

PROJECT REPORT
ON
INTEGRATED WORKFORCE
MANAGEMENT SYSTEM

By

Krish Jain (23012250410088)
Samkit Jain (23012250410089)
Mann Makhecha (23012250410124)
Chirag Gupta (23012250410079)
Het Malvaniya (23012250410133)
Shaan Mansuri (23012250410137)
Dhiraj Jagwani (23012250410087)



DEPARTMENT OF INFORMATION TECHNOLOGY

L.J. POLYTECHNIC, AHMEDABAD

2022-2023

DEPARTMENT OF COMPUTER ENGINEERING
LJ POLYTECHNIC, AHMEDABAD
2024-2025

CERTIFICATE

Date: __/__/__

This is to certify that **Mr. KRISH JAIN, Mr. SAMKIT JAIN, Mr. MANN MAKHECHA, Mr. CHIRAG GUPTA, Mr. HET MALVANIYA, Mr. DHIRAJ JAGWANI, Mr. SHAAN MANSURI** from LJ POLYTECHNIC having Enrollment No.**23012250410088,23012250410089, 23012250410124, 23012250410079, 23012250410133, 23012250410087, 23012250410137** have completed project documentation and partial development on the problem definition of semester IV during the academic year 2024-25 having Title **INTEGRATED WORKFORCE MANAGEMENT SYSTEM (SPHATIK)** in a group consisting of 7 persons.

Institute Guide

Head of The Department

ACKNOWLEDGEMENT

We are very thankful to our faculty member for giving us this wonderful opportunity to this wonderful project on the topic (INTEGRATED WORKFORCE MANAGEMENT SYSTEM).

We are highly thankful to our faculty member for the guidance and constant supervision as well as for providing necessary information regarding the project & due to that we came to know about so many new things.

We would also like to express our gratitude towards our parents & friends for their kind co-operation and encouragement which helped us in completion of the project.

Our thanks and appreciations also go to our colleagues in developing the project and people who have willingly helped us with their abilities.

Krish Jain (23012250410088)

Samkit Jain (23012250410089)

Mann Makhecha (23012250410124)

Chirag Gupta (23012250410079)

Shaan Mansuri (23012250410137)

Dhiraj Jagwani (23012250410087)

Het Malvaniya (23012250410133)

Table of Contents

ABSTRACT.....	V
Chapter 1 Introduction.....	1
1.1 Need of the system.....	1
1.2 Detailed Problem Definition.....	1
1.3 Viability of the System	2
1.4 Presently Available System for the Same	2
1.5 Future Prospect.....	2
Chapter 2 Analysis.....	3
2.1 Requirement Analysis.....	3
2.2 Project Model.....	3
2.3 Schedule Representation.....	4
2.4 Feasibility Study.....	5
Chapter 3 Design.....	6
3.1 Data Flow Diagram.....	6
3.2 E.R. Diagram.....	11
Chapter 4 System Modeling.....	15
4.1 Database Design.....	15
Chapter 5 Technical Specification.....	22
5.1 Hardware Specification.....	22
5.1.1 RAM.....	22

5.1.2 Hardware Storage Needed.....	22
5.2 Platform	22
5.2.1 Supporting Operating System.....	22
5.2.2 Programming Server.....	22
5.3 Framework.....	22
5.3.1 Markup Language.....	22
5.3.2 Programming Language.....	22
5.3.3 Scripting Language.....	22
5.4 Technical Specification	22
5.4.1 Frontend	22
5.4.2 Backend.....	22
5.4.3 IDE	22
5.4.4 UML Tools.....	22
5.4.5 SRS Tools.....	22
CONCLUSION.....	23
BIBLIOGRAPHY.....	24

Table Index

1. Schedule Representation.....	4
2. DFD symbol.....	6
3. ER symbol.....	12
4. Service_booking.....	15
5. User.....	16
6. Service.....	17
7. Member.....	18
8. Courses.....	19
9. Freelancers.....	20
10. Freelancers_Contact.....	21

Figure Index

1. Prototype Model.....	3
2. DFD Level 0.....	7
3. DFD user level 1.....	8
4. DFD member level 1.....	9
5. DFD admin level 1.....	10
6. ER Diagram.....	14

ABSTRACT

The Integrated Workforce Management System is a comprehensive platform designed to streamline recruitment, skill development, and freelance task management. It integrates features inspired by platforms like Zomato and Swiggy, enabling smart task allocation based on user location and availability. The system connects job seekers, freelancers, and employers, catering to a wide range of industries, including IT roles (web development, software development) and local services (Web Development, Software Engineering etc.).

