

ONLINE HEALTH SERVICES

SUBMITTED BY

KAJAL SHASHIKANT WAGHMODE

[Seat No. A039]

DATE OF SUBMISSION

1st November -2018

Under the guidance of Prof. NANDA RUPNAR


KELKAR EDUCATION TRUST'S
VINAYAK GANESH VAZE COLLEGE
OF ARTS, SCIENCE & COMMERCE
Mithagar Road, Mulund(E), Mumbai - 400 081
(Affiliated to University of Mumbai)

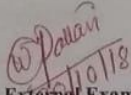
DEPARTMENT OF INFORMATION TECHNOLOGY

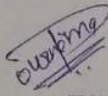


CERTIFICATE

This is to certify that the project entitled, "ONLINE HEALTH SERVICES",
is bonafied work of KAJAL S. WAGHMORE bearing
Seat.No: A039 is submitted in partial fulfilment of the requirements for the
award of degree of **BACHELOR OF SCIENCE** in **INFORMATION TECHNOLOGY**
from University of Mumbai.


Internal Guide


External Examiner


HOD



Date: **30 OCT 2018**

College Seal

Index:

Part 1: Synopsis

Ch. no	Chapter name	Page number
1	INTRODUCTION	3
1.1	Background	3
1.2	Objectives	3
1.3	Purpose, Scope, Applicability	3
1.3.1	Purpose	3
1.3.2	Scope	4
1.3.3	Applicability	4
1.4	Achievements	4
1.5	Organization of report	4

Part 2:Main Report

Ch. no	Chapter name	Page number
2	Survey Of Technologies	5
3	Requirements and Analysis	8
3.1	Problem Definition	8
3.2	Requirements Specification	8
3.2.1	Functional Requirements	9
3.2.2	Non-functional requirements	9
3.2.3	System Requirement	13
3.3	Planning and scheduling	14
3.3.1	Gantt chart	13
3.4	Software and Hardware Requirements	14

	3.4.1	Hardware requirement	14
	3.4.2	Software requirement	14
3.5		Conceptual models	15
	3.5.1	Feasibility Study	15
	3.5.2	ER Diagram	18
	3.5.3	Data Flow Diagram	22
	3.5.4	Class Diagram	28
	3.5.5	Sequence Diagram	30
	3.5.6	Use Case	34
	3.5.7	Activity Diagram	38
4		System Design	40
	4.1	Schema Design	40
	4.2	Data Integrity and Constraints	41
	4.3	User Interface design	44

Part 3: Annexures

Ch. no	Chapter name	Page number
	Table Of Contents	49
	Table of Diagrams	49
	Bibliography	50
A	Book References	50
B	Web References	50

Online Health Services

Chapter 1:Introduction

1.1 BACKGROUND:

The existing work of the online health services are they are provide only the prescription about health related issues.

Here, this system provides the proper diet plan of the day.

This system is going to generate the diet plan.

If the user is not satisfied with the given diet plan then user can ask to dietitian for their diet.

User can also calculate its BMI through the system.

1.2 OBJECTIVES:

- To provide the proper diet plan to the user
- To give the effective dietary guidelines for managing the health issues and to provide the appropriate diet plan
- To aware the people about the proper diet plan and importance of diet in daily busy life
- To make the things easier for user and the dietician
- To reduce the time of user and dietitian

1.3 PURPOSE, SCOPE AND APPLICABILITY:

1.3.1 PURPOSE:

- To maintain the proper diet and health of the people this website going to help
- The people who do not afford the dietitian for them this website is affordable
- The people who want to gain the knowledge about diet for them this website going to help

1.3.2 SCOPE:

- The use of the system is reduce the time

- It is a cost saving
- Website is available 24/7s
- Through the use of this website user would be more health conscious
- There is no age limitation for using this website

1.3.3APPLICABILITY:

- This system provides the proper diet plan for user.
- This system makes the task easier for user and dietitian.
- System is also applicable for reduce time to get the proper health prescription.
- People awareness about health conscious.
- Dietitian can also provide the proper diet to user.

1.4 ACHIEVEMENTS:

- For the people who do not afford the high paid dietitians for them this website is going to use.
- This will make the people more conscious about their diet.
- This website will save the time of both user and dietitian.

1.5ORGANIATION OF REPORT:

- After this introductory chapter, chapter2 describes the Survey of Technology of this project.
- Chapter3 summarizes Requirement specification and analysis which includes Software, Hardware and System requirement as well as Planning and Scheduling for the project.
- Chapter4 describes System Design in detail using various diagrams.

CHAPTER 2-SERVEY OF TECHNOLOGY:

2.1 Microsoft Visual Studio Community 2010:



Diag2.1ASP.net

Microsoft visual studio is found to be a very effective tool for developing websites, web application as well as mobile application. It provides drag and drop facility which helps to design the application easily.it supports all commonly programming languages. With help of visual studio we can design our application and there by instantly look whether the output is exactly what we expected.

I will be using c# language and the .net framework of Microsoft visual studio community 2010 for developing the front end of my application using it.

It has all facilities and tools provided by Microsoft that I, why it is better than java and python platform. It is open source server side web application framework designed for web development to produce dynamic web pages

2.2 Microsoft SQL server Database:



Diag2.2 SQL Server

The Microsoft SQL server is a relational database management system developed by Microsoft. It helps us with creating editing and maintaining the database. It will use SQL server on 'www.viewen.com' as my application is a client-server application.

To manipulate the data in database I will use 'SQL Server Management Studio' which is freely available on internet. I will use it for the back end application.

CHAPTER 3-REQUIREMENTS AND ANALYSIS

3.1 PROBLEM DEFINATION:

- The related problem is that the system is not going to help to user, user should have to fist choose the dietician and then dietician will provide the diet.
- This process is so time consuming because some time it may happen that at that time dietician may be unavailable.
- To reduce the above problem we are going to use the system generated diet.

3.2REQUIREMENTS SPECIFICATION:

3.2.1FUNCTIONAL REQUIREMENTS:

➤ For Admin:

- The admin of the system should login to the system.
- Admin can see all the registered user of the system.
- Admin can search any user from the register.

➤ For Customer:

- The customer can register themselves.
- User can login to system and get access to the website.
- User can able to calculate their BMI.
- User can get their diet plan.
- User can also see the diet chart and food pyramid.

➤ For Dietitian:

- Dietitian is add by admin.
- Dietitian can login with their login id and password.
- The user can also take the diet plan from the dietitian if they are not satisfied.
- The dietitian can only view the user details and dietitian can select the timing for the user for user diet.

3.2.2NON-FUNCTIONAL REQUIREMENTS:

➤ Performance:

- The response time, throughput and utilization these factors are comes under performance.

- The response time should be minimum and throughput is maximum.
- The reply to the action of user should be quick and less in time.
- **Scalability:**
 - It is the capability of a system, network or process to handle a growing amount of work or its potential to be enlarged to accommodate that growth.
 - In our system the system is said to be scalable if it provide the correct output under the increased load when resources are added.
- **Availability:**
 - The degree to which the system or equipment is in a specified operable and committable state at the start of the system.
 - The system is available at any time when the user is want to use or check it.
- **Reliability:**
 - The system should be reliable not only in terms of time but also at the time of accessing.
 - It cannot be provide an accidental exit from the given operable environment.
 - The quality of being trustworthy or of performing consistently well when user is in operable state.
- **Maintainability:**
 - It is in terms of to be do the correct defects or their causes.
 - It should prevent unexpected working condition and maximize the efficiency, reliability and safety.
- **Safety and Security:**
 - The data that the user will provide to the system should be keep the secure into the database from the unauthorized person.
 - Safety means if there is any problem of occurring the uncertain situation the safety should be provide the secure exit.
- **Usability:**

- The system should provide the usable approach to the user of that system.
- User should feel that the system is usable.

3.2.3 SYSTEM REQUIREMENTS:

➤ **Function : Registration**

- Description: user have to register in order to make use of system, then register user can login to the system anytime.
- Input: the details of user such as name, age, height, weight, email-id, phone-No.
Source: From the registration details input are collected.
- Output: user new account will be created.
- Destination: the database of the system.
- Action: fill the registration form properly and providing its necessary output to the user.
- Pre-Condition: user's personal details.
- Post-Condition: account will be created.

➤ **Function : Login**

- Description: user have to login to the system with their id and password.
- Input: user's id and password.
- Source: system will generate id and send it to the user's email-id and phone-no.
- Output: user will get their id and password and will login to the system.
- Destination: the database of the system.
- Action: providing the correct id and password to the system and get login to the system.
- Pre-Condition: user's id and password.
- Post-Condition: get login to the system.

➤ **Function: Disease Analysis**

- Description: users have to add if they have any disease such as hypertension, diabetes, thyroid etc.
- Input: The disease of the user that they have.
- Source: From the form of the disease will be collected.
- Output: Store to the database of the user account.
- Destination: database of the system.

