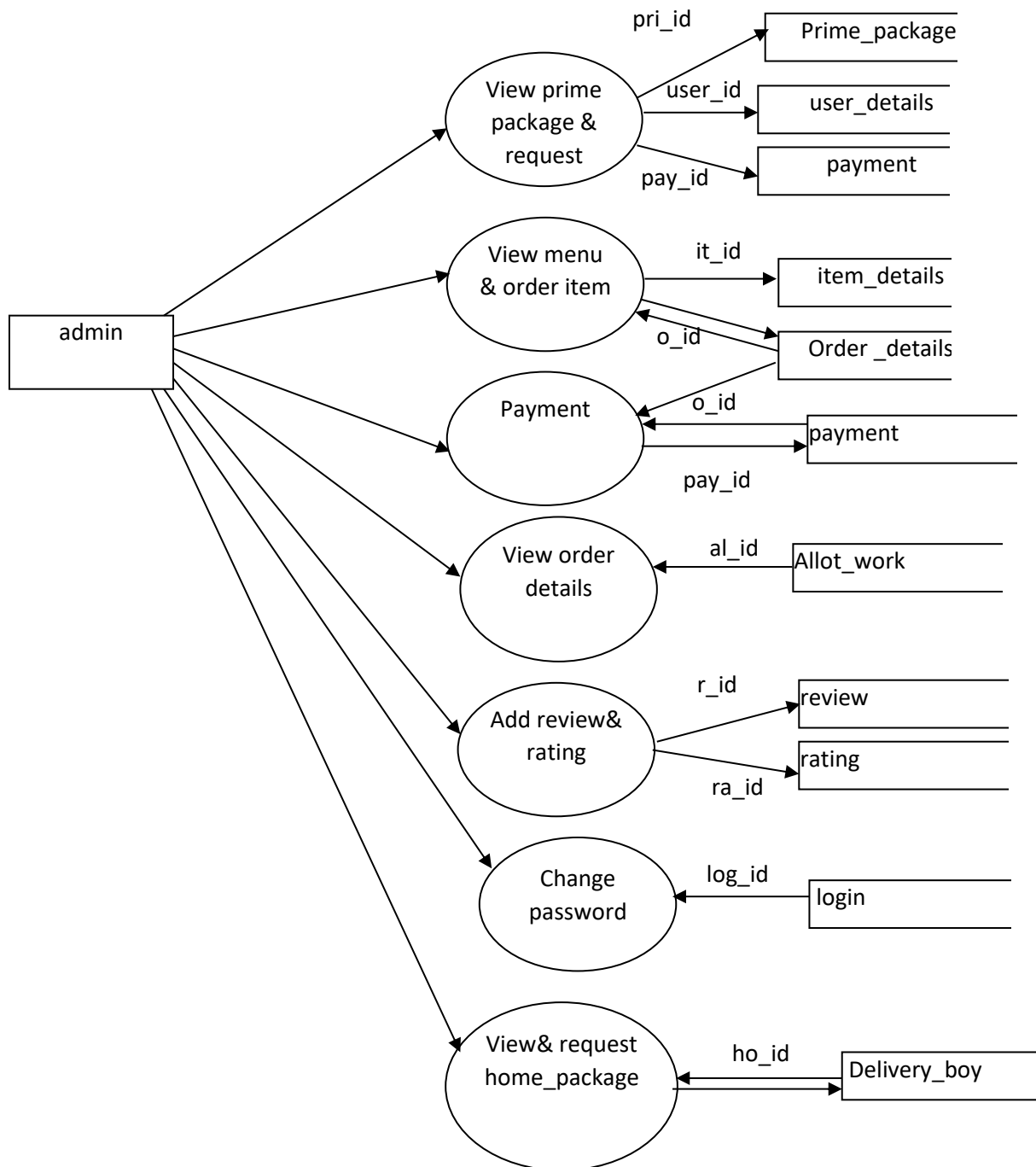
Level 2

Delivery Boy



Level 2

User

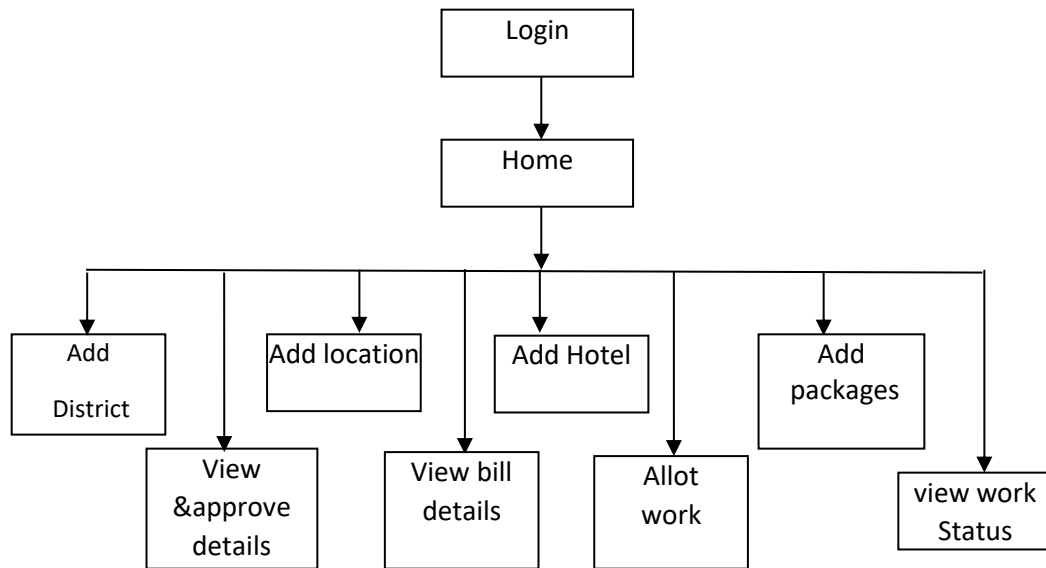


2.7 STRUCTURE CHART

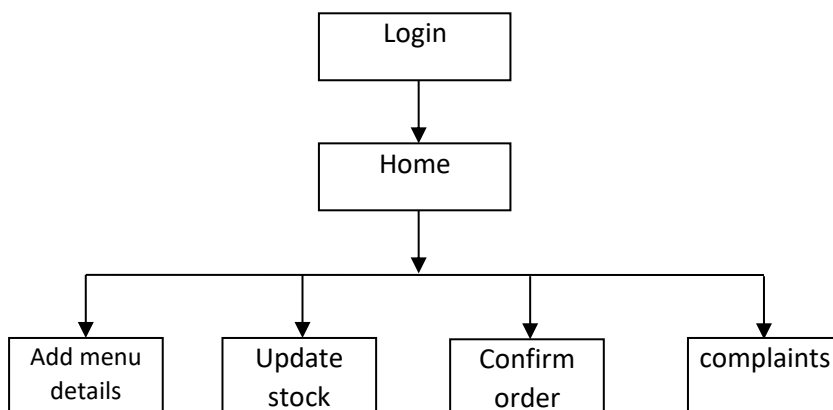
A structure chart is a design tool that pictorially shows the relation between processing modules in computer software. Describes the hierarchy of components modules and the data are transmitted between them. Includes analysis of input-to-output transformations and analysis of transaction. Structure charts show the relation of processing modules in computer software's. Structure charts are developed prior to the writing of program code. They identify the data passes existing between individual modules that interact with one another.

A Structure Chart (SC) in software engineering and organizational theory is a chart which shows the breakdown of the configuration system to the lowest manageable levels. This chart is used in structured programming to arrange the program modules in a tree structure. Each module is represented by a box, which contains the module's name. The tree structure visualizes the relationship between the