1.1 Need for the system

- The need for an integrated workflow management system like Sphatik arises from the increasing demand for a seamless, centralized platform that can efficiently manage multiple services, stakeholders, and roles.
- By offering a unified solution that brings together various services (from delivery partner assignments to freelancer networks and local services), Sphatik addresses several pressing challenges faced by both businesses and individuals in India.


1.2 Detailed problem definition

- There will be detailed description regarding product.
- Admin can easily manage all services.
- In this system we have developed facility to customer can by all services, product.
- User can take order anytime and anywhere.

1.3 Visibility of the system

- The system can easily understand and use it.
- Customer review and rating section are also available.
- This system is user friendly.

1.4 Presently available system for same

- This system is focus on your config, ordering all component from the one source may not be best solution.
- This system is based cloud based.
- This website has all sort of electronic product.
- Examples:  <https://www.zomato.com/>



<https://www.coursera.org/>

1.5 Future prospect

- Advertising of this system
- Better security
- Provide the payment system for pay your amount in 7 days.
- Provide a best quality system in future.

CHAPTER 2

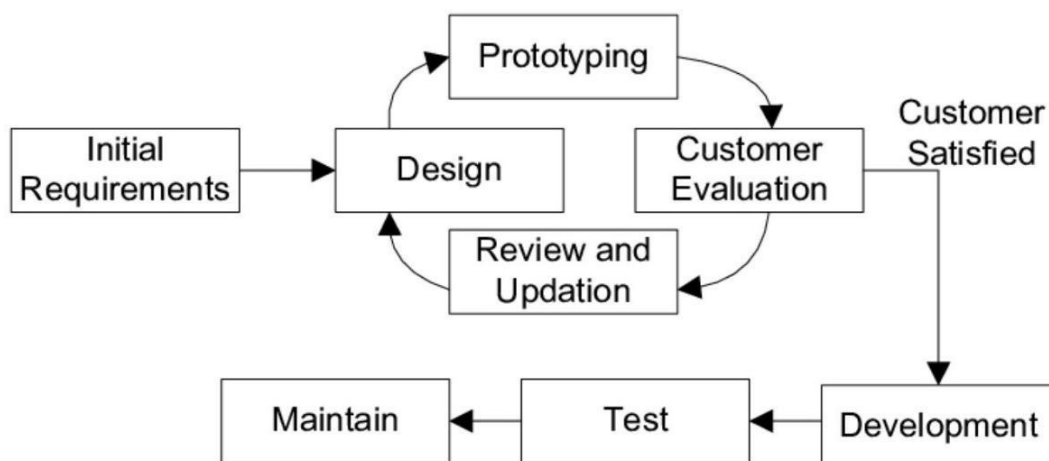
ANALYSIS

2.1 Requirement analysis

- Customer can take service and write review in review box.
- Here we provide home delivery, cash on deliver and UPI services.
- User can upload a photo on login page.

2.2 Project model

- Integrates recruitment, skill development, and freelance task management.
- Similar to Zomato and Swiggy, it assigns tasks based on user location and availability.



[Figure 1 Prototype model]

2.3 Schedule representation

- Generalized project scheduling tools and technique can be applied with little modification to software projects.
- Program evolution and review techniques (PERT) and critical path method (CPM) are two project scheduling method that can be applied to software development. Both techniques are Driven by information already developed in earlier project planning activities:
 - Estimate of effort.
 - A decomposition of the product function.
 - The selection of appropriate process model and task set.
 - Decomposition of tasks.

[Table 1: Schedule representation]

Activity	Start date	End date
Requirement analysis	16/01/25	30/01/25
System analysis	31/01/25	11/02/25
System design	12/02/25	22/02/25
System coding	23/02/25	12/03/25
Testing and integration	13/03/25	15/03/25


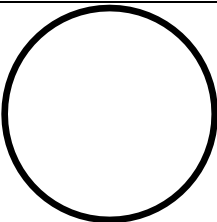
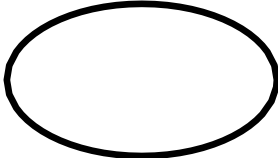
2.4 Feasibility study



- Cost: software should be cost effective as per its usage.
- Maintainability: software maintain can be easy.
- Understandability: software should be easy to understand or efficient.
- Portability: software should be capable to adapted all environment.