- Action: select the proper disease that they have.
 - Pre-Condition: user disease.
 - Post-Condition: store it as information.
- **Function: BMI Calculation**
- Description: user have to enter their weight and height.
 - Input: user weight and height.
 - Source: From the calculation form of system.
 - Output: show is the user is normal, under-weight or over-weight.
 - Destination: database of the system.
 - Action: user enter the height and weight and display the appropriate answer.
 - Pre-Condition: should enter the height and weight.
 - Post-Condition: output will be display to the user.
- **Function: Setting Time For Taking The Food**
- Description: user will set the time of their breakfast, lunch, snacks and for dinner.
 - Input: user will select the time for food intake.
 - Source: data taken from the filled entry.
 - Output: System will take and use it for later.
 - Destination: database of the system.
 - Action: user select the time and diet will generate on the selected time.
 - Pre-Condition: user should fill the time.
 - Post-Condition: system should accept it.
- **Function: System Generated Diet**
- Description: user should get their diet or food as per provided timing.
 - Input: System has n number of food items according to the condition system should take the input.
 - Source: stored food items of the database.
 - Output: show the food to the user.
 - Destination: user display.
 - Action: after processing the diet plan of food will display to the user.

- Requirements: user should register themselves with the system.
- Pre-Condition: the system must have proper details and user should be login.
- Post-Condition: user get their diet plan.
- **Function: Dietitian Help**
 - Description: user wants to take the dietitian help.
 - Input: The registered dietitian should display to the user.
 - Source: the registered dietitian from the database.
 - Output: request will reach to the dietitian.
 - Destination: dietitian request list.
 - Action: user should hire any dietitian of the system.
 - Requirement: dietitian should registered with system.
 - Pre-Condition: the system should have a dietitian and user should registered with dietitian.
 - Post-Condition: request send to the right dietitian.
- **Function: Dietitian help to the user**
 - Description: dietitian will help to the user according to their need from veg and Non-veg.
 - Input: dietitian provide appropriate diet.
 - Source: from the set of the stored food of the database.
 - Output: send the diet to the user.
 - Destination: users diet plan details.
 - Action: dietitian prepare the diet plan for user and send it to the user.
 - Pre-Condition: dietitian should registered.
 - Post-Condition: diet is send to the appropriate user.

3.3PLANNING AND SCHEDULING:

Gantt chart: A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities displayed against time. On the left of the chart is list of the activities and along the top is a suitable time scale. Each activity is represented by a bar, the position and length of the bar reflects the start date, duration and end date of the activity.

Table of Gantt chart:

Task	Start Date	Duration	End Date
System Study And Analysis	20-june-2018	15 days	1-Jul-18
Understanding Existing System	3-july-2018	15 days	20-Jul-18
Problem Definition	22-Jul-18	15Days	10-Aug-18
Information Gathering	12-Aug-18	30Days	15-Sep-18
Database Design	17-Sep-18	15Days	30-Sep-18
Form Design	2-Oct-18	15Days	30-Oct-18
Coding	2-Nov-18	29Days	31-Dec-18
Testing	2-Jan-19	15Days	17-Jan-19
Modification Of Form	20-Jan-19	25Days	28-Feb-19
Implementation	2-Mar-19	9Days	22-Mar-19
Documentation	24-Mar-19	8Days	15-Apr-19

Table 3.1 Gantt chart

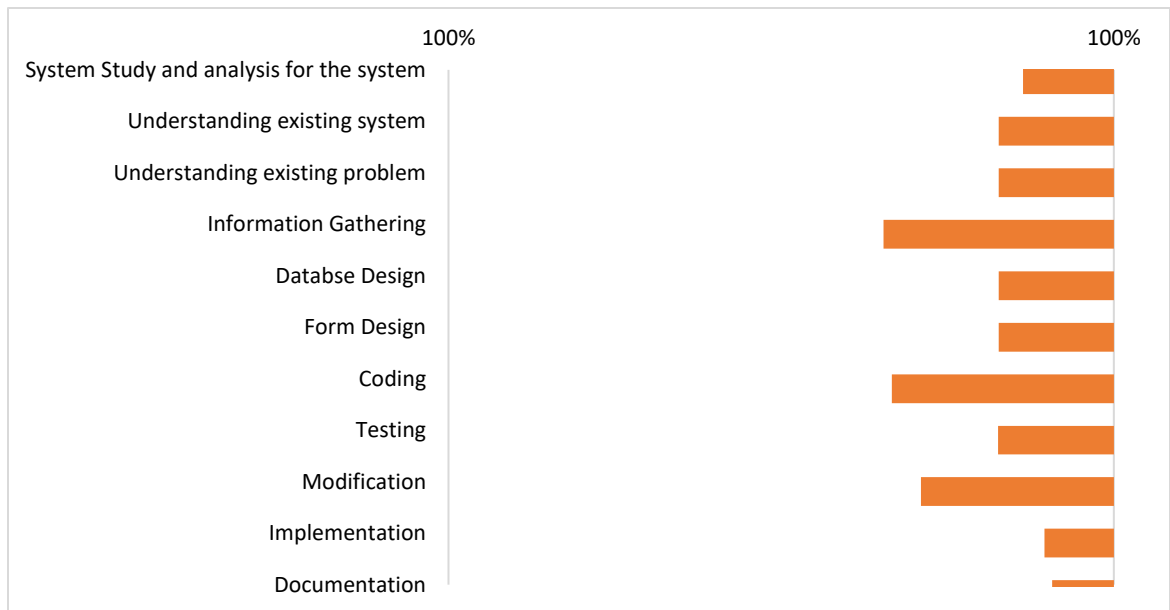


Fig 3.1 Gantt Chart

3.4HARDWARE AND SOFTWARE REQUIREMENTS:

3.4.1 SOFTWARE REQUIREMENTS

- Front-End: ASP.NET
- Back-End: SQL server
- Operating system: windows 10

4.2HARDWARE REQUIREMENTS

- Processor: Pentium processor and above
- RAM: 1GB or more
- Hard disk: 20GB or more
- Processor speed: 1.8 GHz or more.

3.5CONCEPTUAL MODEL:

3.5.1FEASIBILITY STUDY:

A project feasibility study is a comprehensive report that examines in detail the five frames of analysis of a given project. It also takes into consideration its four Ps, its risks and POVs, and its constraints (calendar, costs, and norms of quality). The goal is to determine whether the project should go ahead, be redesigned, or else abandoned altogether.

The five frames of analysis are: The frame of definition; the frame of contextual risks; the frame of potentiality; the parametric frame; the frame of dominant and contingency strategies.

The four Ps are traditionally defined as Plan, Processes, People, and Power. The risks are considered to be external to the project (e.g., weather conditions) and are divided in eight categories: (Plan) financial and organizational (e.g., government structure for a private project); (Processes) environmental and technological; (People) marketing and sociocultural; and (Power) legal and political. POVs are Points of Vulnerability: they differ from risks in the sense that they are internal to the project and can be controlled or else eliminated.

The constraints are the standard constraints of calendar, costs and norms of quality that can each be objectively determined and measured along the entire project lifecycle. Depending on projects, portions of the study may suffice to produce a feasibility study; smaller projects, for example, may not require an exhaustive environmental assessment.

There are four types of feasibility study:

1. Operational feasibility study
2. Technical feasibility study
3. Schedule Feasibility study
4. Cost feasibility study

1. Operational Feasibility Study:

Operational feasibility study is mainly focuses on to analyze and determine whether and how well the organization needs can be met by completing the project.

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

The operational feasibility assessment focuses on the degree to which the proposed development project fits in with the existing business environment and objectives with regard to development schedule, delivery date, and existing business processes.

To ensure success, desired operational outcomes must be imparted during design and development. These include such design-dependent parameters as reliability, maintainability, supportability, usability, reducibility, disposability, sustainability, affordability and others. These parameters are required to be considered at the early stages of design if desired operational behaviors are to be realized. A system design and development requires appropriate and timely application of engineering and management efforts to meet the previously mentioned parameters. A system may serve its intended purpose most effectively when its technical and operating characteristics are engineered into the design. Therefore, operational feasibility is a critical aspect of systems engineering that needs to be an integral part of the early design phases.^[11]

For this project operational feasibility is as follows:

- The user interface of the system is not more complex and also not so critical to be understood by the user.
- User can handle the system at any time.
- Diet pyramid and table are included to help the user to be more informative about their diet.
- User can easily understand their diet by looking at once.
- User can easily navigate through the system and can handle the system very smoothly.
- User can easily grab the requirements of system by looking at once.

2. Technical Feasibility Study:

Technical feasibility study helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working system.

This assessment is based on an outline design of system requirements, to determine whether the company has the technical expertise to handle completion of the project.

The technical feasibility assessment is focused on gaining an understanding of the present technical resources of the

organization and their applicability to the expected needs of the proposed system. It is an evaluation of the hardware and software and how it meets the need of the proposed system

For this project the Technical feasibility is as follows:

- The system project is website which uses internet and without it the site cannot be open on any system.
- The site can be accessible from laptop, PC, mobile phones.
- Technical resources such as operating system can access this system but windows 7 onwards.
- To make this project the hardware such as RAM is should be minimum 4 GB and or greater than this. The processor includes minimum i3 or onwards.

3. Schedule Feasibility Study:

Schedule feasibility is a feasibility in which an organization estimates how much time the project will take to complete.

A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is.

Some projects are initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable.

For this system the schedule feasibility is as follows:

- Project starts at 20 June 2018. The first stage is system study and Analysis is completed within 15-17 days approx. 1 July 2018.
- Next step is Understanding Existing System which will take 16-18 days approximately 3 July 2018.
- Then there are the steps like Problem Definition, Information Gathering and Database Design, Form Design, Coding, and Testing, Modification, Implementation and Documentation will take some duration.
- The total duration of the project is 9-10 months. That is from June 2018 to March 2019.
- This project is started from 15 June 2018 to March 2019.