modules. a structure chart is top down modular design tool, constructed of squares representing the different modules in the system, and lines that connect them. The lines represent the connection and or ownership between activities and sub activities as they are used in organization charts.

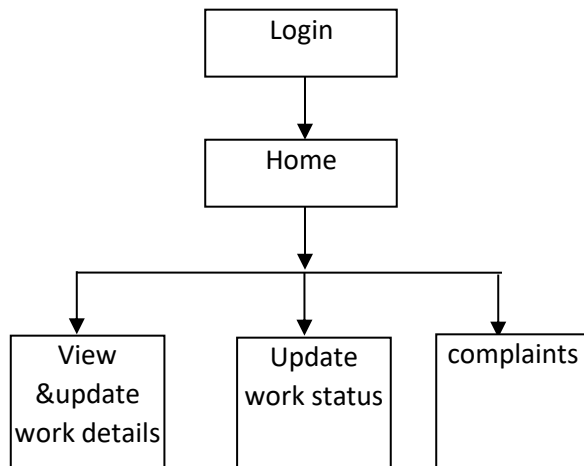
ADMIN



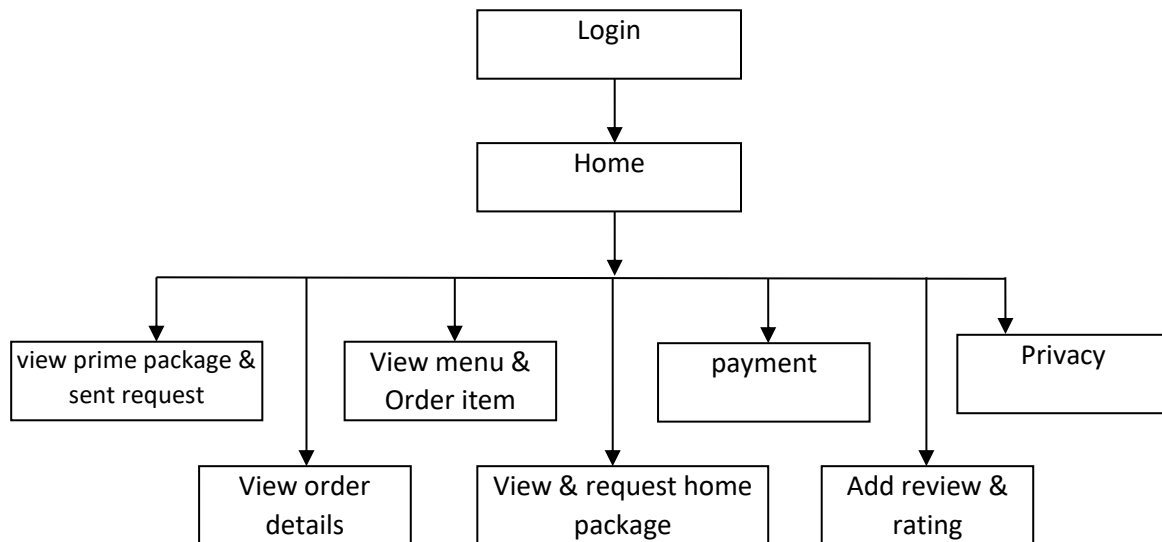
HOTEL



DELIVERY BOY



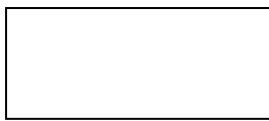
USER



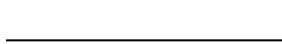
2.8 MENU TREE

Menu Tree is also helpful for representing the simplified version of the system. Which is in the form of a tree structure? Here firstly the entire system taken as the main part, and then each modules come in that system can be represented as branches, and then functions of each module can be represented as same as leaf in the tree. So menu is the hierarchical representation of entire system, so it is very helpful for identifying the system easily. In this system Event Management is the main branch, and administrator, client, manager as the sub branches, the functioning of each module can be represented also.

