

LAB REPORT

Submitted by

BHOGI SARANYA {RA2011003011100}

Under the Guidance of

Dr/Mr/Ms. <Faculty Name>

<Designation>, <Department>

In partial satisfaction of the requirements for the degree of

**BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING**

with specialization in <.>



SCHOOL OF COMPUTING

**COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

KATTANKULATHUR - 603203

JUNE 2022



**SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203**

BONAFIDE CERTIFICATE

Certified that this lab report titled “.” is the bonafide work done by <Name of the Student> (RA.) who carried out the lab exercises under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Ms/Mr/Dr. <Faculty Name>

SEPM – Course Faculty

<Designation>

Department of < >

ABSTRACT

Communication through internet is becoming vital these days. An online communication allows the users to communicate with other people in a fast and convenient way. Considering this, the online communication application must be able share the texts or images or any other files in a faster way with minimum delay or with no delay. Firebase is one of the platforms which provides a real-time database and cloud services which allows the developer to make these applications with ease. Instant messaging can be considered as a platform to maintain communication. IOS provides better platform to develop various applications for instant messaging compared to other platforms such as Android. The main objective of this paper is to present a software application for the launching of a real time communication between operators/users. The system developed on IOS will enable the users to communicate with another users through text messages with the help of internet. The system requires both the device to be connected via internet. This application is based on IOS with the backend provided by google Firebase.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	
	LIST OF FIGURES	
	LIST OF ABBREVIATIONS	
1	PROBLEM STATEMENT	
2	STAKEHOLDERS & PROCESS MODELS	
3	IDENTIFYING REQUIREMENTS	
4	PROJECT PLAN & EFFORT	
5	WORK BREAKDOWN STRUCTURE & RISK ANALYSIS	
6	SYSTEM ARCHITECTURE, USE CASE & CLASS DIAGRAM	
7	ENTITY RELATIONSHIP DIAGRAM	
8	DATA FLOW DIAGRAM	
9	SEQUENCE & COLLABORATION DIAGRAM	
10	DEVELOPMENT OF TESTING FRAMEWORK/USER INTERFACE	
11	TEST CASES & REPORTING	
12	ARCHITECTURE/DESIGN/FRAMEWORK/IMPLEMENTATION	
	CONCLUSION	
	REFERENCES	
	APPENDIX (CODE)	

LIST OF FIGURES

FIGURE NO	TITLE	EXP NO
Fig 1.	LOGO	1
Fig 2.	SWOT ANALYSIS	5
Fig 3.	USE CASE DIAGRAM	6
Fig4.	CLASS DIAGRAM	6
Fig 5	ER DIAGRAM	7
Fig 6	Data Flow Diagram	8
Fig 7	Sequence Diagram	9
Fig 8	Collaboration Diagram	9
Fig 9	UI Interface	10



Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	1
Title of Experiment	To identify the Software Project, Create Business Case, Arrive at a Problem Statement
Name of the candidate	BHOGI SARANYA
Team Members	KRISHNA PRAKASH, BHOGI SARANYA
Register Number	RA2011003011100
Date of Experiment	23/03/2022

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To Frame a project team, analyze and identify a Software project. To create a business case and Arrive at a Problem Statement for the

:- ONE CLICK

Team Members:

S. No	Register No	Name	Role
1	RA2011003011112	SAI PRASAD	Lead/Rep
2	RA2011003011092	KRISHNA PRAKASH	Member
3	RA2011003011100	BHOGI SARANYA	Member

Project Title: ONE CLICK

Project Description

It's a chart app. with this app users can message others instantly and can chat in realtime. users can login with their facebook and google account for easy registration.

users can also send their pictures and location
Business Case

<Incorporate the Business Case template>

Result

Thus, the project team formed, the project is described, the business case was prepared and the problem statement was arrived.

ONE PAGE BUSINESS CASE TEMPLATE

DATE	22/03/2022
SUBMITTED BY	Sai Prasad, Krishna Prakash, Bhogi Saranya, Mukundaan
TITLE / ROLE	Team Leader



THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

It's a chart app. with this app users can message others instantly and can chat in realtime.
users can login with their facebook and google account for easy registration.
users can also send their pictures and location

THE HISTORY

In bullet points, describe the current situation.

In situation users can't make calls, this app will allow users to send messages

LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack of special training, etc.

Limitation of our project. users can't make audio call or can't send audio messages.

and also can't make video calls and video messages

APPROACH

List what is needed to complete the project.

Xcode software and swift language

BENEFITS

In bullet points, list the benefits that this project will bring to the organization.

users can send messages ,pictures and location instantly and in realtime



Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	2
Title of Experiment	Identification of Process Methodology and Stakeholder Description
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD, KRISHNA PRAKASH, MUKUNDAAN
Register Number	RA2011003011100
Date of Experiment	15-03-2022

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To identify the appropriate Process Model for the project and prepare Stakeholder and User Description.

Team Members:

Sl No	Register No	Name	Role
1	RA2011003011112	SAI PRASAD	Rep/Member
2	RA2011003011100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

Project Title:

Selection of Methodology

- <Summarize their understanding of “Waterfall” or “Agile” Methodology>

Incorporate information to below table regarding stakeholders of the project [Make use of below examples]

Stakeholder Name	Activity/ Area /Phase	Interest	Influence	Priority (High/ Medium/ Low)	
SAI PRASAD	PROJECT MANAGEMENT	HIGH	HIGH	HIGH	
BHOGI SARANYA	MARKETING/ ADVERTISING	MEDIUM	MEDIUM	MEDIUM	
KRISHNA PRAKASH	INITIAL AND FREQUENT BASED ON PROGRESS	MEDIUM	MEDIUM	MEDIUM	
MUKUNDAAN	INITIAL DISCUSSION	LOW	MEDIUM	MEDIUM	

/ *

For Example

Stakeholder Name	Activity / Area / Phase	Interest	Influence	Priority (High / Medium/Low)
Regional Head of Sales & Marketing	Subscription using mobile App	High	High	1
Finance Account Receivable consultant	Multiple Currency Payment	High	Low	3

. Interest and Influence matrix

Interest	Influence
High	High
Low	Low
Low	High
High	Low

Stakeholder	Interests	Estimated Project Impact	Estimated Priority
Owner	Achieve targets, Increase sales margin	High	1
Sponsor	Provides new market to expand ventures Negotiate funding for project Reviews changes to project environments.	Med	3
Team members	Demand incentives Retain and upgrade skills New product excitement	High	2
Project Manager	Lead the team in every aspect. Accountable for entire project scope, team, success & failure	High	2
Investors	Promoter of the investment, Provides necessary financial resources	Low	5
Resource Manager	Resource planning and allocation. Ensuring adequate resource according to project needs and budget.	Med	4
Suppliers	Ensuring feasible and realistic in every aspect Managing divergence from budgeted cost.	Med	6
End Users	Provides feedback	Low	7



Result

Thus the Project Methodology was identified and the stakeholders were described.



Department Of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	3
Title of Experiment	System, Functional and Non-Functional Requirements of the Project
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD,KRISHNA PRAKASH,MUKUNDAAN
Register Number	RA2011003011100
Date of Experiment	4/04/2022

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To identify the system, functional and non-functional requirements for the project.

Team Members:

S No	Register No	Name	Role
1	RA201100301112	SAI PRASAD	Rep/Member
2	RA201100301100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

Project Title: ONE CLICK

System Requirements

Processor: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.59 GHz

OS: MAC OS

SOFTWARE: XCODE

Installed RAM: 8.0 GB (7.9GB usable)

System type: 64-bit operating system, x64-based processor

Pen and touch: No pen or touch input is available for this display

SWIFT

2GB Graphics card

IOS 12.0 or higher

High speed internet connection

Functional Requirements

Data output

UI/UX

Additional Extensions

Virtualization

Firebase

API

User requirement

Authorization levels

Authentication

Reporting requirement

Administrative function

Non-Functional Requirements

Security

Data discovery

Data extraction

Extraction scale

performance parameter

recoverability

maintainability

environmental requirement

Result

Thus the requirements were identified and accordingly described.



Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	4
Title of Experiment	Prepare Project Plan based on scope, Calculate Project effort based on resources and Job roles and responsibilities
Name of the candidate	Bhogi Saranya
Team Members	Sai Prasad, Bhogi Saranya, Krishna Prakash, Mukundaan
Register Number	RA2011003011100
Date of Experiment	20/04/2022

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim:

To Prepare Project Plan based on scope, Calculate Project effort based on resources, Find Job roles and responsibilities.

Team Members:

Sl No	Register No	Name	Role
1	RA201100301112	Gudari Sai Prasad	Lead
2	RA201100301100	Bhogi Saranya	Member
3	RA2011003011092	Krishna Prakash	Member

THE PROJECT PLAN TEMPLATE:

- Our project title is TEXT IN, it is a chatting messenger app for ios
- In this project we build an messenger app for ios users
- For this project we are using:
 - a. Swift and Firebase
 - b. Softwares used XCODE
- First, we will plan our project by dividing the project in sub task and allot each task to the respective domain person.