3.5.2 UML DIAGRAM:

A. ENTITY RELATIONSHIP DIAGRAM:




The ER (Entity Relationship) Diagram represents the model of Managing Health Services System. The entity-relationship diagram shows all the visual instruments of database tables and the relation between Admin, Dietician, and User.

The main entities of the system are Dietician, Admin, and User.

ENTITIES AND THEIR ATTRIBUTES:

- Admin Entity-Attributes of admin are id and Password.
- Dietician Entity-Attributes of Dietician entity are Name, id, Phone No, Profile detail, Password, Diseased.
- User Entity: Name, age, height, weight, address, password, email_id, disease_id,foo_id.
- Food Type Entity: veg, Non-Veg.
- Disease: Name, _id, Range value, Display result.

Notation Used In Diagram:

Name	Symbol	Description
Entity		Rectangle Represents Entity Sets.
Weak entity		Double Rectangle Represents Weak Entity Sets
Attributes		Ellipse Represents Attributes.




Relationship		Diamond Represents Relationship Between two Entities.
Derived Attribute		Dotted Ellipse Represents Derived Attributes
Multivalued Attribute		Double Ellipse represents the Attribute which has multiple value.

Table 3.2Notation for ER diagram

- **Entities In Our System:**

Admin

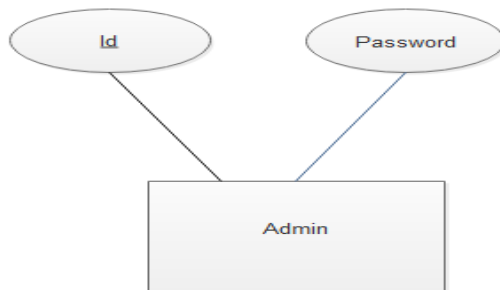


Fig 3.2 ER diagram for Admin

User

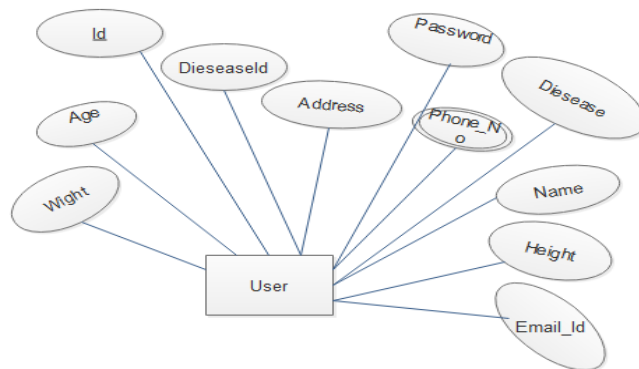


Fig 3.3 ER diagram for User

Dietician:

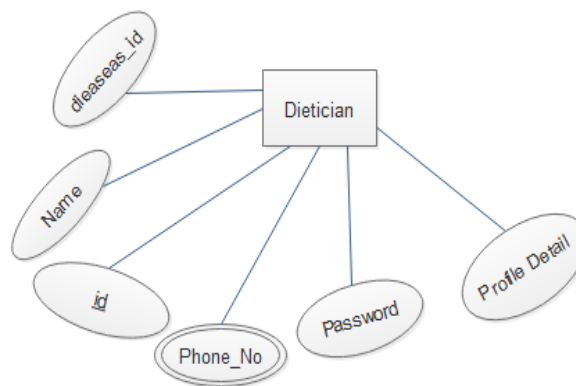


Fig3.4 ER for dietician

Food Type

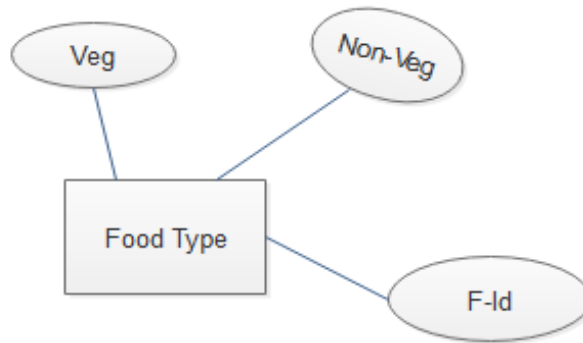


Fig3.5 ER for food type

Disease

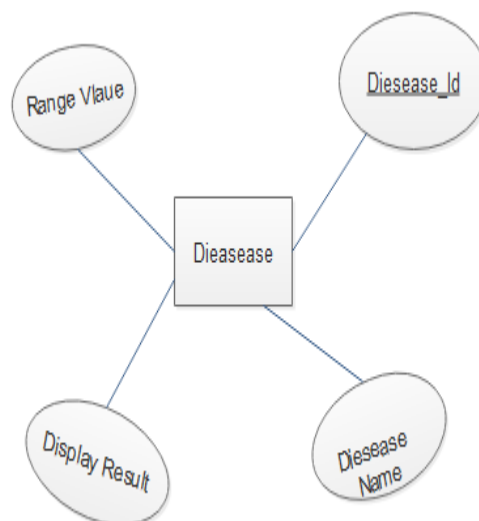


Fig 3.5 ER for Disease

ER Diagram

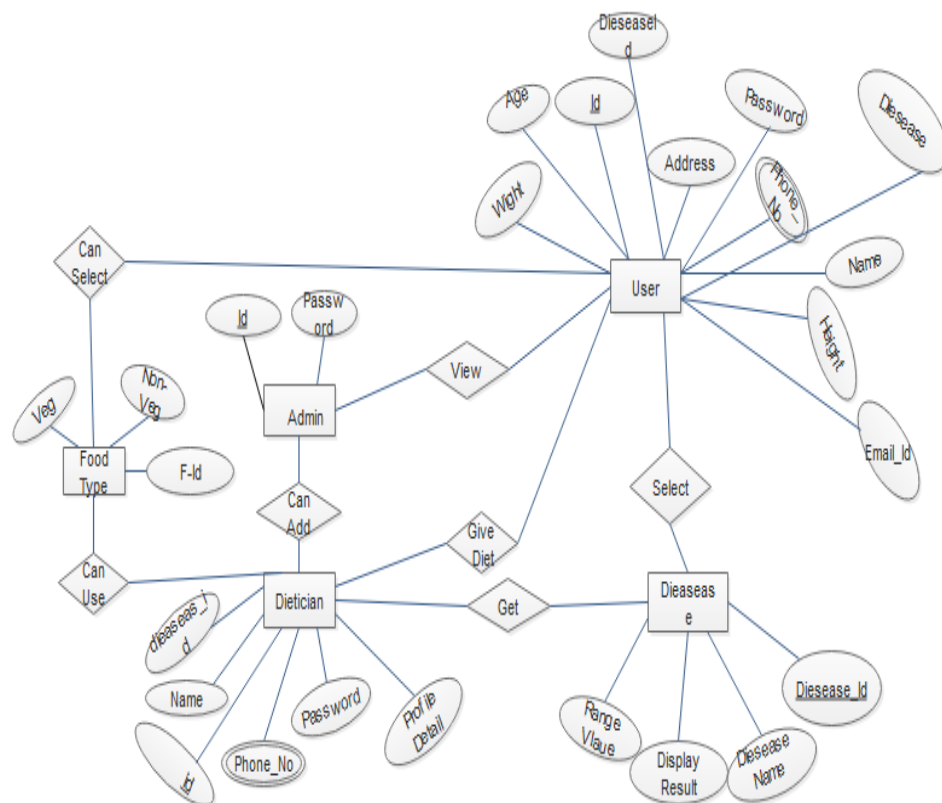


Fig 3.7ER diagram

B. DATA FLOW DIAGRAM(DFD):

A data flow diagram(DFD) is a graphical representation of the “flow” of data through an information system. It differs from the flowchart as it shows the data flow instead of the control flow of the program. A data flow diagram can also be used for the visualization of data processing.

A DFD shows all kind of data will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether process will operate in sequence or in parallel.

Notation:


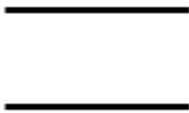
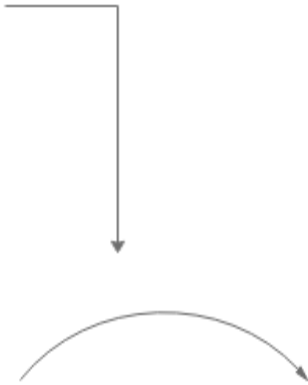
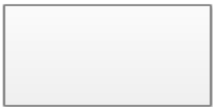
Name	Symbol	Description
Process		A Process transforms incoming data flow into outgoing data flow
Data store Notation		Data store are repositories of data in the system.
Dataflow Notation		Data flows are pipelines through which packets of information on flow. Label the arrows with the name of the data that moves through it.
External entity Notation		External entities are objects outside the system, communicates.