Symbols used in Menu Tree:

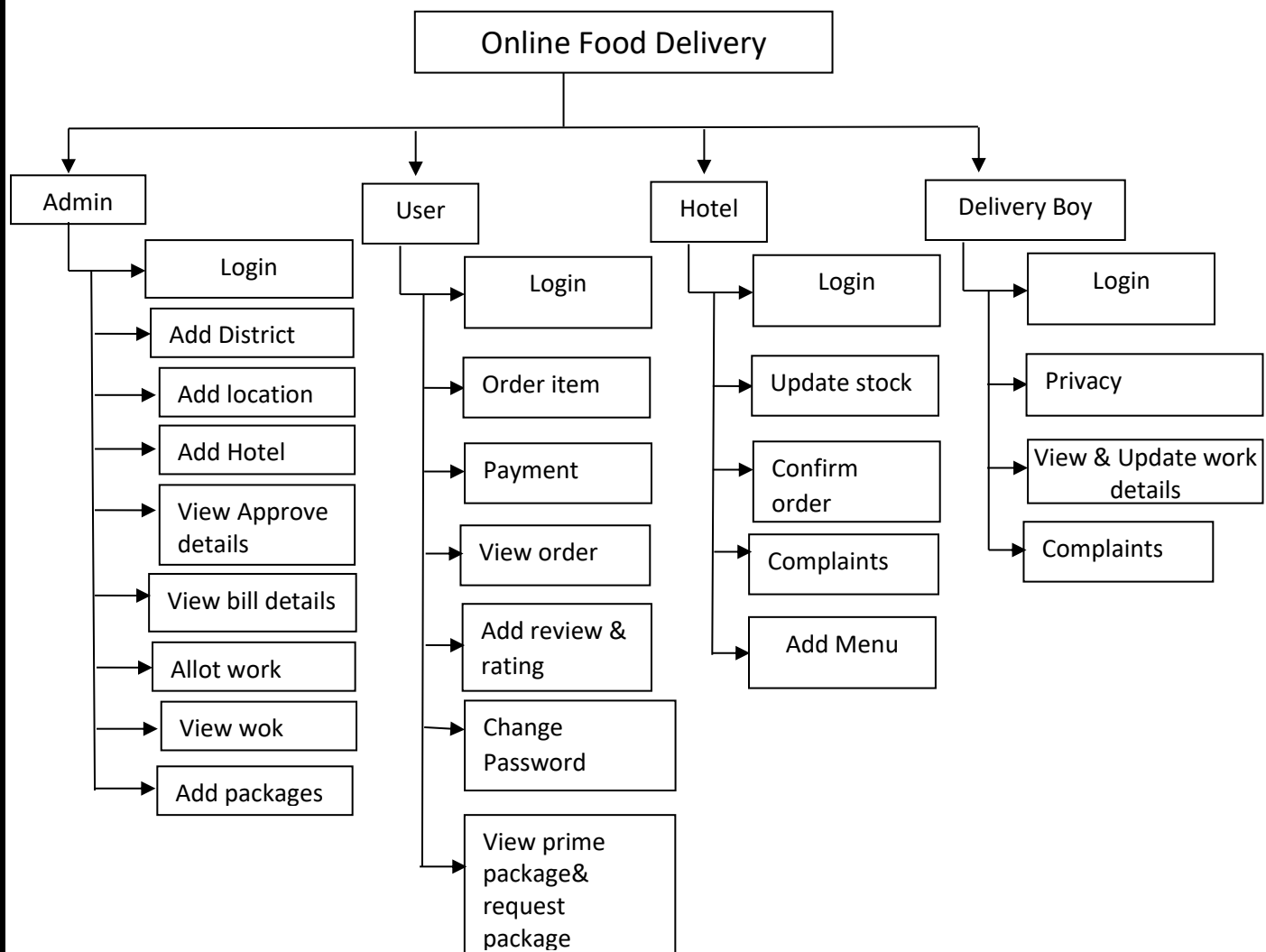


A rectangle represents the

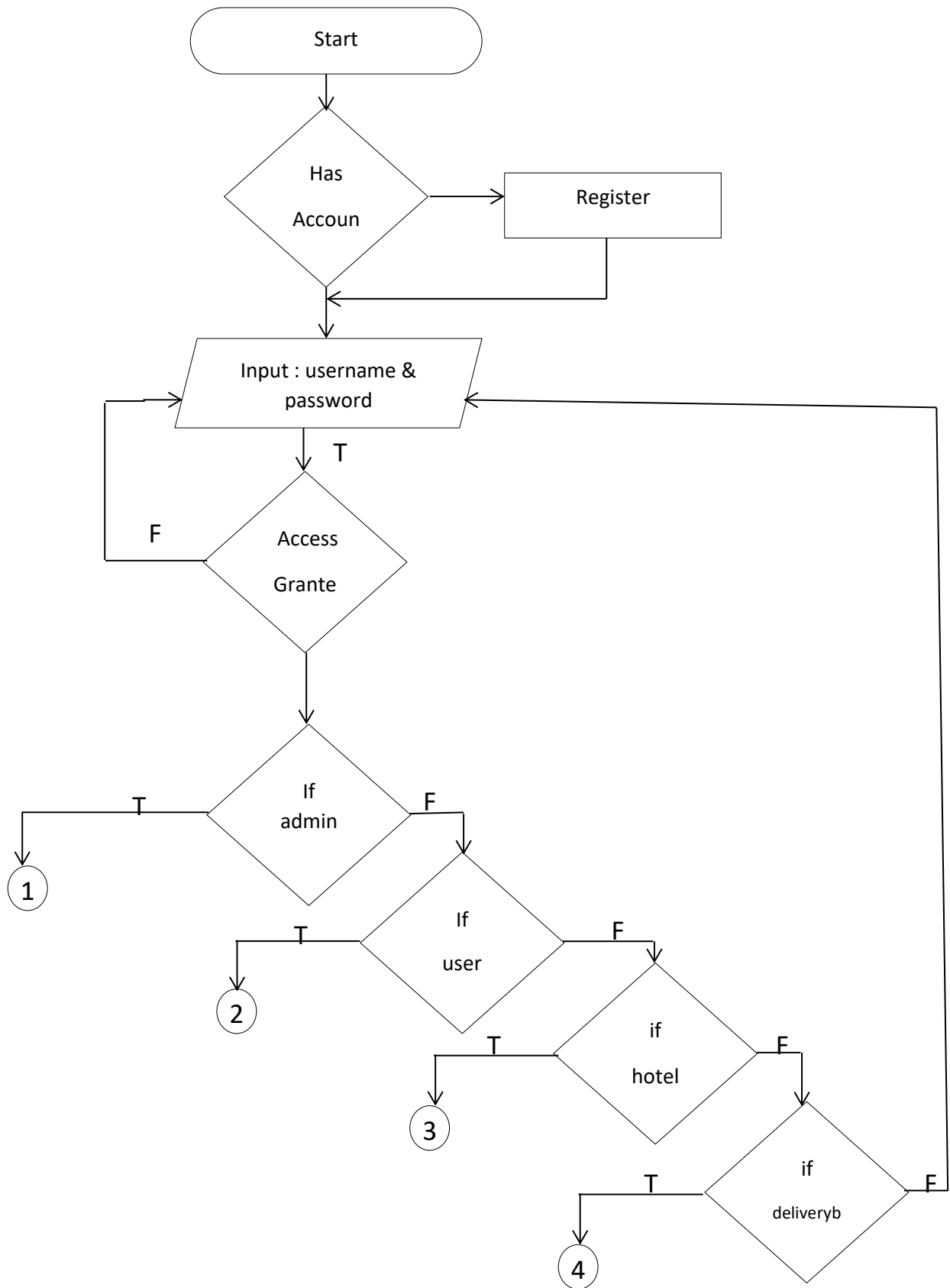


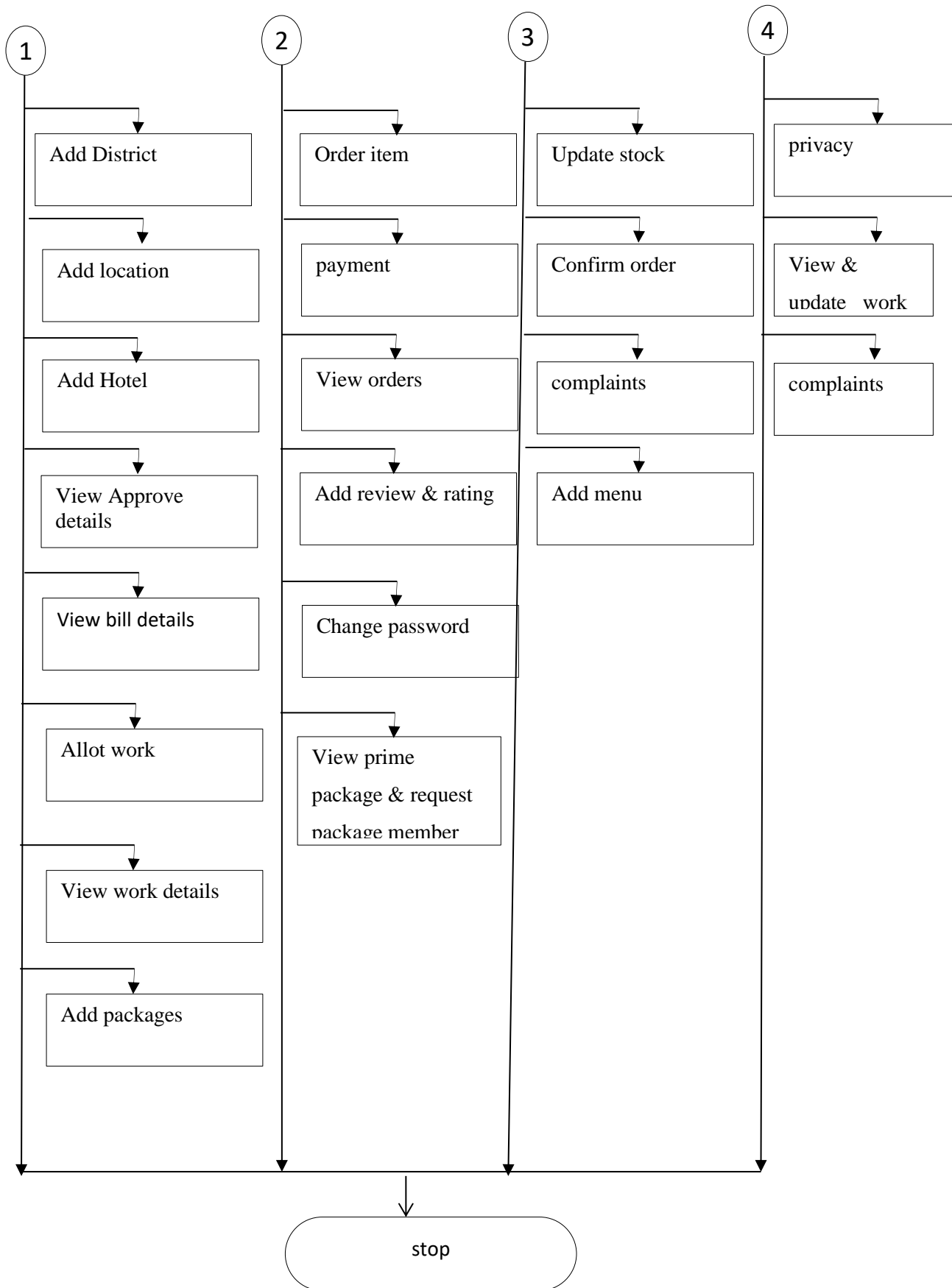
A line represents the connection

3.8 MENU TREE



2.9 SYSTEM FLOW CHART





CHAPTER-3

SYSTEM DESIGN

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

3.1 MODULE DESCRIPTION

1. Admin Level

- Add district
- Add location
- Add hotels
- View and approve details
- Add packages
- View bill details

- Allot work
- View work status

2. User Level

- View prime package & request package member
- View menu & order item
- Add & View Payment
- View order details.
- Add review & rating
- View & request Home package
- Profile updation

3. Hotel Level

- Add Menu details.
- Update stock.
- Confirm order.

4. Delivery Boy Level

- View & update work details.
- Privacy
- Add & view Complaints.

3.2 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspection the computer to read data from a written or printed documents or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the input required, controlling the errors, avoiding delay, avoiding extra steps and

keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Method for preparing input validations and steps to follow when error occur.

3.2.1 Input Objectives

Input design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

- It is achieved by the creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides records viewing facilities.
- When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

3.3 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In outputs design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can easily and effectively.

3.3.1 Output objectives

- Identify the specified output that is needed to meet the requirements.