3.1 Data flow diagram

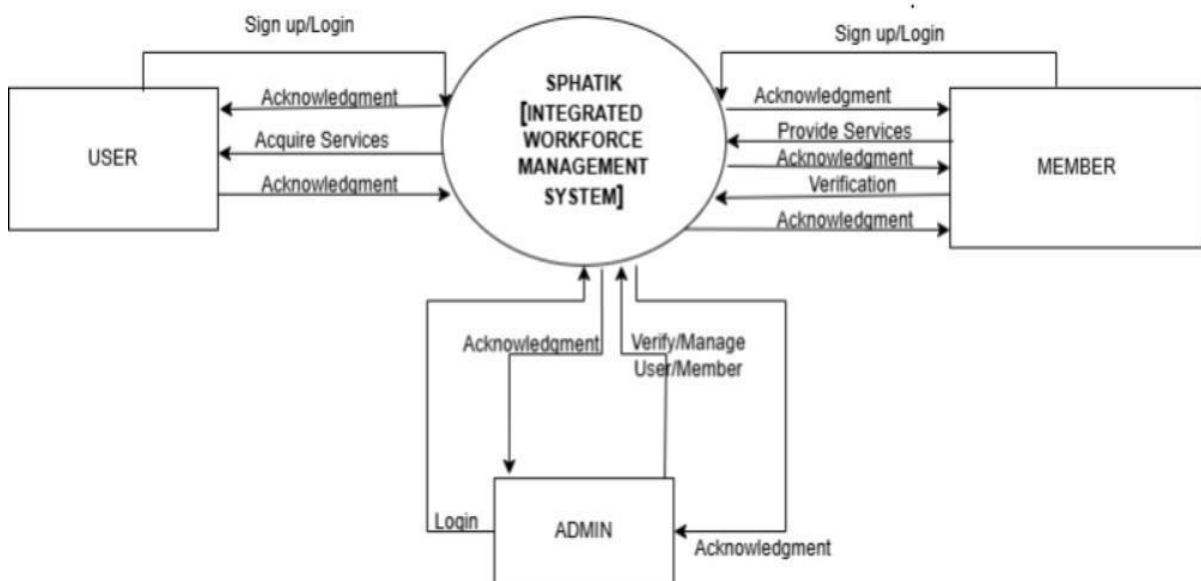
- DFD (data flow diagram) is also known as bubble chart or data flow graph. DFD's are very useful in understanding the system and can be effectively used during analysis. It shows flow of data through a system visually.
- The DFD is a hierarchical graphical model of a system the different processing activities or functions that the system performs and the data interchange among these functions.

[Table 2: DFD symbol]

Symbol	Description
	Entity: Entities are external to the which interacts by inputting the data.
	System: It shows the system name.
	Process: It shows the part of the system that transforms into outputs.