Table 3.3 Data flow diagram

0 Level DFD: A level 0 data flow diagram also known as a context diagram, shows a data system as a whole and emphasizes the way it interacts with external entities this DFD level 0 example shows how such a system might function within a typical retail business

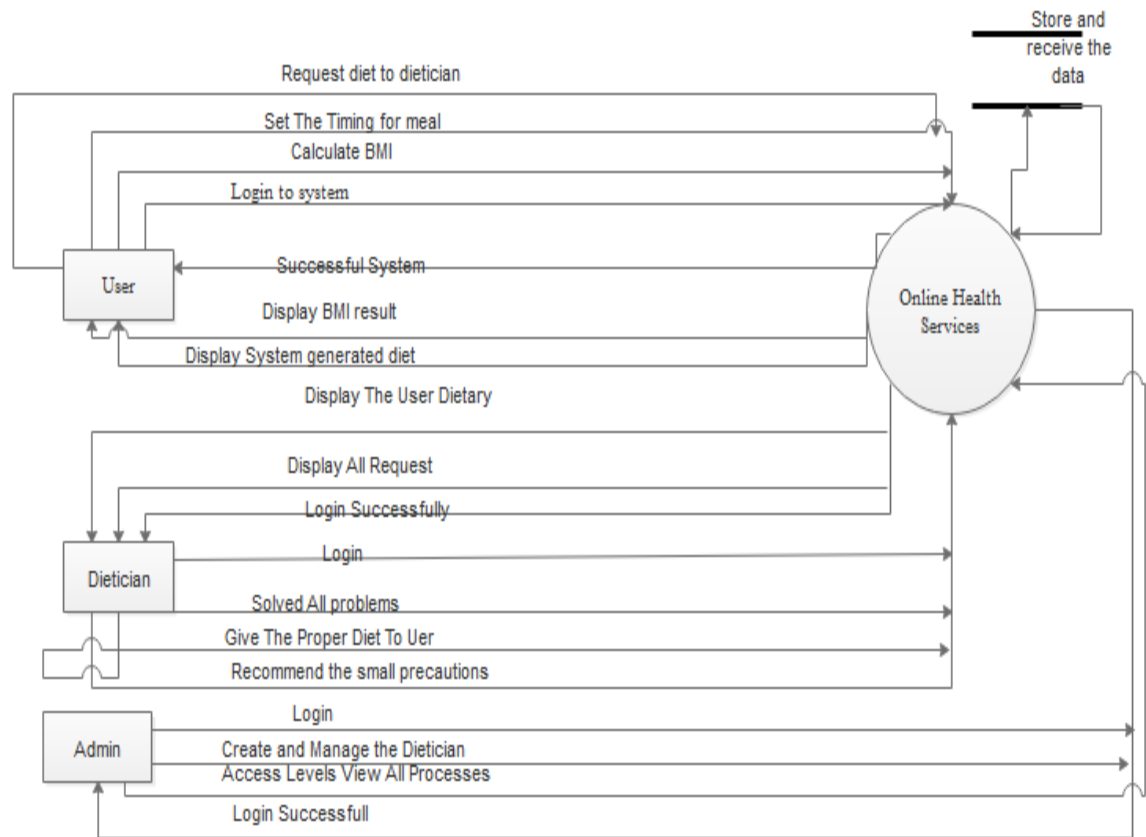


Fig 3.8 0Level DFD

Level 1 DFD:

The next stage is to create the level 1 DFD. This highlights the main functions carried out by the system. As a rule, we try to describe the system using between two and seven functions-two being a complicated system

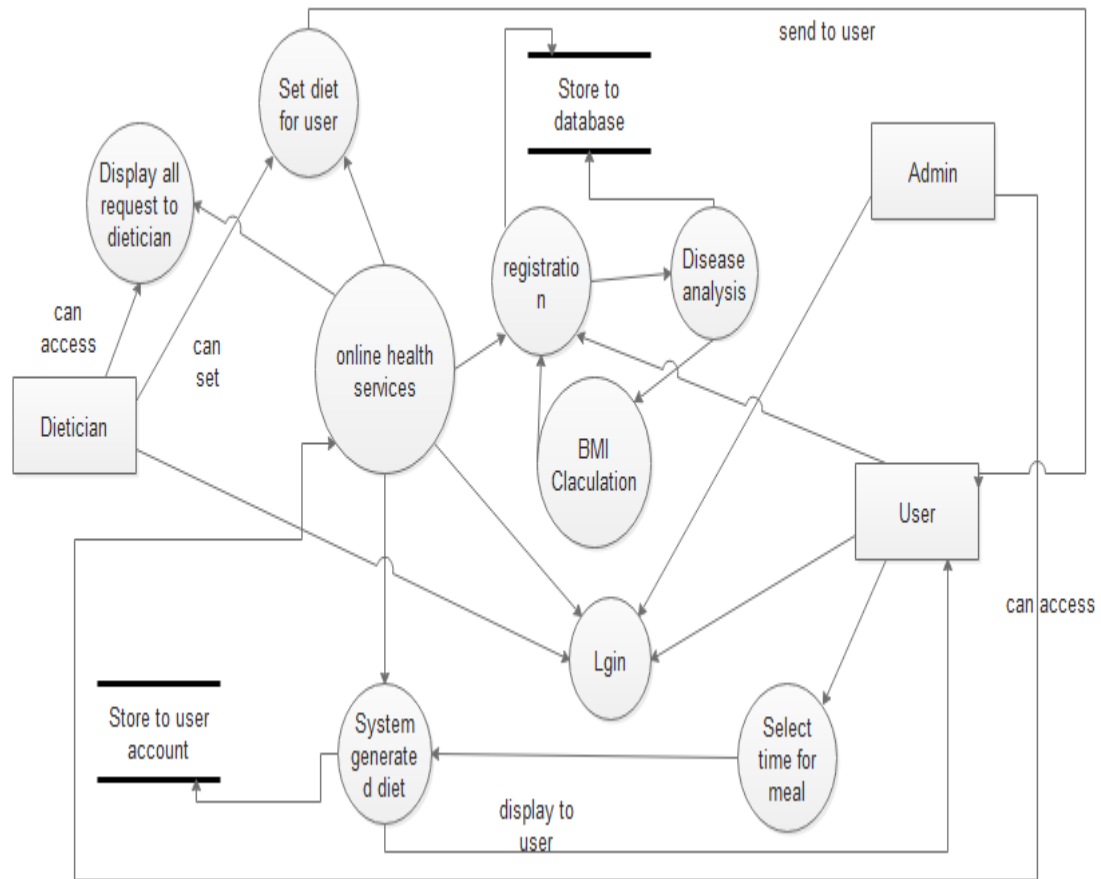


Fig 3.9 DFD level 1

Level 2:

A level 2 DFD offers a more detailed look at the processes that make up information system than a level 1 DFD does. It can be used to plan or record the specific make up of a system.

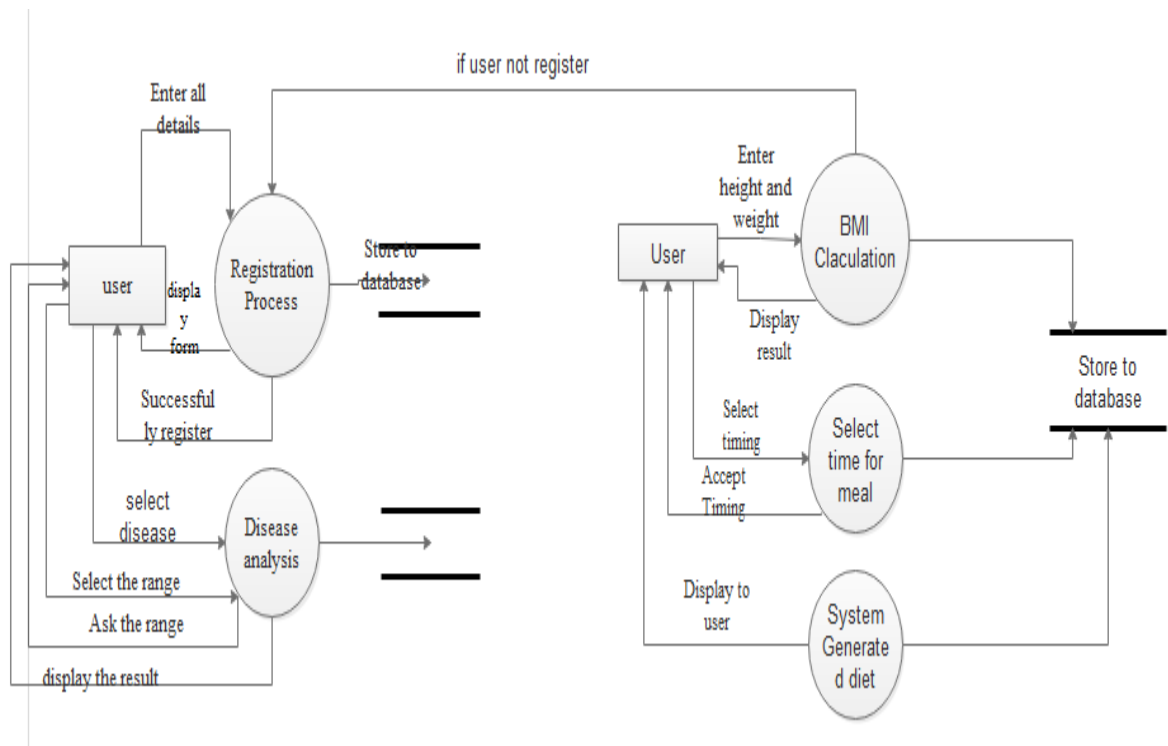
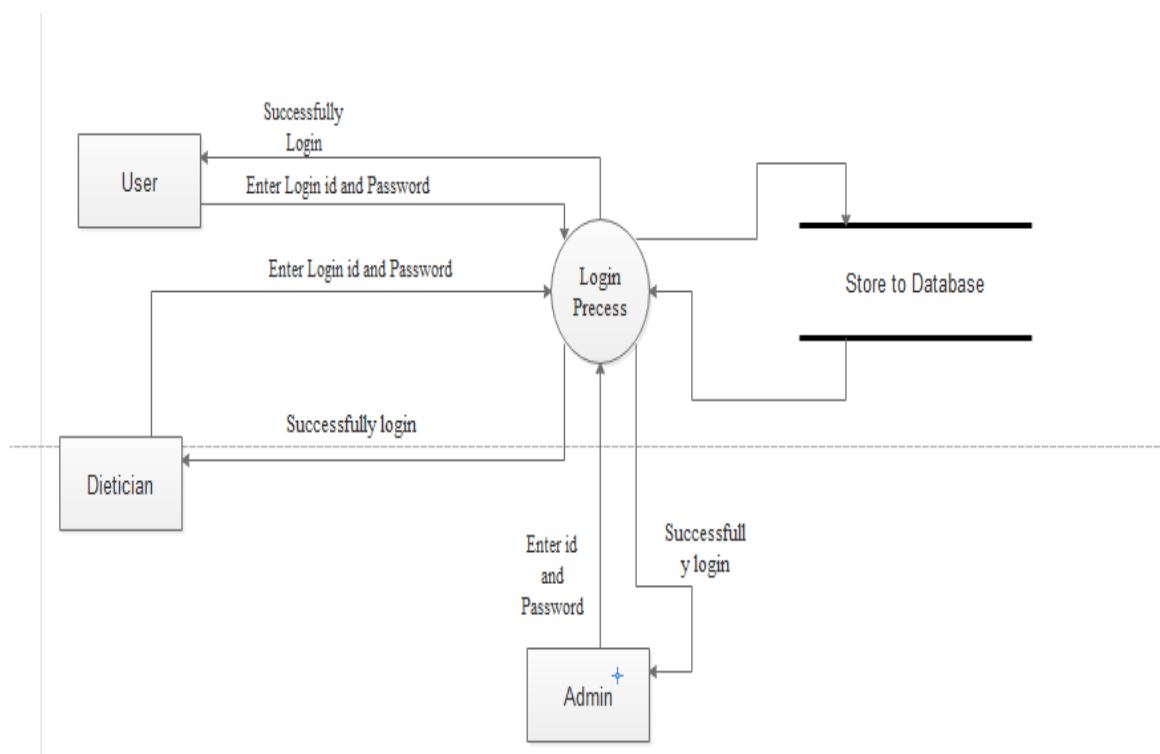