- Select methods for presenting information.
- Create documents, report, or other formats that contain information produced by the system.
- The output form of an information system should accomplish one or more of the following objectives.
- Convey information about past activities, current status or projections of the future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.
- view the Payment details
- It produces after Bill payments

3.4 DATABASE DESIGN

The most important aspects of building an application is the design of tables or database schema. The data stored in the table must be organized in the same manner, which is meaningful. The overall objectives in the process of the table design have been to treat as organizational resources and as integrated whole.

The organization of the data in a database aims to achieve five major objectives that are given below,

- Data Integrity
- Data Independency
- Controlled Redundancy
- Privacy and Security
- Performance

TABLE LIST

List Of Attributes In Each Table

Table: Login

Field Name	Data Type	Constraints	Description
Loginid	int(11)	Primary Key	Login id
Username	int(11)	Not null	username
Password	varchar(10)	Not null	password
Role	varchar(10)	varchar(10)	role

Table: district

Field Name	Data Type	Constraints	Description
D_id	int(11)	Primary Key	District id
District	int(11)	Not null	Name of district

Table: location

Field Name	Data Type	Constraints	Description
l_id	int(11)	Primary Key	Location id
Location	int(11)	Not null	Name of location
D_id	int(11)	Foreign Key	District id

Table: Hotel

Field Name	Data Type	Constraints	Description
h_id	Int(5)	Primary Key	Hotel ID
Logid	Bigint(100)	Foreign Key	User id
Hotelname	Varchar(100)	Not null	Content
H_address	Varchar(35)	Not null	Interval
Licenseno	Varchar(35)	Not null	License number