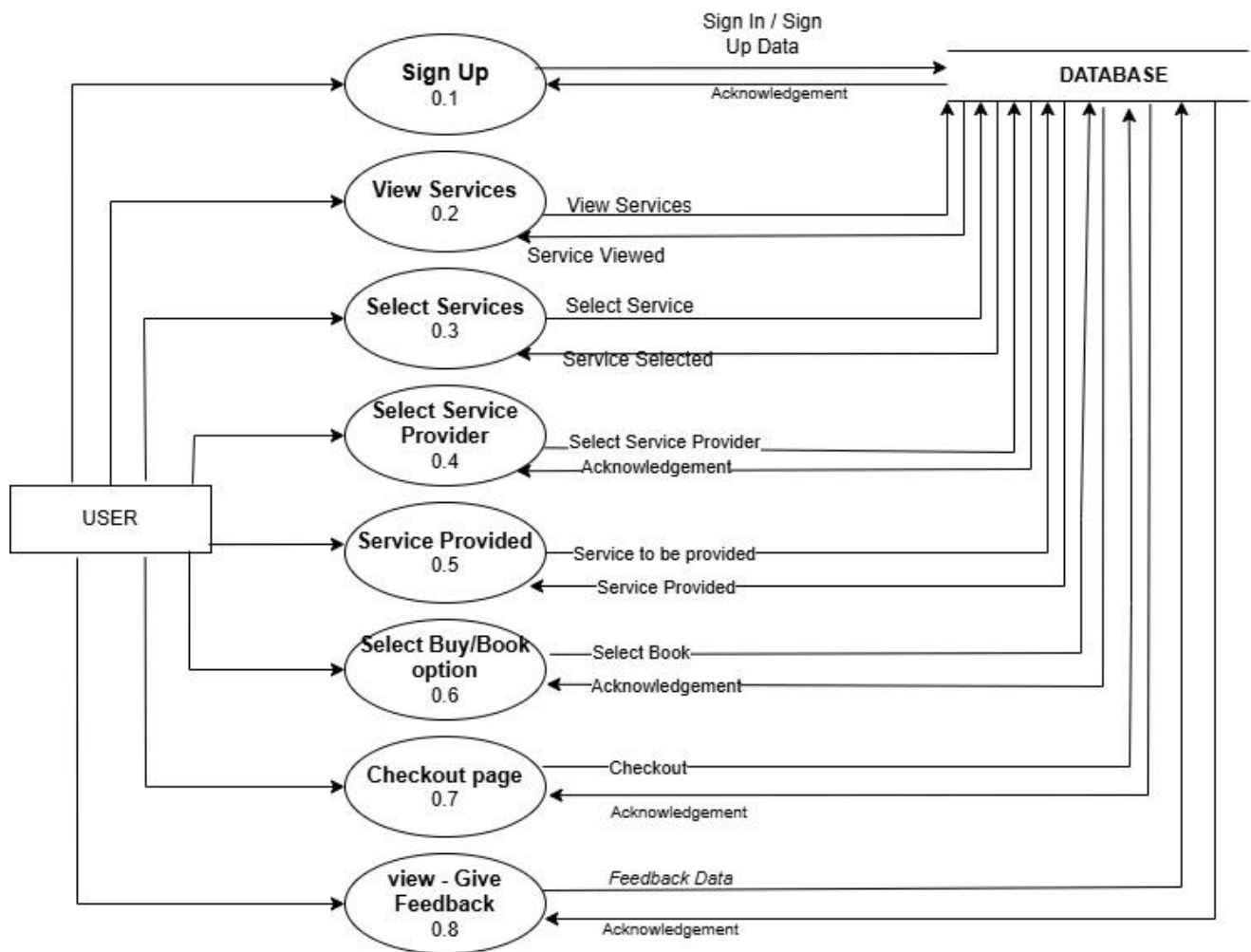
	Data Flow: It passes the data from one part to another.
	Data Store: Data store is represented by two parallel lines. It is generally database.