Fig3.10 DFD level2

Fig3.11 DFD level 2



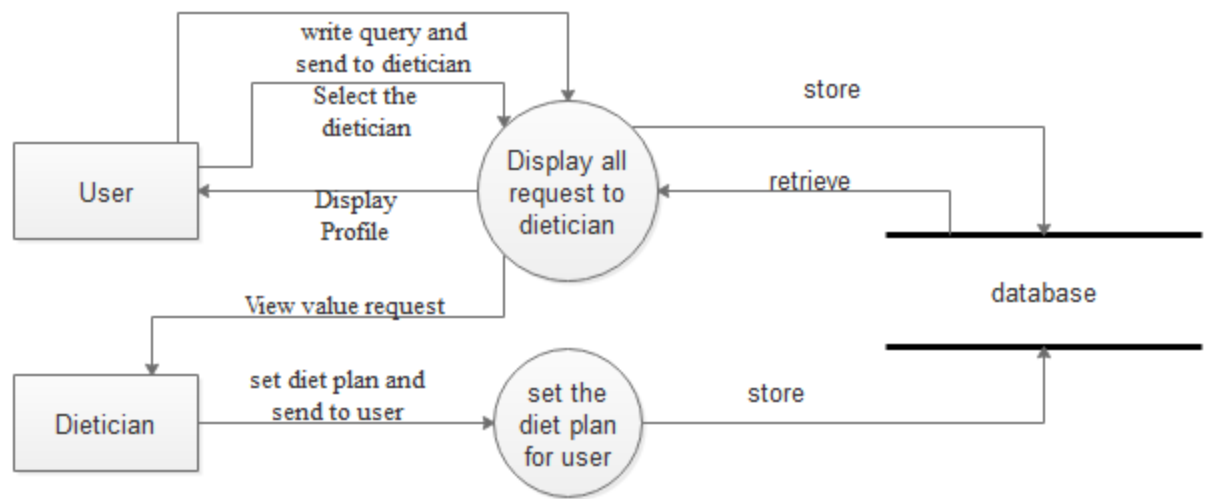


Fig3.12 DFD level 3

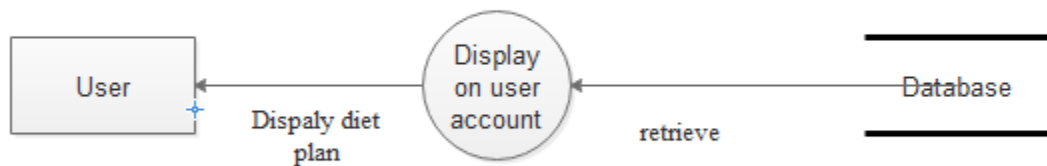


Fig3.13 DFD level 2

C. CLASS DIAGRAM:

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

For this system the class diagram is as shown in below.

The system contains the following classes:

USER class, Admin class, Dietician class, Food Type class, and Disease class.

- Class Diagram For Online Health System

Notation:






Sr.no	Name	Symbol	Description
1	Association		Represents the static relationship shared among the objects of two classes
2	Aggregation		It is an association that represents a part-whole or part-of relationship
3	Composition		
4	Generalization		Indicates that one of the two related classes is considered to be a specialized form of the other
5	Dependency		It is a weaker form of bond which indicated that one class depends on another because it uses it at some point in time

Table3.4 Notation for Class diagram

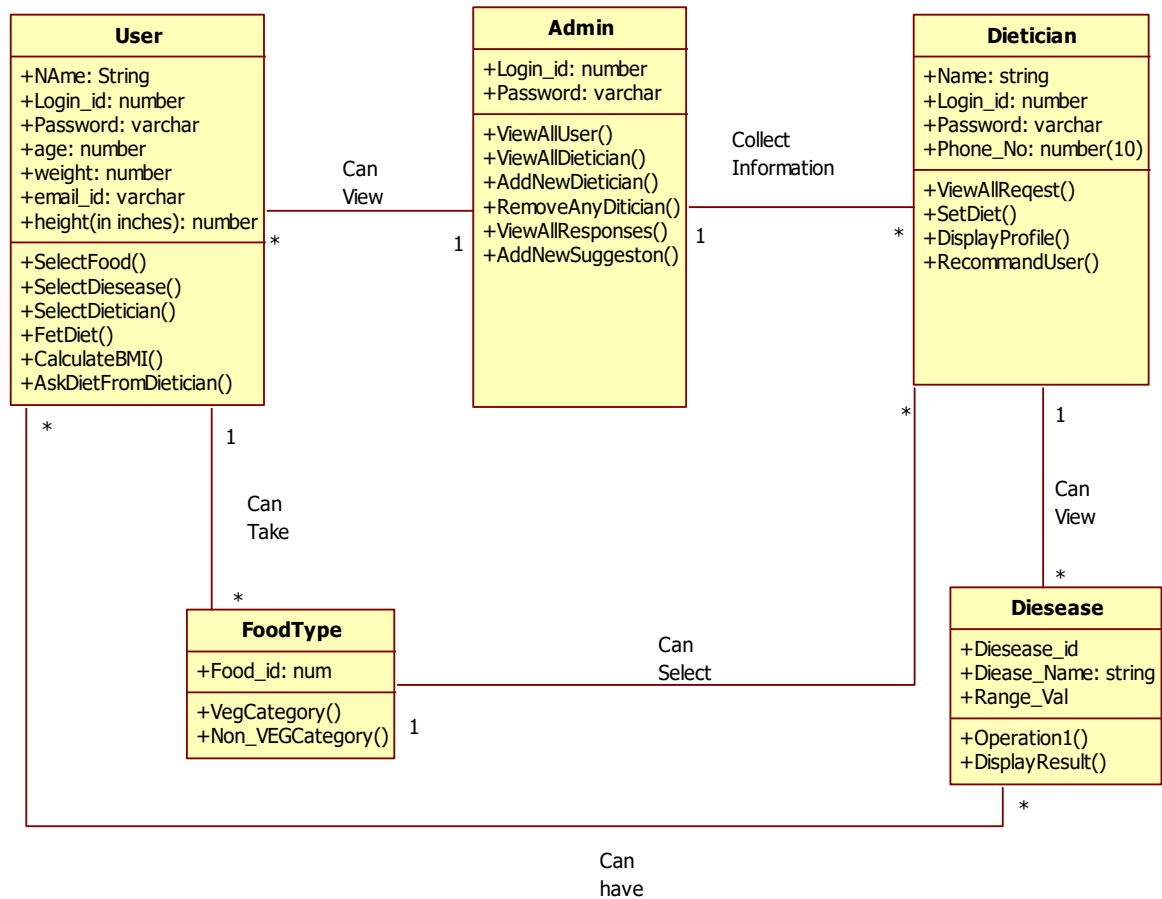


Fig3.13 Class Diagram

D. SEQUENCE DIAGRAM:

From the term Sequence, it is clear that the diagram is used to describe some type of Sequences among the different elements in the model. This Sequence is a part of dynamic behavior of the system.

This Sequence behavior is represented in UML by two diagrams known as **Sequence diagram**. The basic purpose of both the diagrams are similar.