Upload_license	Varchar(35)	Not null	Upload License
Phno	Int(10)	Not null	Phone No
Pinno	Int(10)	Not null	Pin Number
Owner_name	Varchar(30)	Not null	Owner Name
Owner_address	Varchar(100)	Not null	Owner Address
O_phno	Int(10)	Not null	Owner's Photo
O_pinno	Int(10)	Not null	Pin Number of Owner
O_aadhar_img	Varchar(10)	Not null	Aadhar photo

Table: item_details

Field Name	Data Type	Constraints	Description
Item_id	varchar(11)	Primary Key	Item ID
Name	Varchar(50)	Not null	Item Name
Details	Varchar(100)	Not null	Item Details
Item_type	Varchar(50)	Not null	Item Item_type
Price	Int(11)	Not null	Item Price
Image	longblob	Not null	Item Image
Count	Int(11)	Not null	Item Count

Table: order_details

Field Name	Data Type	Constraints	Description
O_id	Int(11)	Primary Key	Order details ID
Item_id	varchar(11)	Foregin Key	Item ID
User_id	Int(11)	Foregin Key	User ID
Delivered_address	Int(5)	Not Null	Delivered address
Date	date	Not null	Date
Time	time	Not null	Time
Status	Varchar(30)	Not null	Status

Table: Complaint

Field Name	Data Type	Constraints	Description
c_id	Int(11)	Primary Key	Complaint id
U_id	int(30)	Not null	User id
Complaint	Varchar(20)	Not null	Complaint
Reply	Varchar(20)	Not null	Reply
Status	Varchar(20)	Not null	Status

Table: Review

Field Name	Data Type	Constraints	Description
Rid	Int(11)	Primary Key	Review id
Reviews	int(30)	Not null	Reviews
O_id	Varchar(20)	Foreign Key	Order id

Table: Rating

Field Name	Data Type	Constraints	Description
Ra_id	Int(11)	Primary Key	Rating id
Rating_no	int(30)	Not Null	Rating Number
O_id	Varchar(20)	Foreign Key	Order id

Table: Bill

Field Name	Data Type	Constraints	Description
<i>Bill_id</i>	int(11)	Primary key	Bill ID
Order_id	int(11)	Foreign Key	Order ID
Totalamount	Int(11)	Not null	Total Amount
Status	varchar(20)	Not null	Status