➤ **DFD context level (level 0)**



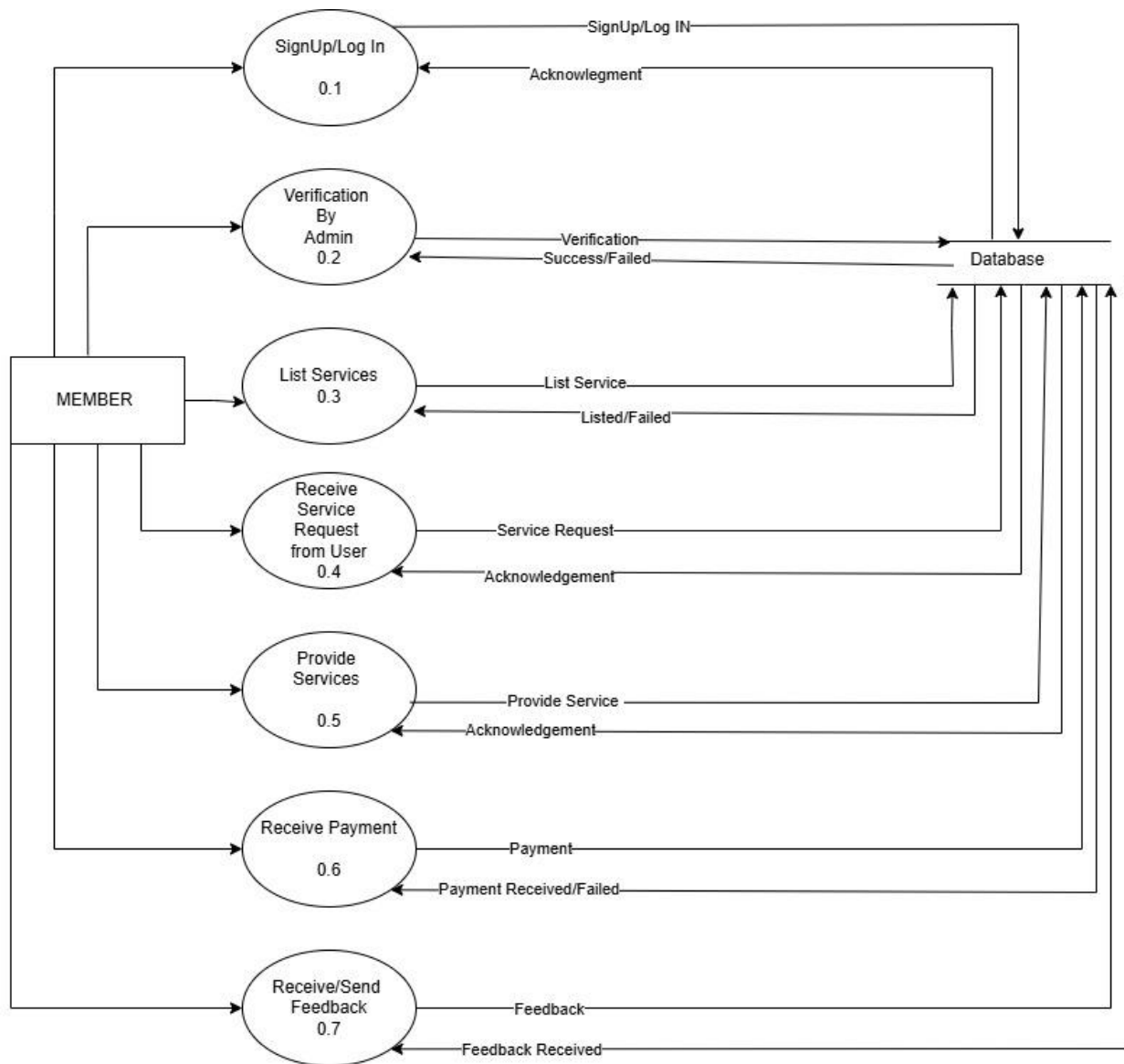
[Figure 2: DFD level 0]

➤ DFD user level 1



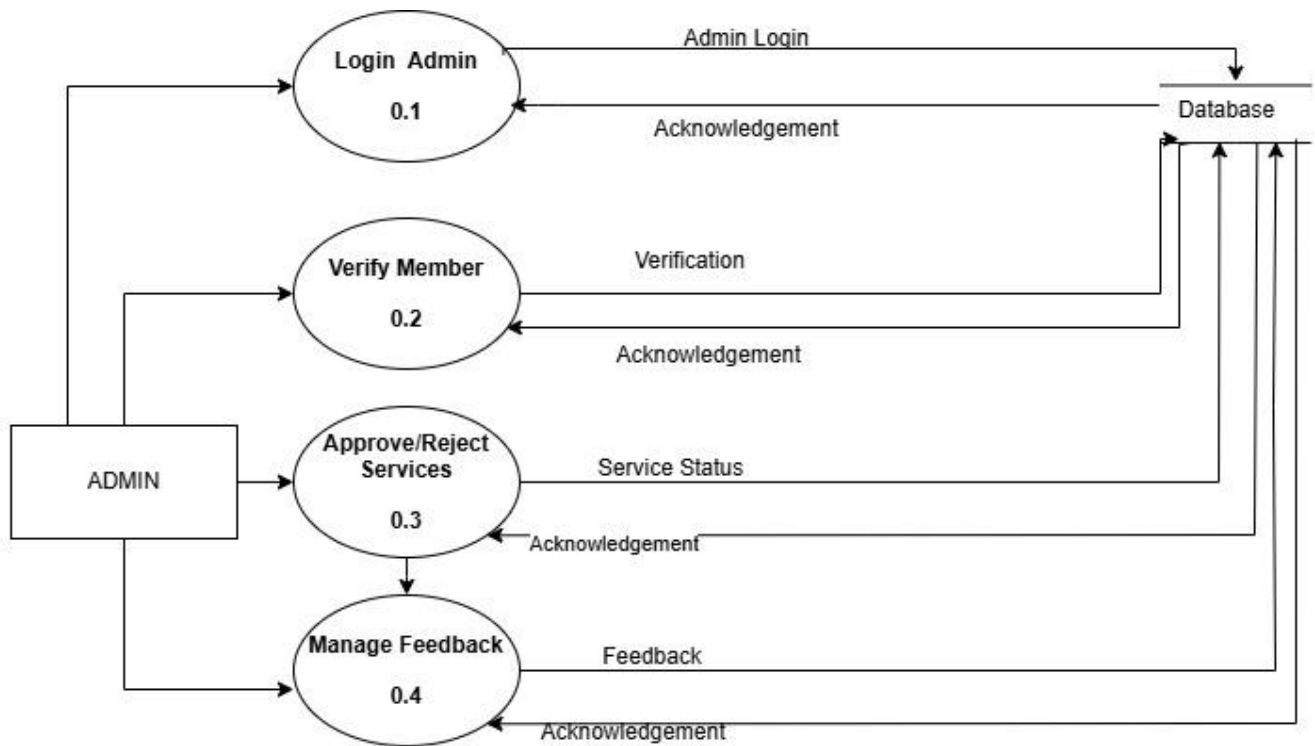
[Figure 3: DFD user level 1]