Sequence diagram emphasizes on time sequence of messages. UML Sequence diagrams are interaction diagrams that detail how operations are carried out. As sequence diagrams can be used to capture the interaction between objects in the context of a collaboration, one of the primary uses of sequence diagrams is in the transition from requirements expressed as use cases to the next and more formal level of refinement. Use cases are often refined into one or more sequence diagrams.

Notation:



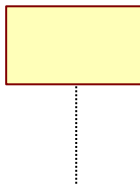
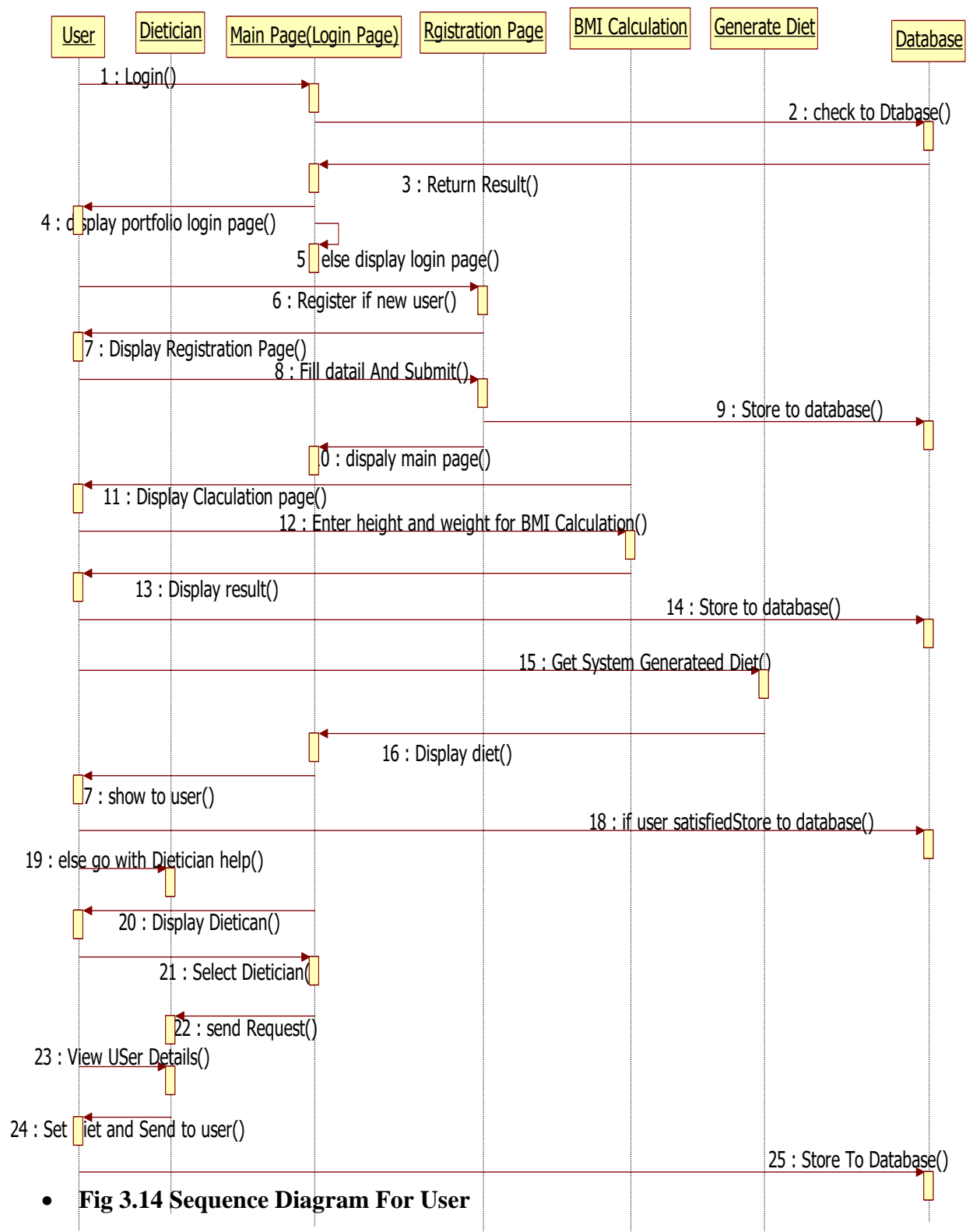
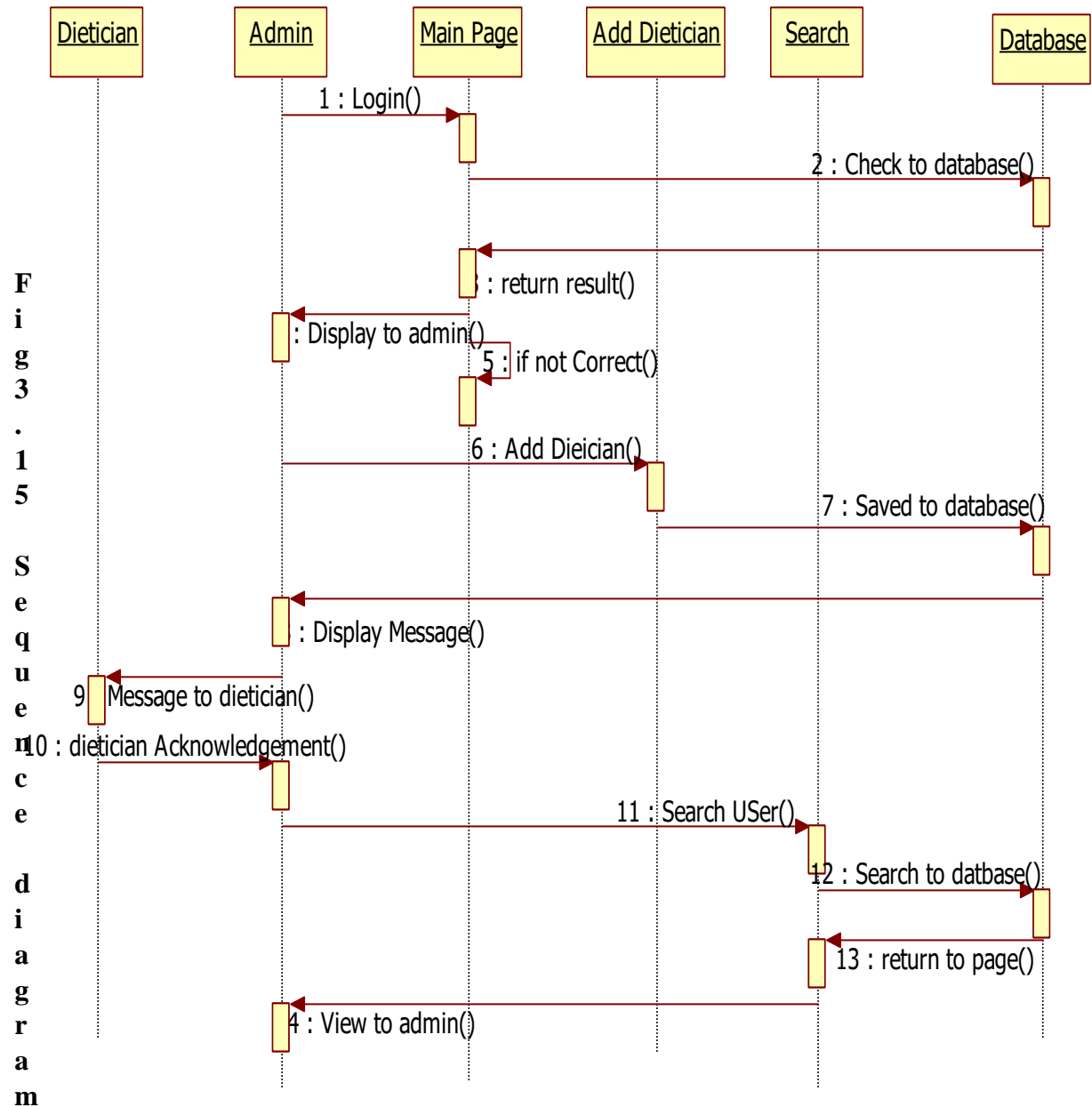
Name	Symbol	Description
Synchronous message		An instantaneous communication between objects that conveys information, with the expectation that an action will be initiated as result
Activation		The period during which an object is performing an action.
Object instance		An object that is created performs action, and/or is destroyed during the lifeline

Table3.5 Notation of Sequence Diagram

- **Sequence Diagram For Online Health System For user**



- **Sequence Diagram For Admin:**



for Admin

E. USE CASE DIAGRAM:

A use case is a list of steps, typically defining interactions between a role (known in UML as an “actor”) and a system, to achieve a goal. The actor can be human or external system.

Notation:


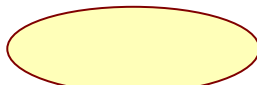
Name	Symbol	Description
Actor		Actor is represents a use or another system that will interact with the system you are modeling
Use Case		A use case is an external view of the system that represents some action the user might perform in order to complete a task

Table 3.6 Notations of Use case

1. Use case :Registration:

- Summary: To create the account of user must register themselves to log into the system.
- Actors: User.
- Precondition: Registration form should be available.
- Description: The registration form which contains the name, age, height, mail ID , address , user ID , password that needs to be filled by user.
- Exception: If the mandatory requirements won't filled by user then the application page cannot store to database and hence user cannot get login to system.

- Post condition: Account has been created successfully.