Table: Allot_work

Field Name	Data Type	Constraints	Description
<i>Al_work</i>	int(11)	Primary key	Bill ID
Bill_id	int(11)	Foreign Key	Order ID
db_id	Int(11)	Foreign Key	Total Amount
Date	Date	Not null	Date of Bill
Time	Time	Not null	Bill Time
Status	varchar(20)	Not null	Status

Table: Payment_details

Field Name	Data Type	Constraints	Description
P_id	int(11)	Primary key	Payment id
O_id	int(11)	Foreign Key	Order ID
Payableamount	Int(11)	Not null	Total Amount
Modeofpayment	Date	Not null	Payment mode
Paymentstatus	Time	Not null	Payment status
Date	date	Not null	date

Table: Delivery_boy

Field Name	Data Type	Constraints	Description
db_id	int(11)	Primary Key	Delivery Boy ID
Log_id	int(11)	Not null	Login ID
Name	varchar(10)	Not null	Name
Address	varchar(50)	Not null	Waiter's Address
phone	Int(20)	Not null	Waiter's Phone Number
Pinno	Int(5)	Not null	Pin code
Upload_aadhar	Varchar(25)	Not null	Upload aadhar
Upload_driving_license	Int(15)	Not null	Upload driving license
Dob	date	Not null	dob
Email	varchar(200)	Not null	email
Status	Varchar(25)	Not null	Status
Work_status	Varchar(25)	Not null	Work status

Table: Hotel

Field Name	Data Type	Constraints	Description
h_id	Int(5)	Primary Key	Hotel ID
Id	Bigint(100)	Foreign Key	User id
Hotelname	Varchar(100)	Not null	Content
H_address	Varchar(35)	Not null	Interval
Licenseno	Varchar(35)	Not null	License number
Upload_license	Varchar(35)	Not null	Upload License
Phno	Int(10)	Not null	Phone No
Pinno	Int(10)	Not null	Pin Number
Owner_name	Varchar(30)	Not null	Owner Name
Owner_address	Varchar(100)	Not null	Owner Address
O_phno	Int(10)	Not null	Owner's Photo
O_pinno	Int(10)	Not null	Pin Number of Owner
O_aadhar_img	Varchar(10)	Not null	Aadhar photo

Table: user

Field Name	Data Type	Constraints	Description
<i>U_id</i>	Int(11)	Primary key	User details id
Loginid	int(30)	Foreign Key	Login id
name	Varchar(30)	Not null	Name
Address	Varchar(40)	Not null	Address
Phone	int(11)	Not null	Phone No
Email	Varchar(20)	Not null	Email ID
pinno	Int(15)	Not null	Pin number
Status	Varchar(30)	Not null	Status

Table: bank

Field Name	Data Type	Constraints	Description
acc_no	Varchar(50)	Primary Key	Account Number
card_no	Varchar(50)	Unique key	Card Number
Cvv	Int(10)	Not null	CVV
Exmonth	Varchar(50)	Not null	Expiry Month
Exyear	Varchar(30)	Not null	Expiry Year
card_type	Varchar(20)	Not null	Type of Card
Holdername	Varchar(100)	Not null	Card Holder Name
Bank_balance	Int(100)	Not null	Balance Amount

Table: home_package

Field Name	Data Type	Constraints	Description
<i>h_id</i>	Int(11)	Primary key	Home id
Logid	int(30)	Foreign Key	Login id
Home_name	Varchar(30)	Not null	Name of home
amount	Varchar(40)	Not null	Amount

Table: prime_package

Field Name	Data Type	Constraints	Description
<i>pr_id</i>	Int(11)	Primary key	Prime package id
Logid	int(30)	Foreign Key	Login id
p_name	Varchar(30)	Not null	Name of prime package
amount	Varchar(40)	Not null	Amount
Discount	Varchar(15)	Not Null	Discount

3.4.1 Normalization

Normalization is the process of decomposing the attributes in an application, which results in a set of tables with very simple structure. The purpose of normalization is to make tables as simple as possible.

Decomposing is the process of splitting relations into multiple relations to eliminate anomalies and maintain anomalies and maintain data integrity. To do this we use normal forms or rules for structuring relation.

- **Insertion anomaly:** Inability to add data to the database due to absence of other data.
- **Deletion anomaly:** Unintended loss of data due to deletion of other data.
- **Update anomaly:** Data inconsistency resulting from data redundancy and partial update.

- **Normal Forms:** These are the rules for structuring relations that eliminate anomalies.

3.4.1.1 FIRST NORMAL FORM:

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. By this we mean simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

3.4.1.2 SECOND NORMAL FORM:

A relation is said to be in second Normal form if it is in first normal form and it should satisfy any one of the following rules.

- 1) Primary key is not a composite primary key
- 2) No non key attributes are present
- 3) Every non key attribute is fully functionally dependent on full set of primary key.

3.4.1.3 THIRD NORMAL FORM:

A relation is said to be in third normal form if there exists no transitive dependencies.

Transitive Dependency: If two non key attributes depend on each other as well as on the primary key then they are said to be transitively dependent.

The above normalization principles were applied to decompose the data in multiple tables thereby making the data to be maintained in a consistent state.

CHAPTER-4

SYSTEM TESTING

4.1 UNIT TESTING

Unit testing is undertaken when a module has been coded and successfully reviewed. Here we test each module separately in order to test a single module we need a complete environment to provide all that is necessary for execution of the module. That is besides the module under test, we will need the following in order to be able to test the module.

- The procedures belong to other modules that the module under test call.
- Non local data structures that the module accesses.
- A procedure to call the function of the module under test with appropriate parameters.

4.2 INTEGRATION TESTING

The primary objective of integration testing is to test the module interfaces in order to ensure that there are no errors in the parameter passing .when one module invokes another module. During integration testing, different modules of a system are integrated in a planned manner using an integration plan. Integration plan specifies the steps and the order in which modules are combined to realize the full system. After each integration step, the partially integrated system is tested.