➤ DFD member level 1



[Figure 4: DFD member level 1]

➤ DFD Admin level 1


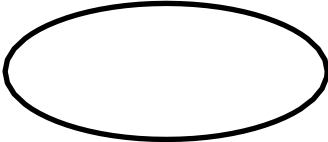
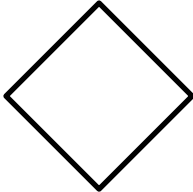
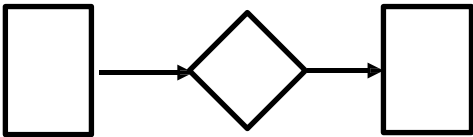


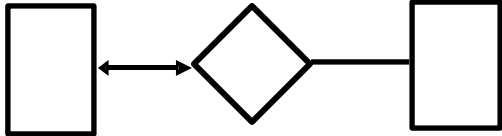
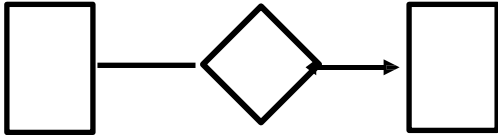
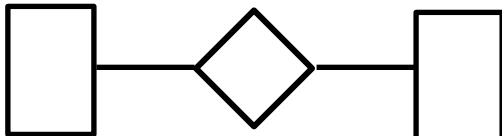
[Figure 5: DFD admin level 1]