2. Use Case : Login:

- Summary: To registered user can login to system with the help of user ID and password.
- Actors: user, admin, dietician.
- Precondition: user must provide correct user ID and password.
- Description: user, admin and dietician login to system then system will check it if use ID and password is correct then system allows to enter to the system.
- Exception: user, admin and dietician can provide the wrong user ID and Password.
- Post condition: user, admin, and dietician get login to the system.

3. Use Case : BMI Calculation:

- Summary: user can calculate BMI to get their diet with the help of their height and weight.
- Actors: user, admin, dietician.
- Precondition: user must provide their height and weight.
- Description: On the basis of user height and weight they get their diet plan for that they must calculate their BMI.
- Exception: If the user provide wrong height and weight or either they do not provide height and weight they get wrong output or they get message that they should provide height and weight.
- Post condition: user get their appropriate result to their BMI.

4. Use Case: Set Time To Take The Meal:

- Summary: user set their time according to their need.
- Actors: User.
- Precondition: user should fill the time.

- Description: user should fill the time at which time they wants to take their breakfast, Lunch, Short snacks, and Dinner timings according to system will generate the diet for it.
 - Exception: user provide dissimilar time to take their diet plan.
 - Post condition: system accepts the timings.
5. Use Case: System Generated Diet:
- Summary: after setting the time user will get the system generated diet.
 - Actor: User,
 - Precondition: For getting the diet user should register themselves to the system.
 - Description: after the filling timing parameter user will get their diet after click on button.
 - Exception: if user had not register they cannot get diet plan.
 - Post condition: system generate it's diet plan.
6. Use Case: Dietician help
- Summary: if user satisfy with system generated diet they can move with dietician plan diet.
 - Actor: user, Dietician.
 - Precondition: system should have dietician and user should registered themselves with dietician.
 - Description: user get dietician planned diet plan with the help of dietician help module.
 - Exception: system do not have registered with dietician.
 - Post Condition: request send to right dietician.
7. Use Case: Dietician planned the diet
- Summary: dietician get message to plan the diet for user and dietician has to plan the diet for user.
 - Actor: Dietician.
 - Precondition: Dietician should register with system.
 - Description: Dietician will create diet plan for user according to their need and disease.

- Exception: user will not get any dietician help if dietician not registered with system.
 - Post condition: diet plan will send to user's account.
8. Use Case: user get their diet
- Summary: user will get the diet table on their registered account.
 - Actor: user.
 - Precondition: Dietician should send the diet plan to appropriate user account.
 - Description: Dietician will send the diet table to user then user get diet table to their account.
 - Exception: User won't get diet on time because of unavailability of dietician.
 - Post Condition: user get diet table.
- **Use Diagram:**

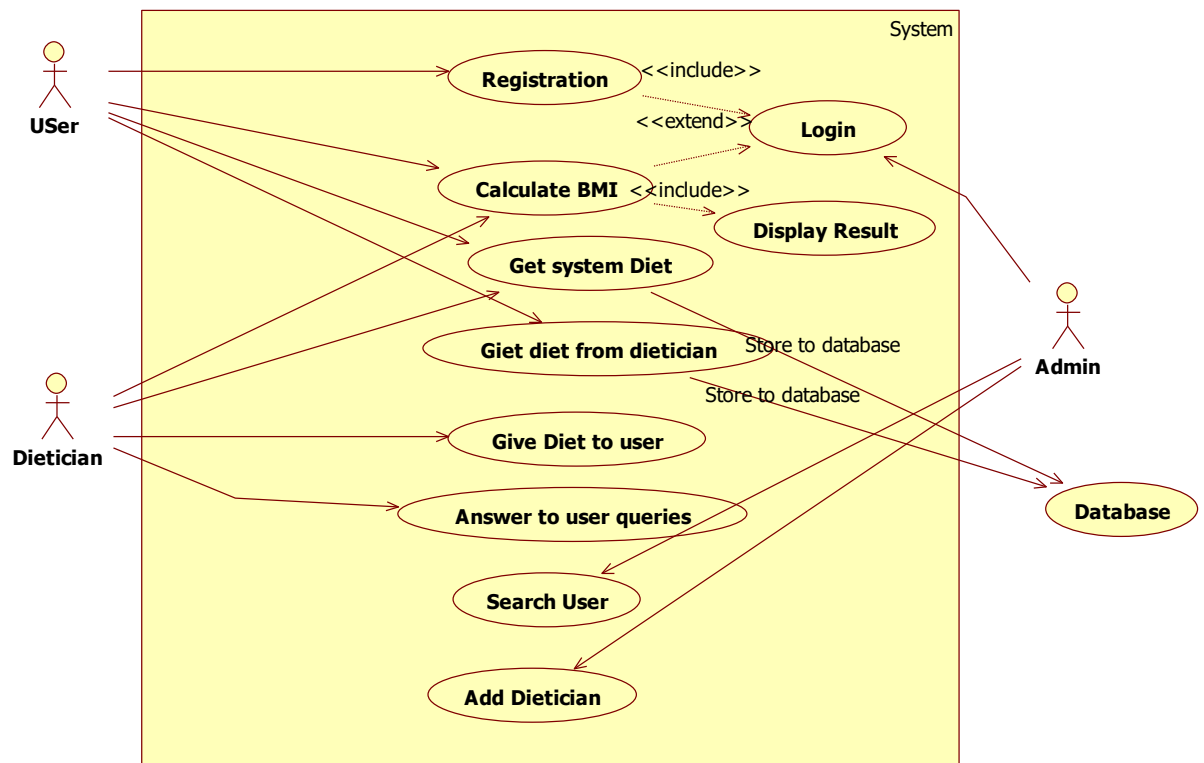


Fig3.16 Use case Diagram

F.ACTIVITY DIAGRAM:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

Notation:



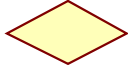


Name	Symbol	Description
Free-form transition link		A transition link represents control flow between nodes.
Rounded Rectangle		It represents the activity
Diamond		A logic where a decision is to be made is depicted by a diamond
Initial Activity		This shows the starting point or first activity of the flow
Final Activity		The end of the activity diagram, also called as a final activity.

Table 3.7Notations of Activity diagram

- Activity Diagram For Online Health System:

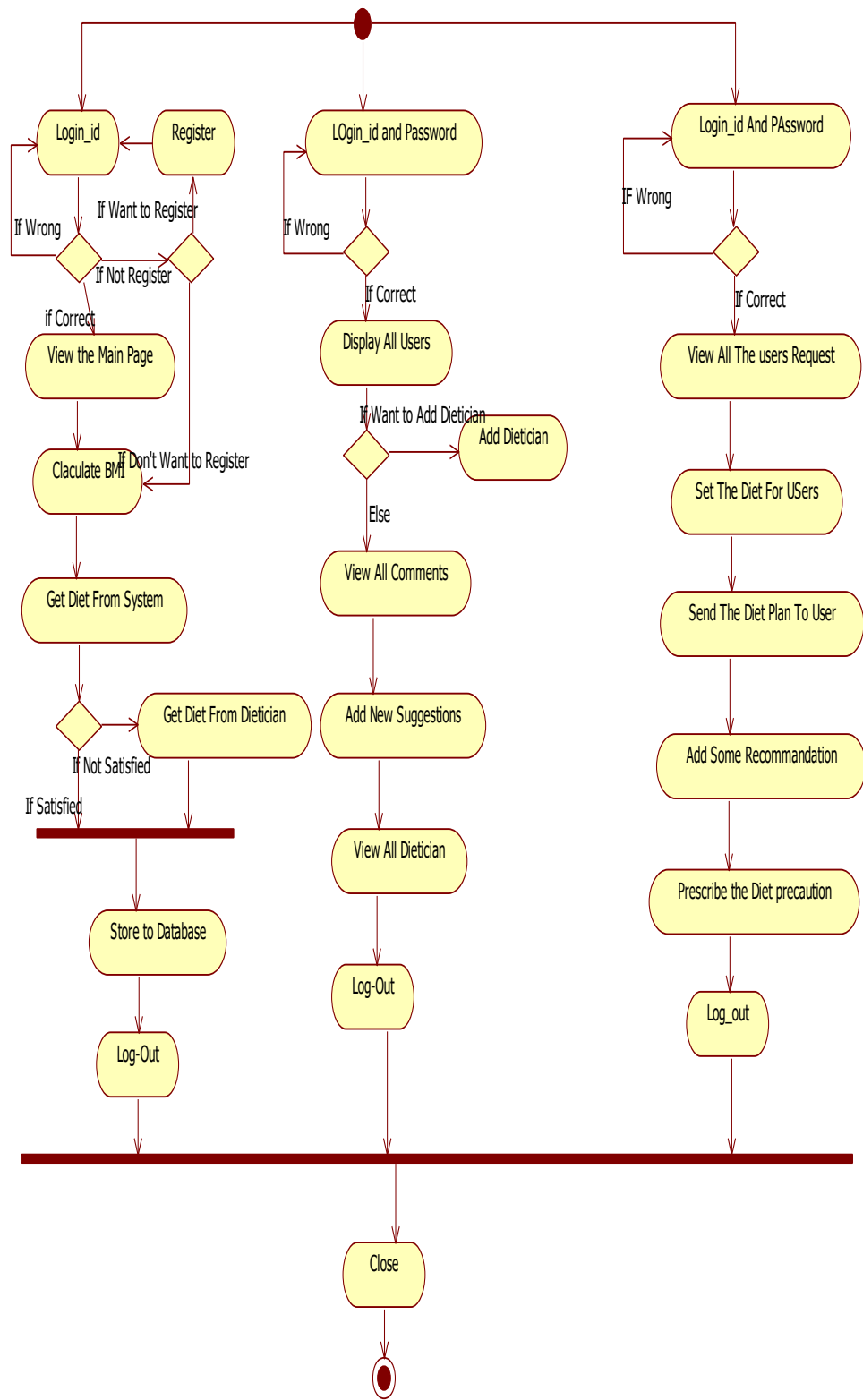


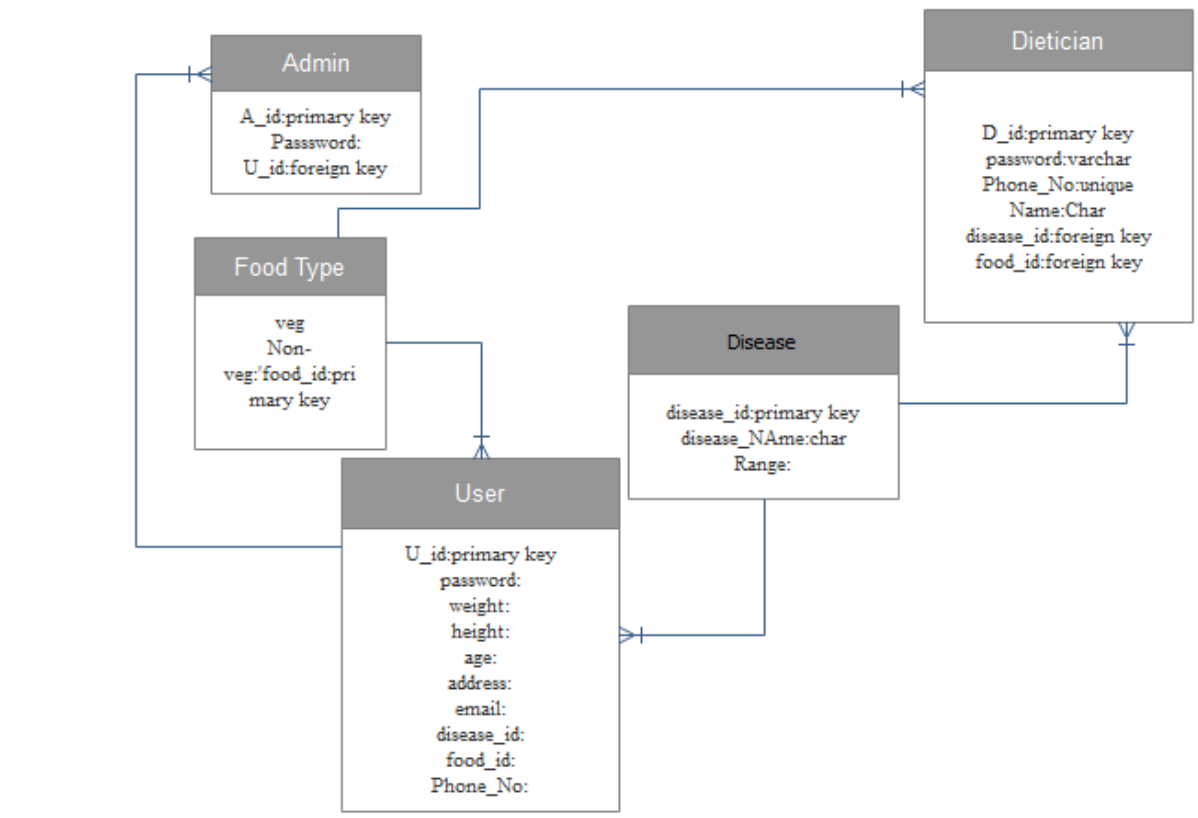
Fig 3.18 Activity Diagram

CHAPTER 4- SYSTEM DESIGN

4.1 DATA DESIGN

4.1.1 Schema Design:

A database schema diagram is skeleton structure that represents the logical view of the entire database. A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagram.



chem4.1 Schema diagram

4.1.2Data Integrity and Constraints

User

Attributes	Data Types	Constraints
Username	Varchar	Username must be 6 characters long.
Password	Varchar	Password must contain the at least 1 special symbol, 2numbers, and characters.
Address	String	
DiseaseId	Integer	Must be same as in the primary key
Age	Integer	
Weight	Integer	
Height	Integer	Should select the proper unit
Name	String	
FoodId	Varchar	Must be same as the described in food table

emailid	Varchar	Must follow the pattern
---------	---------	-------------------------

Table 4.1 User

Admin

Attribute	Data Type	Constraints
Admin_id	Varchar	It must be minimum 6 characters long
Password	Varchar	It must contain at least 1 special symbol, 2-3 numbers, and few characters.

Table 4.2 Admin

Dietician

Attributes	Data Types	Constraints
Name	String	
Dietician_id	Varchar	Must contain minimum 6 digit long
Password	Varchar	It must contain at least 1 symbol, 2-3 digits, and few characters
PhoneNo	Ineger(10)	Must contain 10 digits only

Disease_id	Varchar	Must be same as describe in table
profiledetails	Strings	
Food_id		Must be same as describe in table

Table 4.3 Dietician

Food

Attributes	Data Types	Constraints
Veg	String	
Non-veg	String	
Food_id	Varchar	It must be 6 digits long

Table 4.4 Food

Disease

Attributes	Dta Types	Constraints
Name	String	
Disease_id	Varchar	Must be 6 digit long
Range	Range	
Result display	String	

Table 4.5 Disease

4.3 User Interface:

Hand-drawn sketch of a Login page. The page has a title "Login" at the top. Below the title, there are two input fields: one labeled "Id" and one labeled "Password". Below these fields is a button labeled "Button".

Fig4.1 Login Page

Hand-drawn sketch of a Registration page. The page has a title "Registration Page" at the top. Below the title, there are several input fields and labels: "Name:", "Phone Number:", "Email Id:", "Age:", "Gender: 0 Male 0 Female", "Id:", "Password:", "Confirm Password:", and "Food TYPE:". At the bottom is a button labeled "Register".

Fig 4.2 Registration page

Disease Selection

☐ Hypertension

Range: ▼

☐ Diabetes

Range: ▼

☐ Thyroid

Range: ▼

Fig 4.3 Disease Selection

The Art of being Healthy

[My Details](#)
[BMI Calculator](#)
[Dietchart](#)
[DietPlan](#)
[Logout](#)

BMI Calculator

Weight :-

Lbs (10 kg = 22 lbs)

Height :-

(Inch)

Your BMI is :
 You are : [Under/Normal/over] weight

Fig 4.4 BMI calculation page

Breakfast at - 8:00 AM

Item	Protein	Carbohydrate

Lunch at - 01:30 PM

Item	Protein	Carbohydrates

Snacks at - 5:30 PM

Item	Protein	Carbohydrate

Dinner at - 9:00 PM

Item	Protein	Carbohydrate

Fig 4.6 System generated diet page

Fig 4.6 System generated diet page

Add Dietician View Users BMICalculator viewchart Logout

Add Dietician

Name:

Contact Number:

Email Id:

Gender: ☐ Male ☐ Female

Fig 4.7 Admin view page

View Requests View Users Logout

Create Diet Plan

Disease : None Food Type : Veg

Breakfast	Timing: <input type="text"/> ▼	<input type="text"/>
Lunch	Timing: <input type="text"/> ▼	<input type="text"/>
Evening Snacks	Timing: <input type="text"/> ▼	<input type="text"/>
Dinner	Timing: <input type="text"/> ▼	<input type="text"/>

Fig 4.8 Dietician generated page:

Annexures:

Table of Content:

Table No.	Table name	Page No.
3.1	Schedule Analysis	3
3.2	Notation of ER	12
3.3	Notation of DFD	17
3.4	Notation of Class	23
3.5	Notation of Sequence	25
3.6	Notation of Use case	29
3.7	Notation of Activity	31
4.1	User	32
4.2	Admin	32
4.3	Dietician	33
4.4	Food Type	33
4.5	Disease	33

Table of Figures

Figure No	Figure Name	Page No.
2.1	Microsoft ASP.Net	3
2.2	Microsoft SQL server	4
3.1	Gantt chart	8
3.2	ER Diagram for Admin	13
3.3	ER for user	14
3.4	ER for dietician	15
3.5	ER for Food type	15
3.6	ER for disease	16
3.7	DFD Level 0	18
3.8	DFD Level 1	19
3.9	DFD Level 2 for registration and BMI calculation	20
3.10	DFD level 2 for login	21
3.11	DFD level 2	22
3.12	DFD level 2	23
3.13	Class diagram	24
3.14	Sequence for user	26
3.15	Sequence for admin	27
3.16	Use case	30
3.17	Activity	32
4.1	Schema	33
4.2	Login page	34
4.3	Registration page	36
4.4	Disease selection page	37
4.5	BMI calculation page	38
	System generated diet	39
4.7	Admin View page	39
4.8	Dietician set diet	39

Bibliography:

A Book References:

The book referred from 10/06/2018 to 25/03/2019 are:

- Software Engineering, “Ian Somerville”, 8th edition, Pearson education.

- Object Oriented Modeling and Design with UML, Michael Blaha & James Rumbaugh, Prentice Hall of India, 2nd Edition
- Database System Concepts | Henry F. Korth, Abraham Silberschatz, S. Sudarshan | McGraw-Hill 4th Edition

B web References:

- www.w3schools.com
- www.TutorialsPoint.com
- www.wikipedia.com
- www.creately.com