4.3 SYSTEM TESTING

In this phase, the entire software system was tested. After integration testing, the entire software system was tested against various clients. The software has been tested for its functionality as well as limitation. The various interfaces developed were thoroughly debugged and were found to be working correctly.

CHAPTER-5

SYSTEM IMPLEMENTATION

The implementation is the final and important phase. It involves user training, system testing in order to ensure successful running of the proposed system. Once the system design phase is over, the next stage is to implement and monitors the operation of the system to ensure that it continues the work effectively and efficiently.

The three main phases in implementation take place in series. These are the initial installation, the test of the system as a whole and evaluation maintenance and control of the system. The implementation plan and action to implement should be bound closely together. The implementation plan is a function of line management at least as far as key decision or alternative plans are concerned.

The implementation plan was to convert the existing clerical files to the computer. The implementation plan listed all sub tasks so that individuals in the organization may be assigned specific responsibilities.

The installation of the new system that is bound to replace the current one may require a major revision of computer facilities as well as completely new after space. Space planning took into account the space occupied by the people, space by equipment and the movement of people and equipment in the working investment. After conduction the initial testing the system is loaded on the client office's computer. Some of the user employees in this case are selected. These users are trained first and they run the system. A detailed documentation is prepared to this set of employees. There may be slight modifications to meet the organization.

After all modifications specified by the users in the documentation are made, the computer system is run along with manual system. Even though this kind of parallel run make extra burden to the employees and management, the system is run in parallel for the sake of checking reliability and efficiency. After this document, which compares the result of the manual system with those of the computerized is prepared. If there is any modifications are made as needed.

A procedure is developed for delivering instructions and forms to supervisors for coordination and integrating the proposal with other parts of the organization, and for working out of problems with people involved. This procedure also helped for evaluation of hardware

and software. A program was developed to emphasis the nature and goals of the new system on the management and the support personnel and train operation personnel in their new tasks.

In the case of management many of whom participated in the development of the system short seminars were given. Particular attention was paid to the training of end users. The training sessions were aimed at giving the user staff the specific skills required in their new jobs. They were given practical training to have a thorough understanding of what the new system is like and how it behaves.

Education involved creating the right atmosphere and motivation of user staff. It explained the need for changes and helped to overcome the resentment caused by the feeling that computers took away the responsibility from individual departments.

Various measures have been taken by department officials in order to find suitable solutions by the following issues:

- About the skill to be acquired.
- Reduction of man power in department
- About the new form having all required option.

CHAPTER-6

SYSTEM MAINTENANCE

The term maintenance is used to describe the software engineering activities that occur following the delivery of a software product to the customer. The maintenance phase of the software lifecycle is the time period in which software performs useful work. We may define maintenance by describing four activities that are undertaken after a program is released for use. During the use of any large program, errors will occur and be reported to the developer. The process that includes the diagnosis and correction of one or more error is called corrective maintenance. Thus problem correction involves modification and revalidation of software to corrective maintenance.

The second activity that contributes to a definition of maintenance occurs because of the rapid change that is encountered in every aspect of computing. An activity that modifies software to properly interface with a changing environment is adaptive maintenance. The third activity occurs when a software package is successful. As software is used recommendations for new capabilities, modifications to existing function and general enhancements are received from users. To satisfy request in this category, perfective maintenance is performed. The fourth and last maintenance activity occurs when software is changed to improve future maintainability or reliability or to provide activities including correcting, coding and design, updating, documentation and test data and upgrading user support. Maintenance means restoring something to its original conditions. In contrast, enhancements mean adding or modifying or reexecution.

6.1 Types of system maintenance

6.1.1 Corrective maintenance

Corrective maintenance of a system becomes necessary to rectify the bugs while the system is in use.

6.1.2 Adaptive maintenance

A system might need maintenance when the customers need the product to run on new platforms, on new operating system, or when they need the product to be interfaced with new hardware or software.

6.1.3 Perfective maintenance

A system needs maintenance to support the new features that users want it to support to change different functionalities of the system according to customer demands, or to enhance the performance of the system.

6.1.4 Preventive maintenance

This occurs when software is changed to improve future maintainability or reliability or to provide a better basis for future enhancements.

CHAPTER-7

CONCLUSION

The project was successfully completed within the time span allotted. Every effort has been made to present the system in more users-friendly manner. All the activities provides a feeling likes an easy walk over to the user who is interfacing with the system. A trial run of the system has been made and is giving good results.

The software has been developed in PHP. All the modules are tested separately and put together to form the main system. The system has been developed in an attractive dialogue fashion. So users with minimum knowledge about the computers operate the system easily.