- As our project is based on website building the tasks are:
 - a. Forming an outlook of our app.
 - b. Making the UI easy to operate.
 - c. Designing the complete app.
 - d. Completing app with presentation look.
 - e. Doing the final corrections and modifications needed.
 - f. Reviewing the app.
 - g. Implementing the app.
 - h. Promoting the app

RESULT:

Thus, the Project Plan was documented successfully.

1. Project Management Plan

Describe the key issues driving the project. [Min 3 Focus Areas]

Focus Area	Details
Schedule Management	Time estimates for all project task: 3.5 months Start date for the task: 23 March 2022 Last date of the task: July 2022 Sequence of tasks: <ul style="list-style-type: none">• Layout design• Website development using HTML, CSS• Mini search engine using Python• Connecting search engine using API• Finally Deploying Website on Servers.
Quality Management	Quality Assurance: Quality assurance will be managed including governance, roles and responsibilities, tools and techniques and reporting Quality Control: Specify the mechanisms to be used to measure and control the quality of the work products
Resource Management	People: Team members required with Python, ML, HTML, CSS skills Resources: Open source API Finance: Budget Required Physical: Laptop with high specification
Stakeholder	Identifying, Analysing, Engaging Stakeholders
Communication Management	Determine communication requirements: <ul style="list-style-type: none">• Work with team members to generate new ideas and strategies and share among each other. Roles and responsibilities:

	<ul style="list-style-type: none"> Main duties include preparing detailed reports. <p>Tools and techniques:</p> <ul style="list-style-type: none"> ZOOM, GMEET, LAPTOP
Risk Management	<ul style="list-style-type: none"> Privacy risk Open source is limited Lack of data which might lead to bad reviews

2.Estimation

Effort and Cost Estimation

Activity Description	Sub-Task	Sub-Task Description	Effort (in hours)	Cost in INR
Login	Sign in	User will enter email id and password.	3	300
	Verification	A mail will be sent to the registered email for verification.		
	Sign out	Now user can search the college.		
Search box		System allows the user to search for the who have already registered in the app	5	500

		User will get required search results		
Effort (hr)	Cost (INR)			
1	100			

/Resource Cost [CapEx]

Onetime Infra requirements

Infrastructure Requirement	Qty	Cost per qty	Total Cost
Web server	2	1,50,000	3,00,000
API	1	20,000	20,000

Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, System, Middleware and DB admin Developer , Support Consultant	3	2,000,000	6,000,000
License	Operating System Database Middleware IDE	10	10000	100,000
Infrastructures	Server, Storage and Network	20	20000	400,000

3.Project Team Formation

Identification Team members

Name	Role	Responsibilities
Krishna Prakash	Key Business User (Product Owner)	Provide clear business and user requirements
Bhogi Saranya	Project Manager	Manage the project
Sai Prasad	Business Analyst	Discuss and Document Requirements
Sai Prasad	Technical Lead	Design the end-to-end architecture
Mukundaan	UX Designer	Design the user experience
Bhogi Saranya	Frontend Developer	Develop user interface

Krishna Prakash	Backend Developer	Design, Develop and Unit Test Services/API/DB
Mukundaan	Cloud Architect	Design the cost effective, highly available and scalable architecture
Bhogi Saranya	Cloud Operations	Provision required Services
Krishna Prakash	Tester	Define Test Cases and Perform Testing

Responsibility Assignment Matrix

RACI Matrix		Team Members			
Activity		Name (BA)	Name (Developer)	Name (Project Manager)	Key Business User
User Requirement Documentation		A (Bhogi Saranya)	C / I (Sai Prasad)	I(Bhogi Saranya)	R (Krisha Prakash, Mukundaan)
A	Accountable				
R	Responsible				
C	Consult				
I	Inform				

Reference

1. <https://www.pmi.org/>
2. <https://www.projectmanagement.com/>
3. <https://www.tpsgc-pwgsc.gc.ca/biens-property/sngp-npms/ti-it/ervcpgrm-dsfvpmppteng.html>



Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	5
Title of Experiment	Prepare Work breakdown structure, Timeline chart, Risk identification table
Name of the candidate	Bhogi Saranya
Team Members	Sai Prasad Krishna Prakash Mukundaan
Register Number	RA2011003011100
Date of Experiment	12.04.2022

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

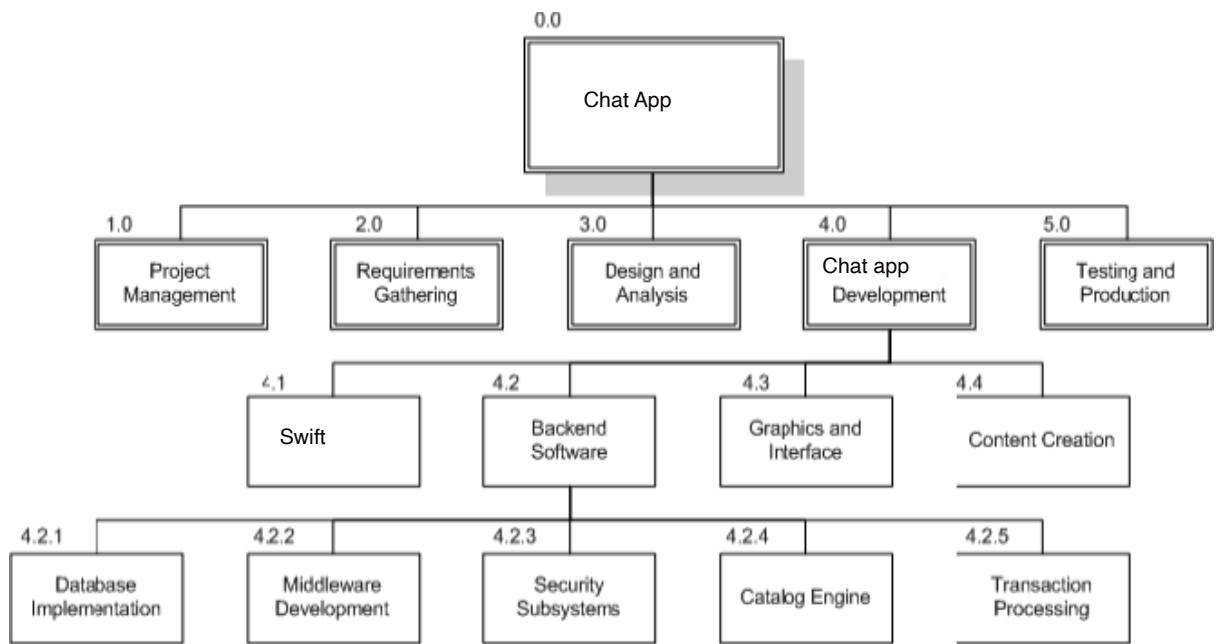
To Prepare Work breakdown structure, Timeline chart and Risk identification table

Team Members:

Sl No	Register No	Name	Role
1	RA201100301112	Gudari Sai Prasad	Rep
2	RA2011003011100	Bhogi Saranya	Member
3	RA2011003011092	Krishna Prakash	Member