3.2. E-R Diagram

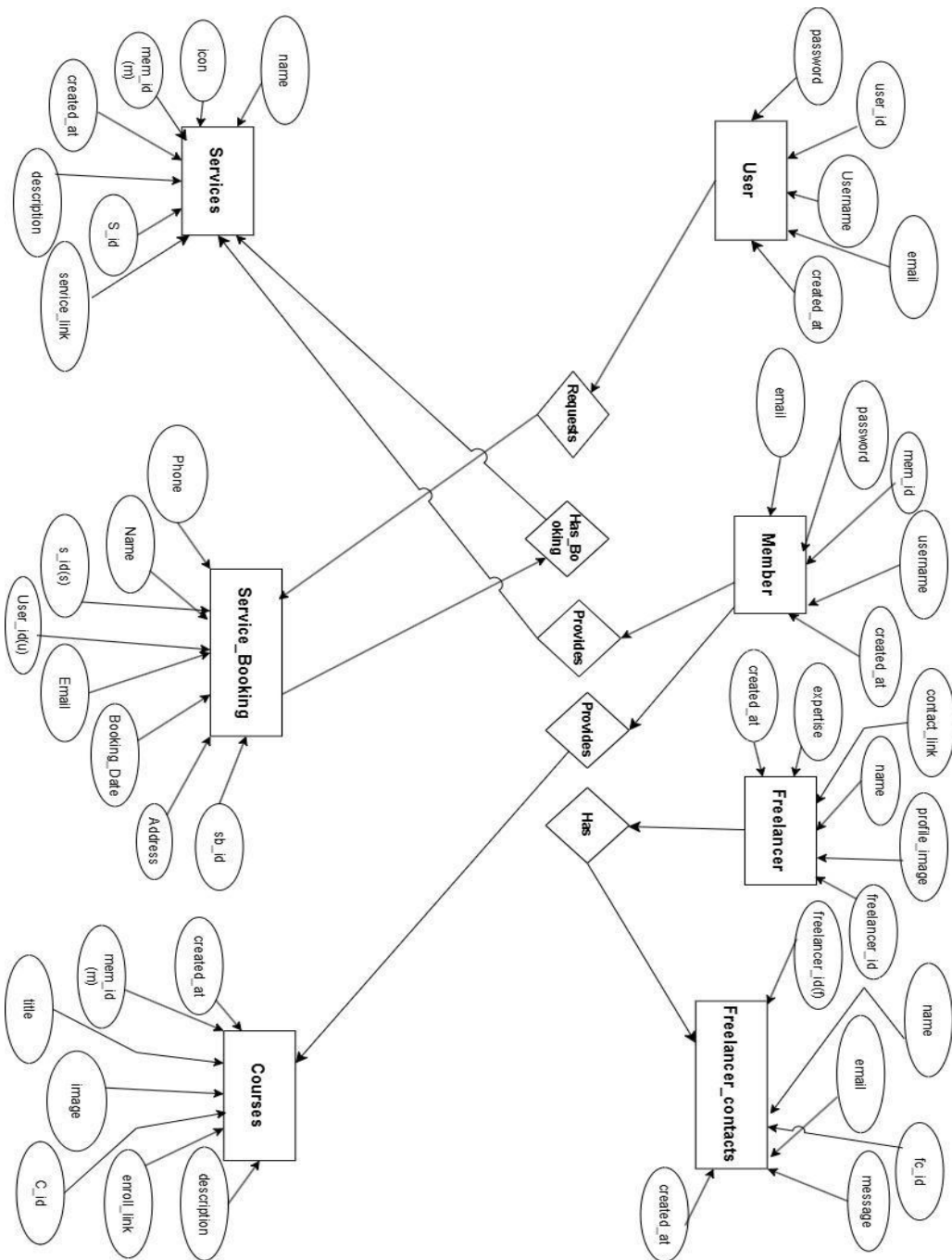
- An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how "entities" such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education, and research.
- Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships, and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

[Table 3:ER symbol]

Symbols	Description
	Entity: Data object is real world entity or thing. It is represented by a rectangle shape. An entity is an object or concept about which you want to store information.
	Attributes: An attribute is property of characteristic of an entity. It is represented by oval shape.
	Relationship: Entity are connected each other via relations. Generally, relationships in binary because there are two entities are related to each other.
	Cardinality (One to One): A instance of entity A can relate to one instances of entity B

	<p>Cardinality (One to Many): An instance of entity A can relate to one or many instances of B but we can only relate one instance of A.</p>
	<p>Cardinality (Many to One): One or more instances of entity A can relate to one instances of B.</p>
	<p>Cardinality (Many to Many): One or more instances of entity A can relate to one more instance of entity B.</p>

➤ E-R Diagram



[Figure 6:ER diagram]

CHAPTER 4

SYSTEM MODELING

Table name: service_booking

Primary key: sb_id

Foreign key: user_id, s_id

[Table 4:service_booking]

Field name	Data type	Constraints	Description
sb_id	Int(11)	Primary key	Stores service booking id.
user_id	Int(11)	Foreign key	Stores id of user who book the service.
s_id	Int(11)	Foreign key	Stores id of service provider.
Name	Varchar(255)	Not null	Stores name of service booker.
Email	Varchar(255)	Not null	Stores email of service booker.
Phone	Varchar(20)	Not null	Stores phone of service booker.
Address	Text	Not null	Stores address of service booker.
Booking_date	timestamp	Not null	Stores booking date.

Table name: user

Primary key : user_id

[Table 5: user]

Field name	Data type	constraints	Description
User_id	Int(11)	Primary key	Stores user's id.
Email	Varchar(100)	Not null	Stores user's email.
Password	Varchar(255)	Not null	Stores user's password.
User name	Varchar(50)	Not null	Stores user's username
Created_at	timestamp	Not null	Stores account creation date.