CHAPTER-8

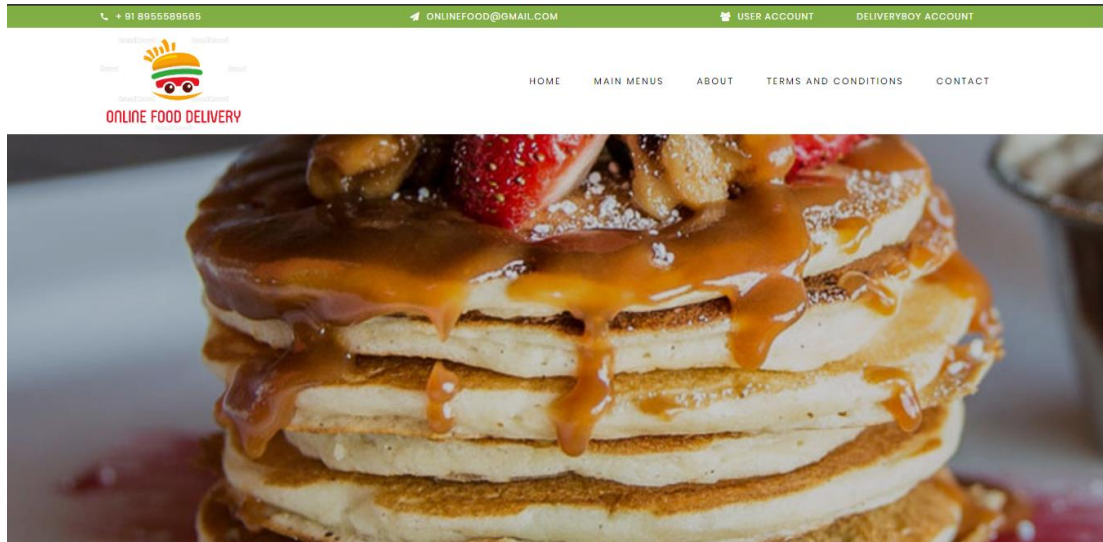
FUTURE ENHANCEMENT

In real life situations, we may face some sort of difficulties while making food orders in a hotel. Waiting for the waiter to arrive is itself a time consuming process. Also the menu card may lack details like the chef who prepared it and its recipes. Updating the printed rates and details in the menu card is also an expensive work. as software is used, the customer/user will recognize additional functions that will provide benefit. Perceptive maintenance extends the software beyond its original functional requirements. For example we can add the directions to cook. Also an online shopping of the recipes of the food item along with directions to prepare can also be included.

The project made here is just to ensure this product could be valid in real challenging world. Here all the facilities are made tested. Every module in the system is being developed carefully such that the future enhancements do not affect the basic performance of the system. In future we can add any links or services to the System very easily.

APPENDIX A: SCREEN LAYOUTS

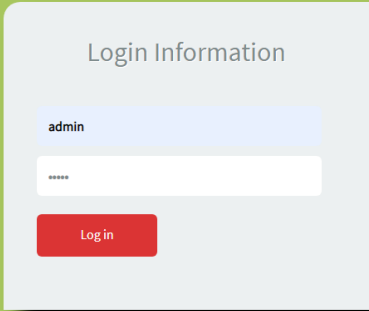
User Homepage



Register

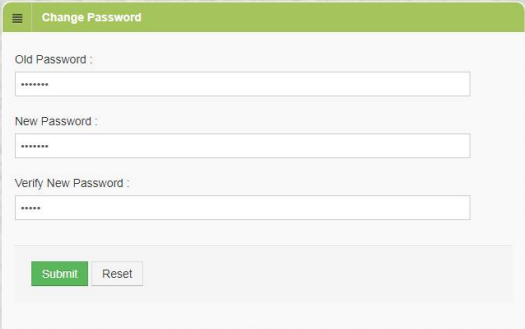
A screenshot of the 'Register' form on the 'Online Food Delivery' website. The form is divided into two main sections: 'Name' and 'Address' on the left, and 'Login' on the right. The 'Name' section includes fields for 'Name' (containing 'sarath'), 'Phone' (containing '8956895623'), and 'Pin Code' (containing '691004'). The 'Address' section includes fields for 'Address' (containing 'test address') and 'Email Address' (containing 'sarath@gmail.com'). Below the 'Email Address' field is a note: '* Your Email Address is our Username'. The 'Login' section includes fields for 'Username/Email *' (containing 'sarath@gmail.com') and 'Password *' (containing '*****'). A green 'LOG IN' button is located below the password field. A green 'REGISTER' button is located below the 'Pin Code' field. At the bottom center of the form, there is a green circular button with an upward-pointing arrow.

Admin login




A login form titled "Login Information" is centered on a green background. The form has a light gray border and a shadow. It contains a text input field with the username "admin", a password input field with masked characters "*****", and a red "Log in" button.

Change password



A "Change Password" form is displayed within a web application interface. The form has a green header bar with a menu icon and the title "Change Password". It contains three input fields: "Old Password :", "New Password :", and "Verify New Password :". Each field has a masked password "*****". Below the fields are two buttons: a green "Submit" button and a gray "Reset" button.

List district



A table titled "District List" is shown within a web application interface. The table has three columns: "Sl No", "District Name", and "Action". There are two rows of data. The first row shows "1" for Sl No and "Alappuzha" for District Name, with an "Action" button. The second row shows "2" for Sl No and "kollam" for District Name, with an "Action" button. An "Add District" button is located at the top right of the table.

Sl No	District Name	Action
1	Alappuzha	Action
2	kollam	Action

APPENDIX B: TEST CASES

Sl.No	Test Case	Test Procedure	Precondition	Expected Result	Passed/ Failed (Yes/No)
1	Login Page	To check whether the control from the login screen goes to the main menu	Enter a valid user name and password on the login screen	The control should go to the home page	yes
2	User Registration	To check whether the control goes to the user registration screen when the user select the registration from home page	Select registration from the home page	The control should go to registration page	Yes
3	Order	There is a option in the users home screen for order food. Here the user can order food items.	Select order page from the user's home page	The control should go to booking page	Yes
4	Make payments	After ordering food a particular user is responsible for making payment.	Select payment option from user page	The control should go to payment page	yes
5	Add review	Here the users can add their reviews about foods	Select review option from the user home page	The control should go to complaint page	Yes

CHAPTER-9

BIBLIOGRAPHY

Text books

- PHP complete reference - Steven Holzner
- PHP and MySQL Bible - Tim Converse, Joy Spark
- PHP Essentials - Julice Meloney
- Software Engineering - Pressman

Websites

- <http://www.php.net>
- <http://www.sql.com>