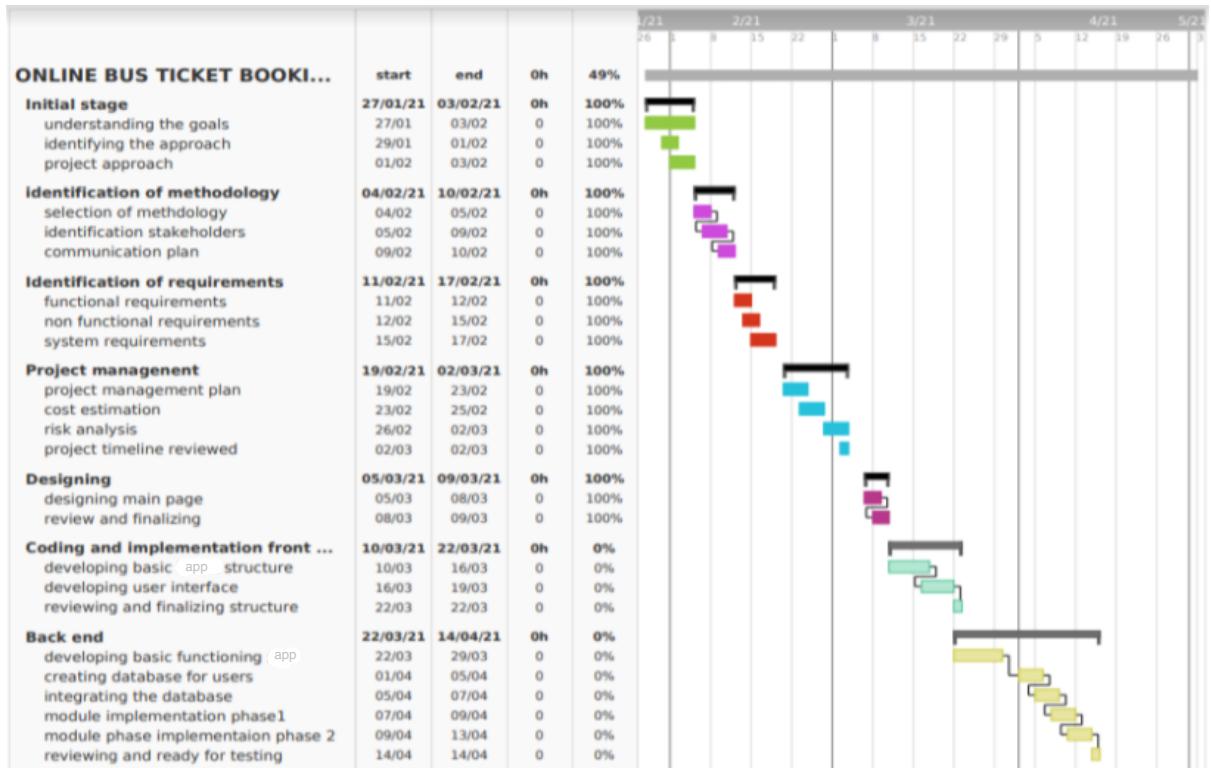
<Incorporate WBS, Timeline chart and Risk table>

WBS – Examples

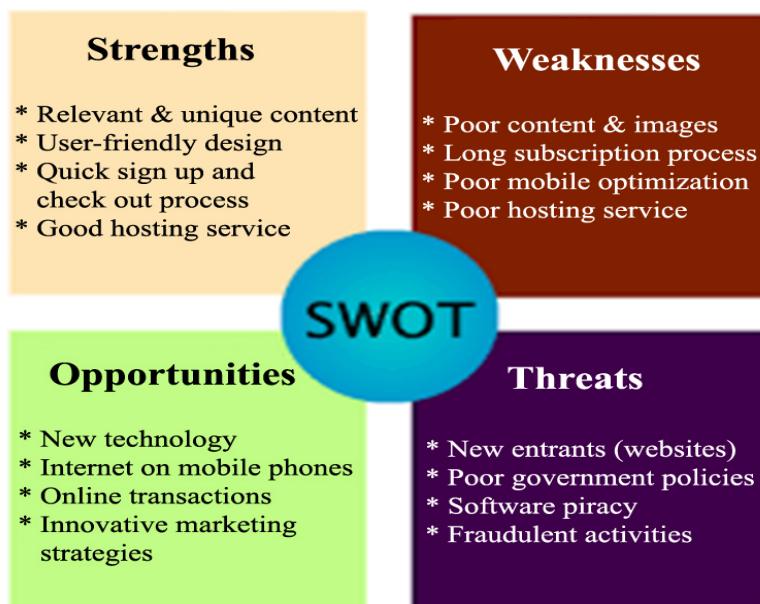


- 0.0 Chat App
 - 1.0 Project Management
 - 2.0 Requirements Gathering
 - 3.0 Analysis & Design
 - 4.0 Site Software Development
 - 4.1 Swift Design and Creation
 - 4.2 Backend Software
 - 4.2.1 Database Implementation
 - 4.2.2 Middleware Development
 - 4.2.3 Security Subsystems
 - 4.2.4 Catalog Engine
 - 4.2.5 Transaction Processing
 - 4.3 Graphics and Interface
 - 4.4 Content Creation
 - 5.0 Testing and Production

TIMELINE – GANTT CHART



RISK ANALYSIS – SWOT & RMMM





Risk Management Framework- Risks And Mitigation ...

Response	Strategy	Examples
Avoid	Risk avoidance is a strategy where the project team takes action to remove the threat of the risk or protect from the impact	<ul style="list-style-type: none">Extending the scheduleReducing/removing scopeChange the execution strategy
Transfer	Risk transference involves shifting or transferring the risk threat and impact to a third party. Rather transfer the responsibility and ownership	<ul style="list-style-type: none">Purchasing insurancePerformance bondsWarrantiesContract issuance (lump sum)
Mitigate	Risk mitigation is a strategy where the project team takes action to reduce the probability of the risk occurring. This does not risk or potential impact , but rather reduces the likelihood of it becoming real.	<ul style="list-style-type: none">Increasing testingChanging suppliers to a more stable oneReducing process complexity
Accept	Risk acceptance means the team acknowledges the risk and its potential impact, but decides not to take any preemptive action to prevent it. It is dealt with only if it occurs.	<ul style="list-style-type: none">Contingency reserve budgetsManagement schedule floatEvent contingency

Slide 1 of 5

WBS and Risk Management Plan

Lab Session #5

Table of Contents

1. Executive Summary	2
2. WBS With Project Schedule	2
3. Risk Identification	2
3.1. List (Describe) Register	2
3.2. Managing Risk	2
Reference	3

1. Executive Summary

In our project, a milestone is a specific point in time within our project lifecycle which is used to measure the progress of Park My Pro toward its ultimate goal. In project management, milestones are used as signal posts for significant events, decision points, or deliverables such as: The project's start date (March 2022) and Project end date(July 2022)

Risk management is the process of analysing exposure to risk and determining how to best handle such exposure. Our project undertakes a best practices approach and focuses on understanding the key risks and managing them within acceptable levels.

2. WBS With Project Schedule

< Assign team members for sub-tasks based on RACI and skill requirement>

Module (#)	Activity (#)	Sub-Task(#)	Assignee(s)	Planned Start Date	Planned End Date	Actual Start Date	Actual End Date	Status
UX Design	Design the user interface	Display the user privacy policy	Bhogi Saranya	08 Mar 2022	15 Mar 2022	-	-	Open
		Create a login page for the user to enter their credentials		11 Mar 2022	20 March 2022	10 Apr 2022	10 Apr 2022	Closed
		Present information provided by the backend server		10 May 2022	20 May 2022	-	-	Open
Database management	Designing backend	Refer to what data must be exchanged between user and admin	Gudari Sai Prasad	08 Apr 2022	1 May 2022	10 Apr 2022	12 Apr 2022	Closed
		Encrypt credentials		15 Apr 2022	16 May 2022	-	-	Open
		Acquire the empty parking slots from park my pro using the credentials provided		05 May 2022	09 May 2022	11 Apr 2022	15 Apr 2022	Closed
		Integrate back end to the front end of the app		24 May 2022	25 June 2022	-	-	Open

Admin	Control and repository maintenance	Create a privacy policy for the user to ensure the safety of their credentials	Gudari Sai Prasad	03 June 2022	30 June 2022	-	-	Open
		Maintain the server / repository		10 June 2022	14 June 2022	-	-	Open
Test releases	Testing	Test the software with multiple test cases	Krishna Prakash,Mukundaan	14 June 2022	25 June 2022	-	-	Open
		Report bugs		20 June 2022	26 June 2022	-	-	Open

3. Risk Identification

SWOT ANALYSIS

STRENGTHS	WEAKNESSES
<ul style="list-style-type: none"> Strong need to students and staff in the institution Knowledgeable team We are able to respond very quick and give good customer care 	<ul style="list-style-type: none"> Low reputation Small team Unforeseen problems
OPPORTUNITIES	THREATS
<ul style="list-style-type: none"> Broadening the team Broadening the project Unique product 	<ul style="list-style-type: none"> Failing to use https or any other security protocols Sudden rework of Park My Pro framework Technical bugs

3.1. List (Describe) Register

<Issue can potentially occur in future and list all risks identified >

Risk ID (#)	Risk Description	Impact Description
R01	Technical Bugs	Sudden rework of the Park My Pro framework
R02	Disbanding of a team member	Not being able to complete the project before the deadline
R03	Developing Glitch	Failing to use https or any other security protocols
R04	Inability to gain user commitment	Lack of effort / diligence

3.2. Managing Risk

<Risk should be categorised So action can be derived to address these risks could become an issue in future>

Risk ID (#)	Status [Open / Closed]	Risk Appetite [Accept/ Mitigate/ Transfer/Avoid]	Action	Action Owner	Target Date	Remarks
R01	Closed	Accept	Circumscribe the damage, Look into the potential problems and take precautions	Team Member	05 July 2022	-
R02	Closed	Avoid/Mitigate	Training the developers skillfully without any further obstacles/ shortcomings	Team Leader	10 July 2022	-
R02	Closed	Accept	Communicate the fix and resolve the issue	Team Member	20 July 2022	-
R04	Closed	Accept	Frequent and clear communication with the user	Team Member	08 July 2022	-

Reference

1. <https://www.pmi.org/>
2. <https://www.projectmanagement.com/>
3. <https://www.tpsgc-pwgsc.gc.ca/biens-property/sngp-npms/ti-it/ervcpgrm-dsfvpmpmt-eng.html>

Result:

Thus, the work breakdown structure with timeline chart and risk table were formulated successfully.