Table name: service

Primary key: s_id

Foreign key: mem_id

[Table 6: service]

Field name	Data type	constraint	Description
S_id	Int (11)	Primary key	Stores service id.
Mem_id	Int(11)	Foreign key	Stores member id.
Name	Varchar(255)	Not null	Stores name of service.
Icon	Varchar(100)	Not null	Stores icon of service.
Description	Text	Not null	Stores description of service.
Service link	Varchar(255)	Not null	Stores service link.
Created_at	Timestamp	Not null	Stores service creation date.

Table name: member

Primary key: mem_id

[Table 7: member]

Field name	Data type	constraints	Description
mem_id	Int(11)	Primary key	Stores member's id.
Email	Varchar(100)	Not null	Stores member's email.
Password	Varchar(255)	Not null	Stores member's password.
User name	Varchar(50)	Not null	Stores member's username
Created_at	timestamp	Not null	Stores account creation date.

Table name: courses

Primary key: c_id

Foreign key: mem_id

[Table 8: courses]

Field name	Data type	constraint	Description
c_id	Int (11)	Primary key	Stores course id.
Mem_id	Int(11)	Foreign key	Stores member id.
Title	Varchar(255)	Not null	Stores title of course.
Image	Varchar(255)	Not null	Stores image of course.
Description	Text	Not null	Stores description of course
Enroll link	Varchar(255)	Not null	Stores enroll_link.
Created_at	Timestamp	Not null	Stores course creation date.

Table name: freelancers

Primary key: freelancer_id

[Table 9: freelancers]

Field name	Data type	constraints	Description
freelancer_id	Int(11)	Primary key	Stores freelancer's id.
Name	Varchar(255)	Not null	Stores freelancer's name.
Expertise	Varchar(255)	Not null	Stores freelancer's expertise.
Profile_image	Varchar(255)	Not null	Stores freelancer's profile image.
Contact_link	Varchar(255)	Not null	Stores contact link.
Created_at	timestamp	Not null	Stores account creation date.

Table name: freelancer_contact

Primary key: id

Foreign key: freelancer_id
[Table 10: freelancers_contact]

Field name	Data type	constraints	Description
id	Int(11)	Primary key	Stores id.
Freelancer_id	Int(11)	Foreign key	Stores freelancer's id.
Email	Varchar(255)	Not null	Stores freelancer's email.
Message	Text	Not null	Stores message.
name	Varchar(255)	Not null	Stores freelancer's name.
Created_at	timestamp	Not null	Stores creation date.

CHAPTER 5

TECHNICAL SPECIFICATION

5.1 Hardware Specification

5.1.1 RAM:500 MB

5.1.2 Hard Drive Storage Needed: 10MB

5.2 Platform

5.2.1 Supported Operating System: Cross Platform

5.2.2 Programming Server: APACHE Server (AMP Module)

5.3 Framework

5.3.1 Markup Language: HTML

5.3.2 Programming Language: PHP

5.3.3 Scripting Language: Java Script

5.4 Technical Specification

5.4.1 Front-End: HTML 5, CSS 4, Java Script

5.4.2 Back-End: MySQL 8.2, PHP 8.4

5.4.2 IDE: VS code, Notepad++

5.4.2 UML Tools: Draw.io

5.4.2 SRS Tools: Microsoft Word 2019

CONCLUSION

This project successfully implements a dynamic platform where users can register, enroll in courses, book services, and contact freelancers. The system ensures seamless data management using a MySQL database, with auto-incrementing IDs for unique user identification. Services, courses, and freelancers are fetched dynamically, making the platform scalable. The integration of clickable links enhances user experience, allowing easy access to enrollments and bookings. Secure data handling is ensured through prepared statements. Responsive styling ensures accessibility across devices. Overall, this project provides a functional, user-friendly, and efficient solution for managing online services, education, and freelancing in a single platform.

BIBLIOGRAPHY

➤ **Book references:**

- 1) Anuradha A. Puntambekar, "Software Engineering"
- 2) Jon Duckett, "HTML and CSS; design and build websites", Wiley. [Edition 2]
- 3) W. Jason Gilmore, "Beginning PHP and MySQL from novice to professional", A press. [Edition 4]

➤ **Website references:**

- 1) MDN Web Docs= <https://developer.mozilla.org/>
- 2) PHP= <https://www.php.net/>