Department of Networking and Communications

SRM IST, Kattankulathur – 603 20

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	6
Title of Experiment	Design a System Architecture, Use Case and Class Diagram
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD,KRISHNA PRAKASH,MUKUNDAAN
Register Number	RA201100301112, RA2011003011092, RA2011003011099
Date of Experiment	17/05/2022

Mark Split Up

S.No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To Design a System Architecture, Use case and Class Diagram

Team Members:

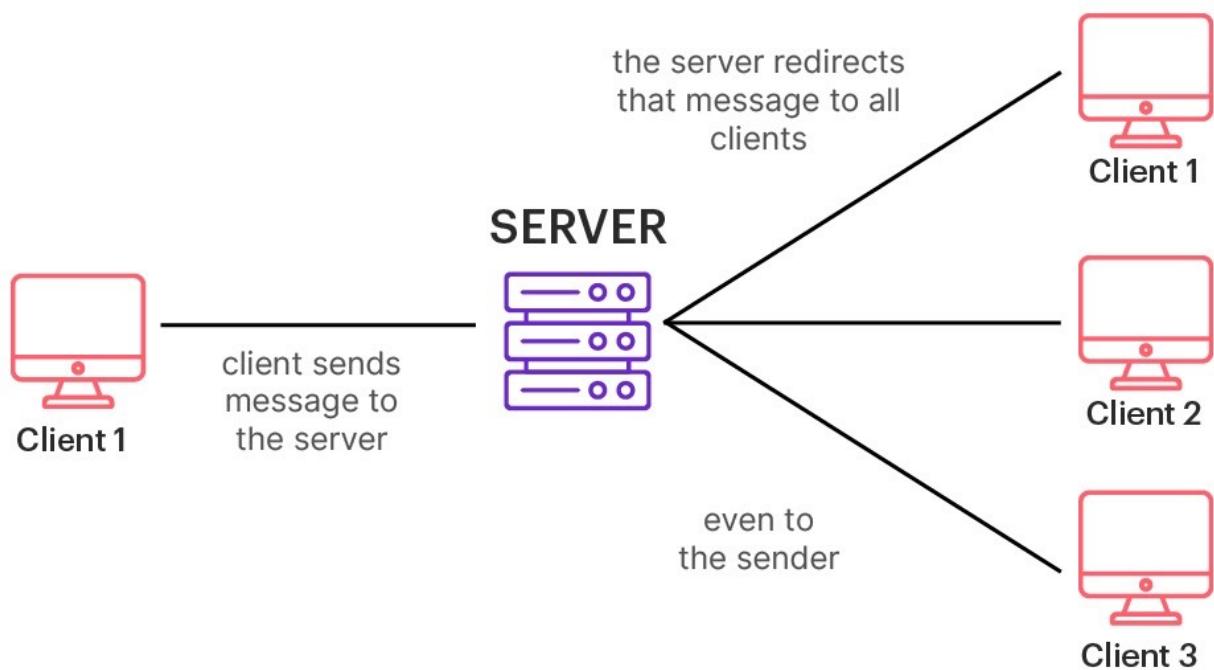
Sl No	Register No	Name	Role
1	RA201100301112	GUDARI SAI PRASAD	Rep

2	RA2011003011092	KRISHNA PRAKASH	Member
3	RA2011003011100	BHOGI SARANYA	Member

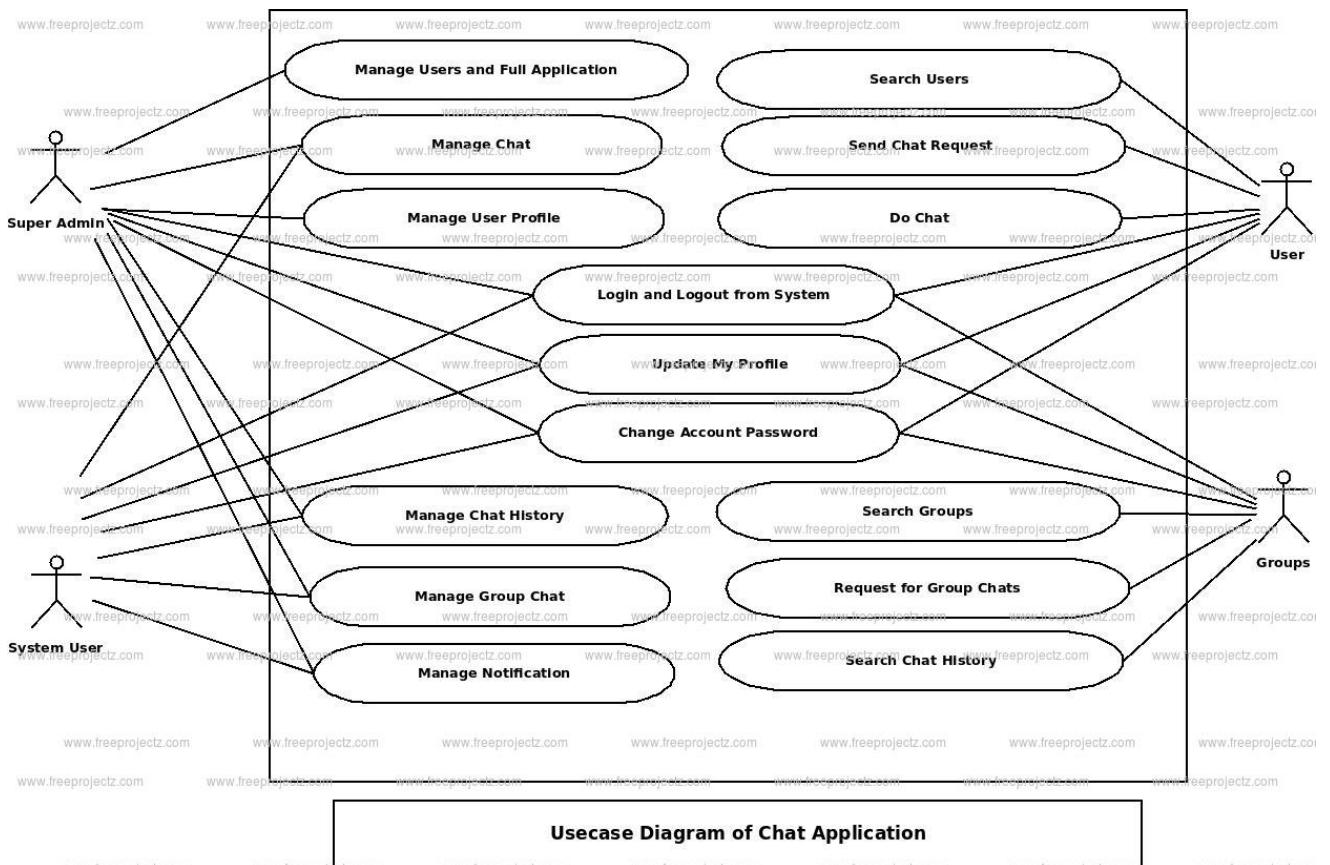
Requirements

<System Architecture, Use Case and Class Diagram>

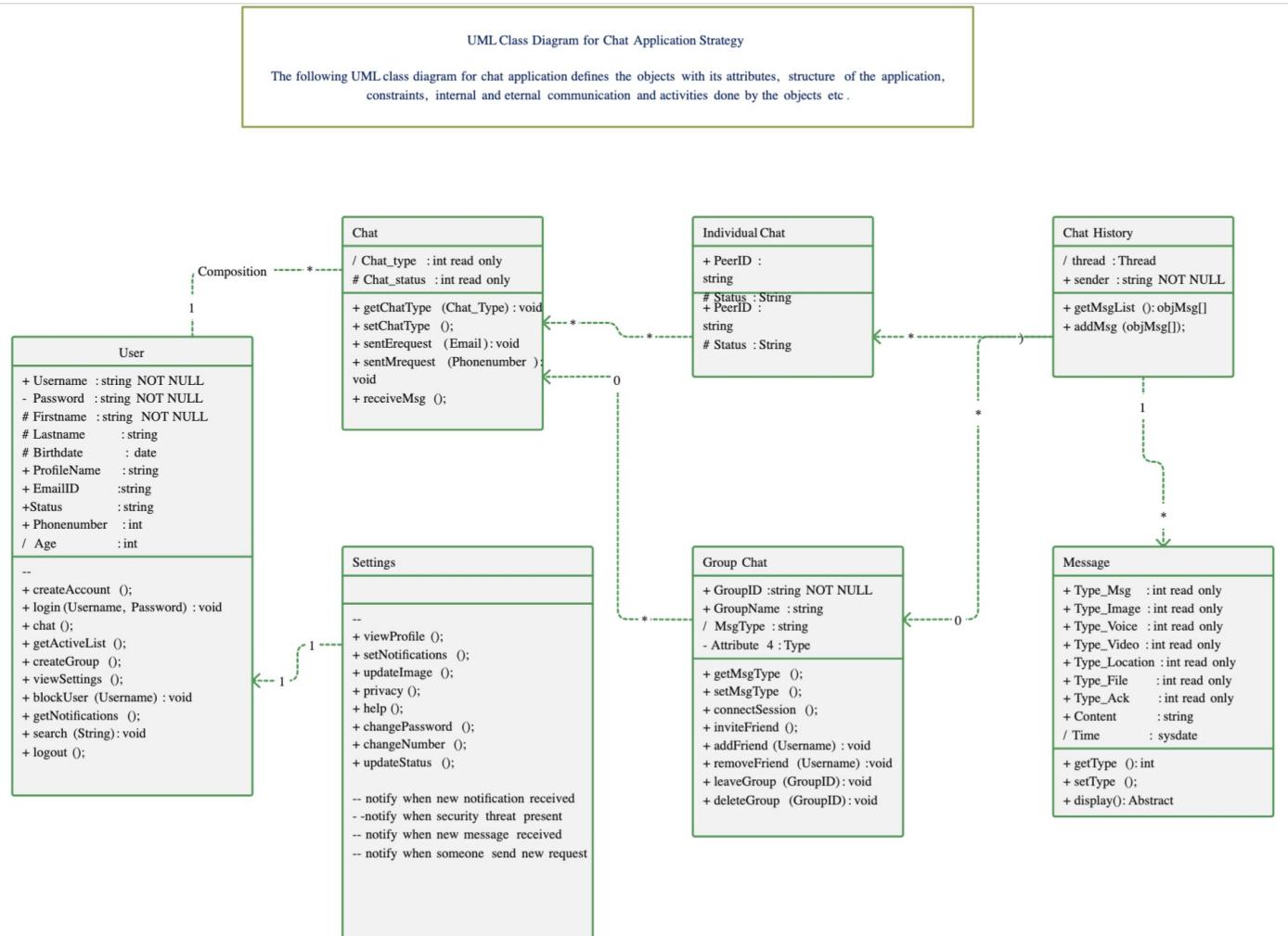
SYSTEM ARCHITECTURE



USE CASE DIAGRAM



CLASS DIAGRAM



Result:

Thus, the system architecture, use case and class diagram created successfully.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	7
Title of Experiment	Design a Entity relationship diagram
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD,KRISHNA PRAKASH,MUKUNDAAN
Register Number	RA201100301100
Date of Experiment	18-05-2002

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To create the Entity Relationship Diagram

Team Members:

S No	Register No	Name	Role
1	RA201100301112	GUDARI SAI PRASAD	Rep
2	RA2011003011100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

<ER Diagram >

Result:

Thus, the entity relationship diagram was created successfully.

***/ ER Diagram, Notation and Example**

What is ER Diagram?

- ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.
- ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.
- At first look, an ER diagram looks very similar to the flowchart. However, ER Diagram includes many specialized symbols, and its meanings make this model unique. The purpose of ER Diagram is to represent the entity framework infrastructure.

What is ER Model?

- ER Model stands for Entity Relationship Model is a high-level conceptual data model diagram. ER model helps to systematically analyze data requirements to produce a well-designed database.

- ER Model represents real-world entities and the relationships between them. Creating an ER Model in DBMS is considered as a best practice before implementing your database.
- ER Modeling helps you to analyze data requirements systematically to produce a well-designed database. So, it is considered a best practice to complete ER modeling before implementing your database.

Why use ER Diagrams?

Here, are prime reasons for using the ER Diagram

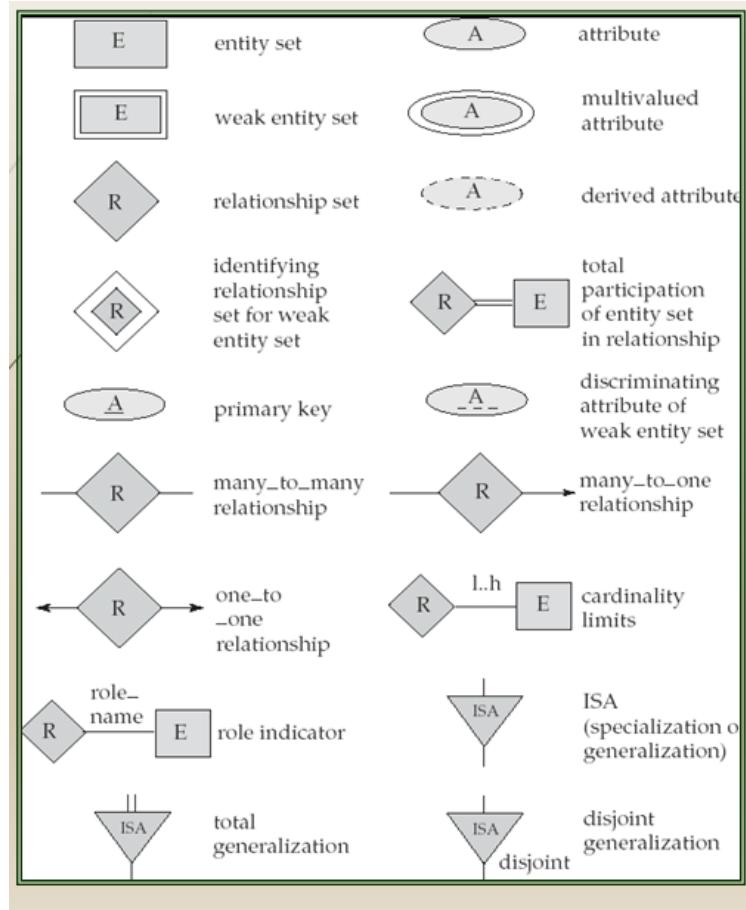
- Helps you to define terms related to entity relationship modeling
- Provide a preview of how all your tables should connect, what fields are going to be on each table
- Helps to describe entities, attributes, relationships
- ER diagrams are translatable into relational tables which allows you to build databases quickly
- ER diagrams can be used by database designers as a blueprint for implementing data in specific software applications
- The database designer gains a better understanding of the information to be contained in the database with the help of ERP diagram
- ERD Diagram allows you to communicate with the logical structure of the database to users

Components of the ER Diagram

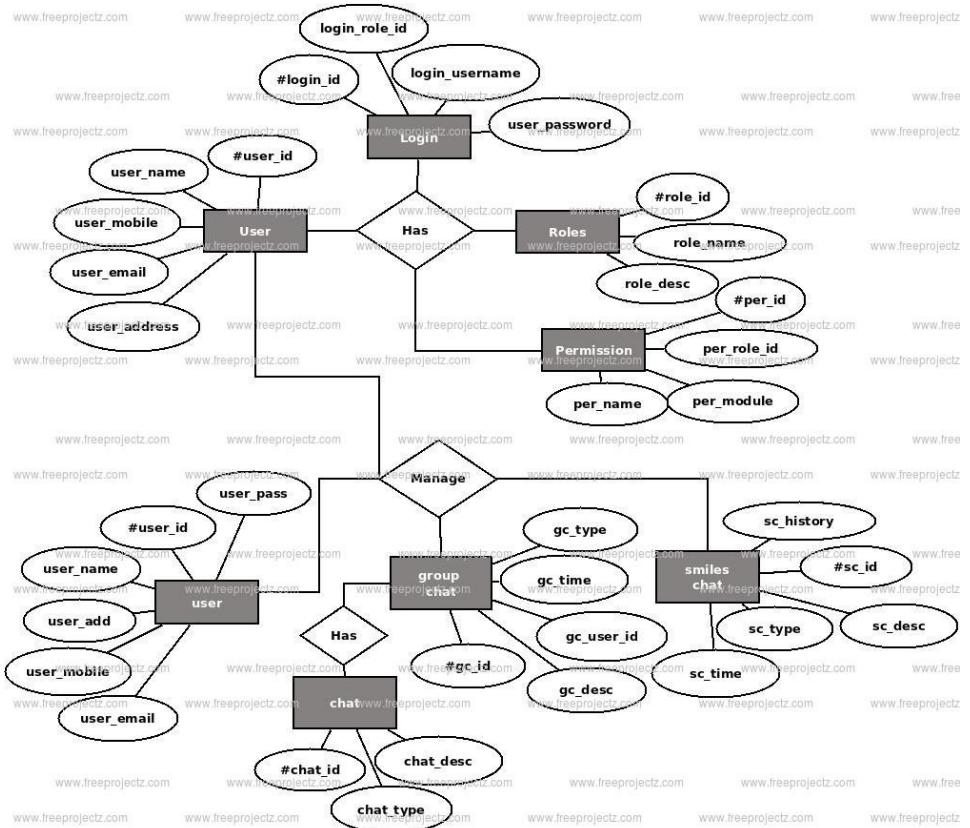
This model is based on three basic concepts: Entities, Attributes, Relationships

ER Diagram – Notations

- Rectangles represent entity sets.
- Diamonds represent relationship sets.
- Lines link attributes to entity sets and entity sets to relationship sets.
- Ellipses represent attributes
- Double ellipses represent multivalued attributes.
- Dashed ellipses denote derived attributes.
- Underline indicates primary key attributes



ER Diagram of driver drowsiness detection



ER Diagram For Chat Application

ADDITIONAL NOTES

- A database can be modeled as a collection of entities, relationship among entities.
- An entity is an object that exists and is distinguishable from other objects.

Example: specific person, company, event, plant

- Entities have attributes.

Example: people have names and addresses

- An entity set is a set of entities of the same type that share the same properties.

Example: set of all persons, companies, trees, holidays

- Express the number of entities to which another entity can be associated via a relationship set.

- Most useful in describing binary relationship sets.

- We express cardinality constraints by drawing either a directed line (->), signifying “one,” or an undirected line (—), signifying “many,” between the relationship set and the entity set.

- An entity is represented by a set of attributes, that is descriptive properties possessed by all members of an entity set.

Example: customer = (customer-id, customer-name, customer-street, customer-city)

loan = (loan-number, amount)

- Domain – the set of permitted values for each attribute

- Attribute types:

1. Simple and composite attributes.

2. Single-valued and multi-valued attributes

E.g. multivalued attribute: phone-numbers

3. Derived attributes-Can be computed from other attributes

E.g. age, given date of birth

Cardinality

- For a binary relationship set the mapping cardinality must be one of the following types:

1. One to one

A customer is associated with at most one loan via the relationship borrower. A loan is associated with at most one customer via borrower

2. One to many

A loan is associated with at most one customer via borrower, a customer is associated with several (including 0) loans via borrower

3. Many to one

A loan is associated with several (including 0) customers via borrower, a customer is associated with at most one loan via borrower

4. Many to many

A loan is associated with several (including 0) customers via borrower, a customer is associated with several loans (including 0) via borrower

Weak Entity Set

- An entity set that does not have a primary key is referred to as a weak entity set and represented by double outlined box in E-R diagram.

Example : Consider the entity set payment which got three attributes : payment_number, payment_date and payment_amount. Payment numbers are sequential starting from 1

generally separately for each loan. Although each payment entity is distinct, payments for different loans may share the same payment number. Thus this entity set does not have a primary key.

Discriminator

- The discriminator (or partial key) of a weak entity set is the set of attributes that distinguishes among all the entities of a weak entity set

Example: discriminator of weak entity set payment is the attribute payment_number since for each loan a payment number uniquely identifies one single payment for that loan.

Specialization-Generalization-ISA

- E-R model provides means of representing these distinctive entity groupings
- Process of designating subgroupings within an entity set is called specialization depicted by triangle component labelled ISA ("is a")
- Bottom up design process in which multiple entity sets are synthesized into higher level entity set - Generalization
- ISA relationship may also be referred to as superclass-subclass relationship
- Higher and lower level entity sets are designated by the terms superclass and subclass.
- Specialization and generalization are simple inversions of each other; they are represented in an E-R diagram in the same way.

Total & Partial Participation

- Total participation (indicated by double line): every entity in the entity set participates in at least one relationship in the relationship set

E.g. participation of loan in borrower is total, every loan must have a customer associated to it via borrower

- Partial participation: some entities may not participate in any relationship in the relationship set

Example: participation of customer in borrower is partial

Cardinality limits

- Cardinality limits can also express participation constraints
- Minimum and maximum cardinality is expressed as l..h where l is the minimum and h is the maximum cardinality
- Minimum value of 1 indicates total participation of entity set in relationship set
- Maximum value of 1 indicates entity participates in atmost one relationship set.
- Maximum value of * indicates no limit

Role indicator

- Entity sets of a relationship need not be distinct
- The labels "manager" and "worker" are called roles; they specify how employee entities interact via the works-for relationship set.
- Roles are indicated in E-R diagrams by labeling the lines that connect diamonds to rectangles.
- Role labels are optional, and are used to clarify semantics of the relationship

Disjoint Generalization

- Disjointness constraint requires that an entity belong to more than one lower level entity set.
- Example: account entity can satisfy only one condition for account_type attribute ; entity can either be savings or chequing account but not both.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	8
Title of Experiment	Develop a Data Flow Diagram (Process-Up to Level 1)
Name of the candidate	BHOGI SARANYA
Team Members	G.SAI PRASAD, KRISHNA PRAKASH, MUKUNDAAN
Register Number	RA201100301110019-05-2022
Date of Experiment	19-05-2022

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To develop the data flow diagram up to level 1 for the <project name>

Team Members:

S No	Register No	Name	Role
1	RA201100301112	SAI PRASAD	Rep
2	RA201100301100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

<DFD >

Result:

Thus, the data flow diagrams have been created for the <project name>.

Data Flow Diagram

The DFD takes an input-process-output view of a system. That is, data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software. Data objects are represented by labeled arrows, and transformations are represented by circles (also called bubbles). The DFD is presented in a hierarchical fashion. That is, the first data flow model (sometimes called a level 0 DFD or context diagram) represents the system as a whole. Subsequent data flow diagrams refine the context diagram, providing increasing detail with each subsequent level.

The data flow diagram enables you to develop models of the information domain and functional domain. As the DFD is refined into greater levels of detail, you perform an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding refinement of data as it moves through the processes that embody the application.

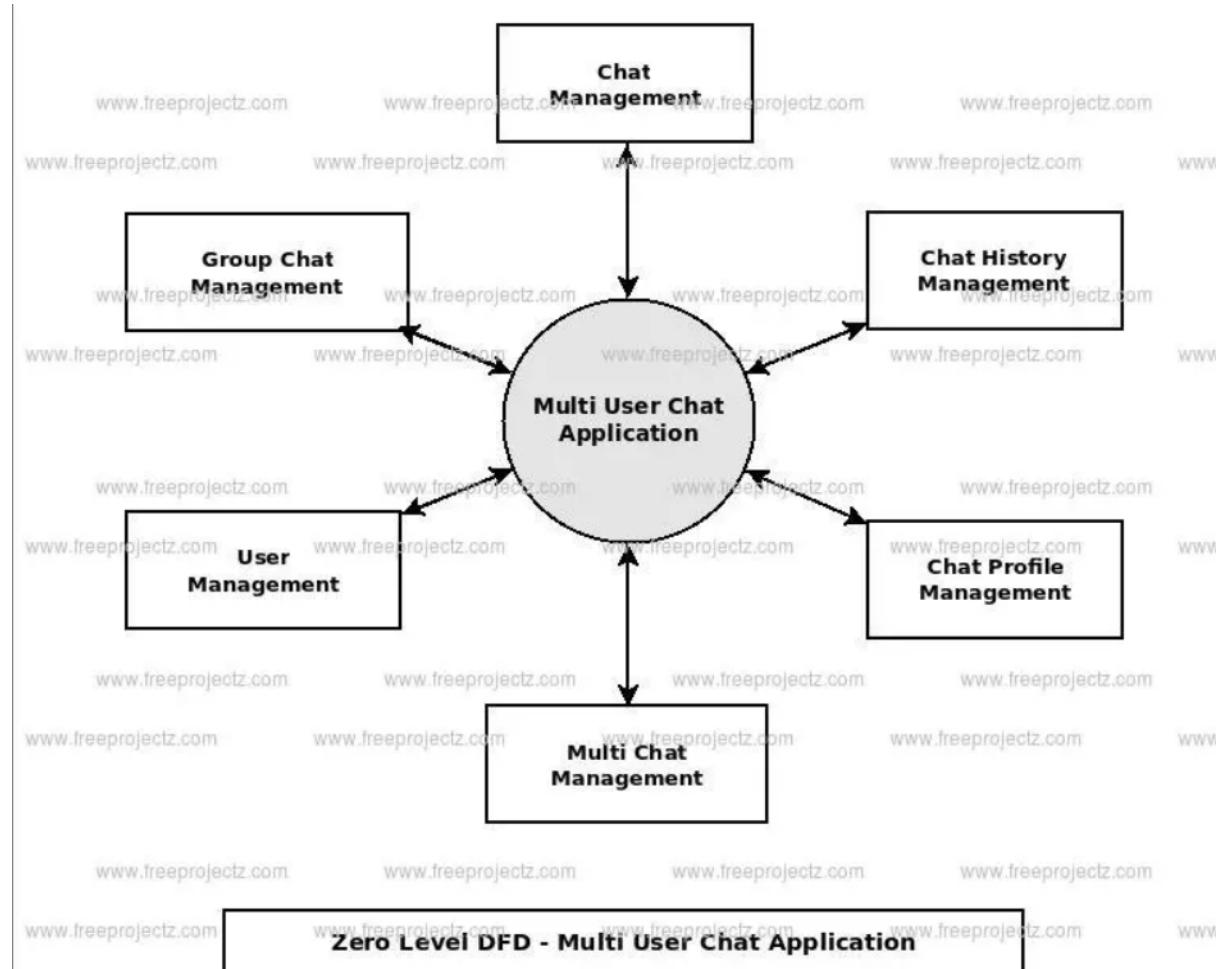
A few simple guidelines can aid immeasurably during the derivation of a data flow diagram:

- (1) Level 0 data flow diagram should depict the software/system as a single bubble;
- (2) Primary input and output should be carefully noted;
- (3) Refinement should begin by isolating candidate processes, data objects, and data stores to be represented at the next level;
- (4) All arrows and bubbles should be labeled with meaningful names;
- (5) Information flow continuity must be maintained from level to level and

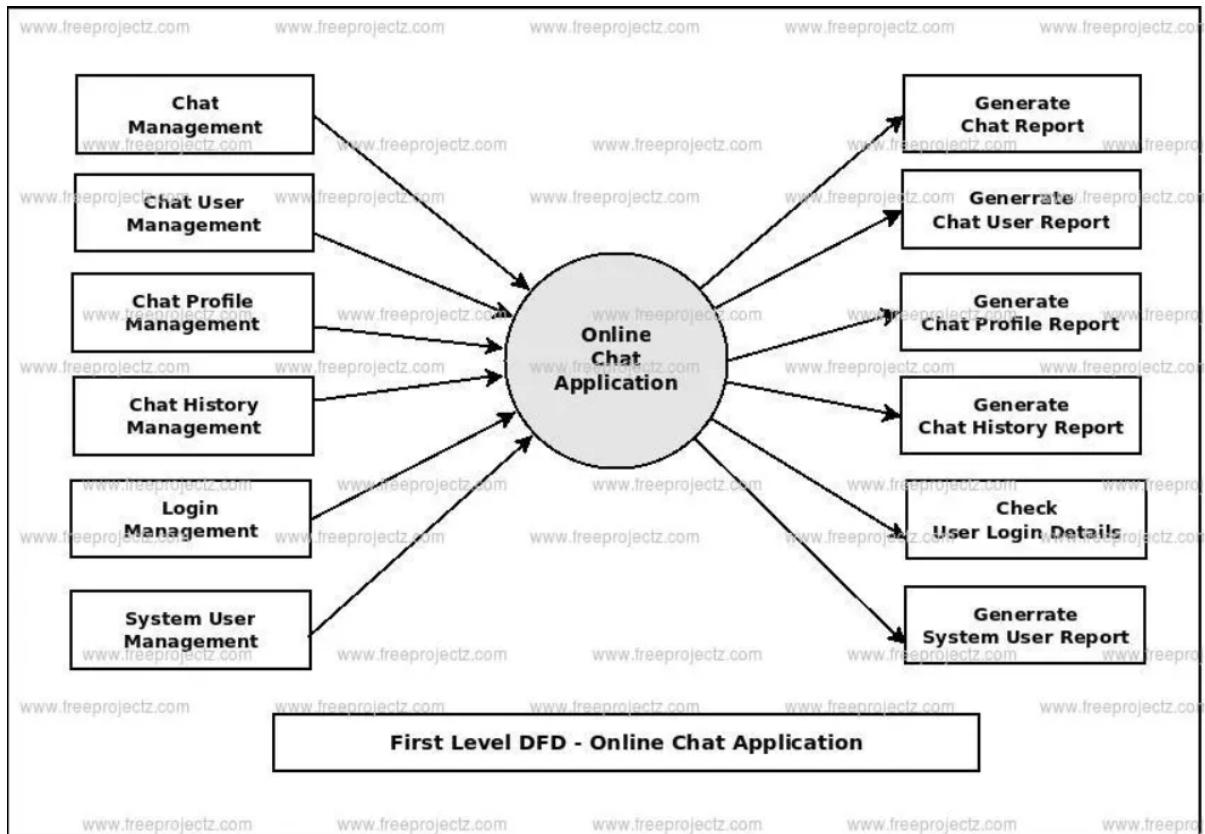
(6) One bubble at a time should be refined. There is a natural tendency to overcomplicate the data flow diagram. This occurs when you attempt to show too much detail too early or represent procedural aspects of the software in lieu of information flow.

*/ For Example

DFD Level 0



DFD Level 1



Result:

Thus, the data flow diagram have been created for the online chatting app



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	9
Title of Experiment	Design a Sequence and Collaboration Diagram
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD, KRISHNA PRAKASH, MUKUNDAAN
Register Number	RA2011003011100
Date of Experiment	09-06-2022

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

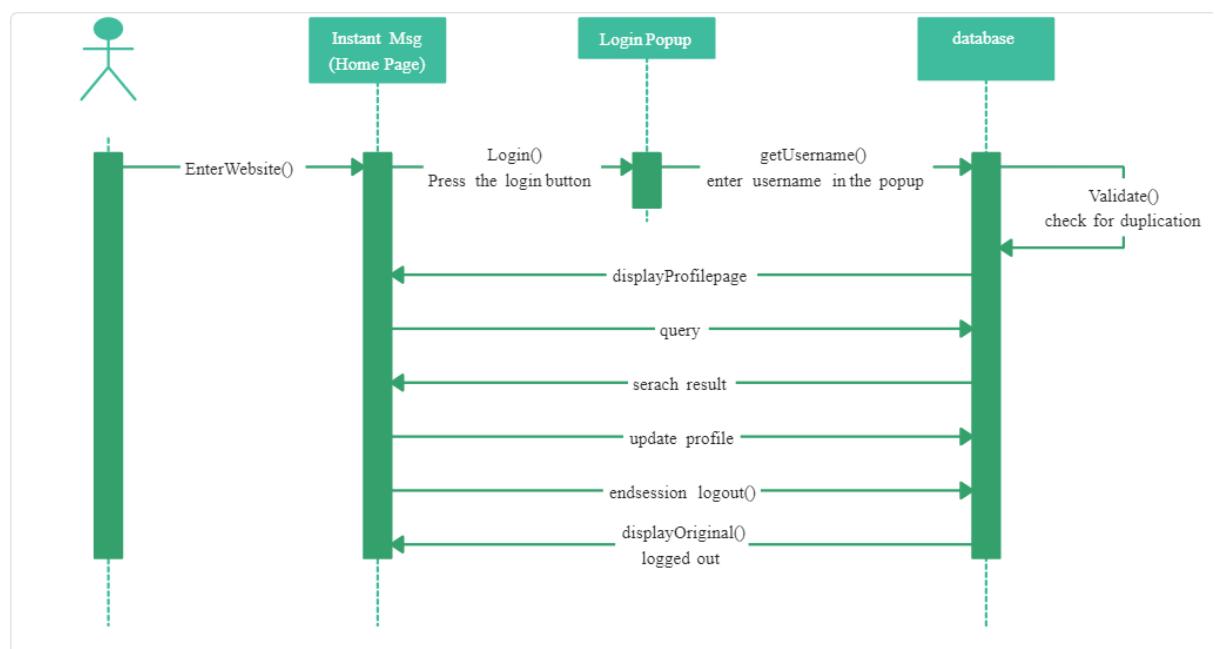
To create the sequence and collaboration diagram for the <project name>

Team Members:

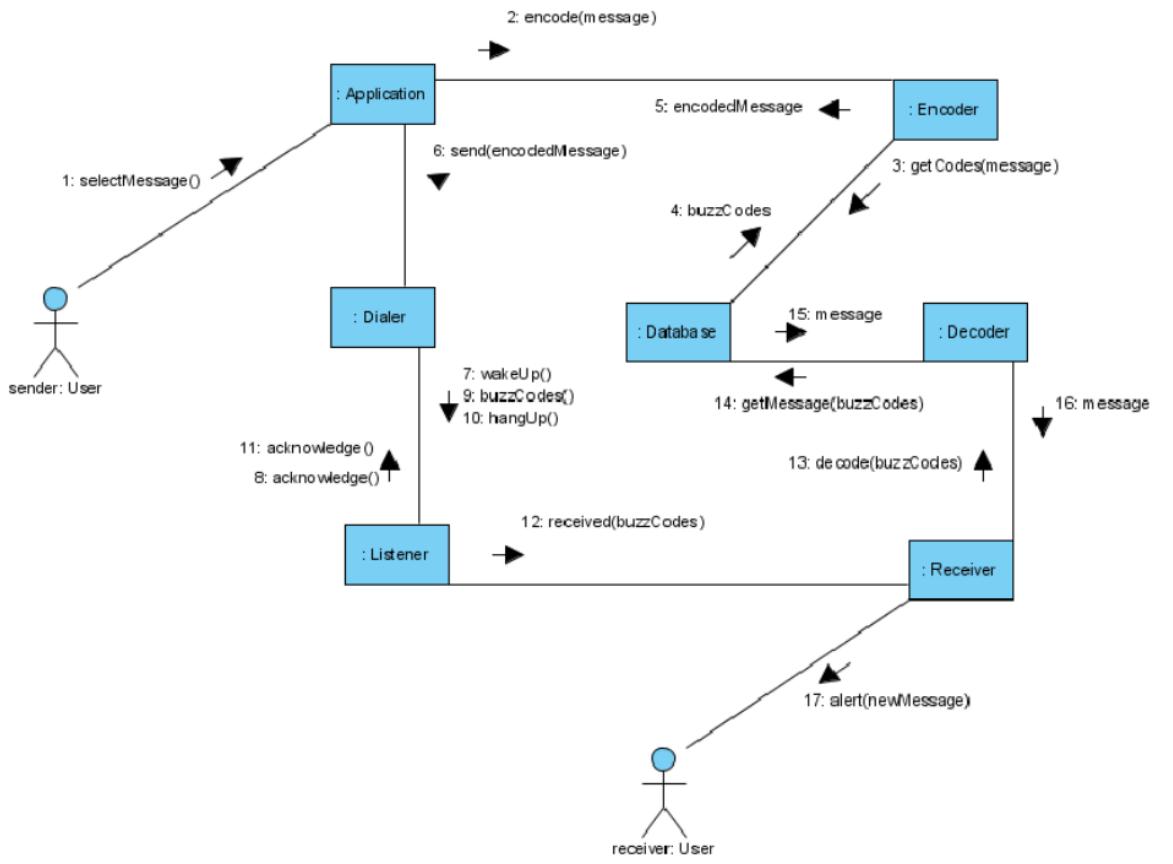
S No	Register No	Name	Role
1	RA2011003011112	G. SAI PRASAD	Rep/Member
2	RA2011003011100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

<Sequence and Collaboration Diagram>

SEQUENCE DIAGRAM



COLLABORATION DIAGRAM



Result:

Thus, the sequence and collaboration diagrams were created for the chatting app.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	10
Title of Experiment	Develop a Testing Framework/User Interface
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD, KRISHNA PRAKASH, MUKUNDAAN
Register Number	RA2011003011100
Date of Experiment	14/06/2022

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To develop the testing framework and/or user interface framework for the <car sales system>

Team Members:

S No	Register No	Name	Role
1	RA2011003011112	G. SAI PRASAD	Rep/Member
2	RA2011003011100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

User interface of the project:

Our frontend of the project will consist of a website which includes application of JavaScript and CSS. The website will consist of search bar through which the user will interact with the website and obtain the recommendations based on it. The search bar will be based on python and Machine learning models which will interact with the database systems. The website will interact with the API'S from the dataset and the obtained data will be shown on the website. The admin will manage the details of the users which in turn will avoid the bots and intruders ruin our website and hence increasing the user-experience.

Testing:

The website will be tested thoroughly on various devices so that our website don't get crashed on a set of devices. The testing will be done by sharing a link to a piece of people on various devices so that we can check whether the website is working properly or not. There are various types of testing phases which will be discussed below:

Functional testing: It checks for different functions that are performed or offered by a website. It checks for different activities that a website can perform:

- It tests the JavaScript and CSS that is been used to build the website.
- It tests the forms that are present on the website.
- It checks for the links of all types present on the web page.

Compatibility Testing: Compatibility testing is done on the basis of different contexts. The compatibility testing contexts are listed below in the list:

- **Based on Browser:** It checks for the working of the website in different browsers such as Firefox, Chrome, Safari, etc.
- **Based on Operating systems:** It checks for the compatibility of the website on different operating systems like: Windows, Linux, and Mac.

Usability testing: Usability testing is the most important testing that has to be done before deploying a website for live use. This testing is performed by users or a team of clients.

- It checks the controls of the website.
- Checks the content of web pages.

- Checks whether the user requirements are satisfied or not.

Interface testing: Interface testing is done to check or verify the interface and the data flow from one system to another system. It checks for data flow from the **Web server-side** and **Database server** side.

- **Database Server:** Here it checks for the data that is transported from the database is as required or not. It measures the correctness of the data that is delivered.
- **Web server:** It will check for the webserver that it handles all the queries that are being asked by the users.

Performance testing: It checks for the performance of the website under different conditions.

- It checks for the response time for any query at different connections speed.
- It tests for the load that a website can handle under various conditions.
- It checks for the performance of the website when the load exceeds the upper limit of the maximum load.
- It checks for the recovery of the website after the breakdown of the website due to excessive load.

Result:

Thus, the testing framework/user interface framework has been created for the **car sales system**.

Executive Summary

Scope, Objective, and Approach to test the website:

Scope:

Web testing or website application testing is the testing that is done before hosting or making your website live for general use to users. Web testing is done **to find out the bugs and errors in your website that can lead to website failure in the future.**

Objective:

The main motto of testing is to find maximum defects in a software product while validating whether the program is working as per the user requirements or not. Defects should be identified as early in the test cycle as possible.

Approach:

There are 6 best approaches which we use while testing the website:

- Functionality Testing
- Compatibility Testing
- Usability Testing
- Interface Testing
- Security Testing
- Performance/Load Testing

Test Plan

Scope of Testing

Web testing or website application testing is the testing that is done before hosting or making your website live for general use to users. Web testing is done to find out the bugs and errors in your website that can lead to website failure in the future.

Non-Functional: Are all NFR (Non-Functional Requirements) covered?

Types of Testing, Methodology, Tools

Category	Methodology	Tools Required (Software)
Functional	Manual	Spira Test, Appium
Compatibility	Manual	Browser Stack
Interface	Manual	Apptim
Security	Manual	Acunetix, Appspider
Performance/Load	Manual	Performance JMeter, CloudTest
Usability	Manual	Lookback, Loop11



School of Computing

SRM IST, Kattankulathur - 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	11
Title of Experiment	Test Cases
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD, KRISHNA PRAKASH MUKUNDAAN
Register Number	RA2011003011100
Date of Experiment	12/06/2022

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
	Total	10	

Staff Signature with date

Aim

To develop the test cases manual for the <Car sales system>

Team Members:

S No	Register No	Name	Role
1	RA2011003011112	G. SAI PRASAD	Rep
2	RA2011003011100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

Test Case

Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1.	System Performances	Can system provide recommendations under 50ms at the 99.99th percentile?	User will enter the name, mail id, password	Account created	Account created	Pass	Success
2	Verify user registration	Accept valid mobile number.	1. User clicks on User Registration link 2. Enter the mobile Number on the text box 3. Click	User should be taken to the next page for entering more user details	User should be taken to the next page for entering more user details	Pass	Success

			Register button				
--	--	--	-----------------	--	--	--	--

Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Security	How a system is safeguarded against deliberate and sudden attacks from internal and external sources?	<ul style="list-style-type: none"> A password should be in encrypted format Application or System should not allow invalid users Check cookies and session time for application 	Secure	Secure	Pass	Success
2	Reliability	The extent to which any software system continuously performs the specified functions without failure.	<ul style="list-style-type: none"> How system is giving output while searching same user every time? 	Reliable output.	Reliable outcome	Pass	Success
3	Efficiency	The extent to which any software system can handles capacity, quantity and response time.	<ul style="list-style-type: none"> How the system behaves under heavy load? 	Speed of computer is fast.	Speed of computer is moderate.	Pass	Success
4	Usability	The ease with which the user can learn, operate, prepare inputs and outputs	<ul style="list-style-type: none"> Is the system easy to use? Help is provided for the users to understand the application/web site. 	Emails and phone conversations are used to provide assistance.	Emails and phone conversations are used to provide assistance.	Pass	Success

	through interaction with a system.						
--	------------------------------------	--	--	--	--	--	--

Result:

Thus, the test case manual has been created for the Chatting App.



School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	12
Title of Experiment	Provide the details of Architecture Design/Framework/Implementation
Name of the candidate	BHOGI SARANYA
Team Members	SAI PRASAD, KRISHNA PRAKASH, MUKUNDAAN SARANYA
Register Numbers	RA201100301112, RA2011003011092, RA2011003011099 RA2011003011100
Date of Experiment	19/06/2022

Mark Split Up

S. No	Description	Maximum Mark	Mark Obtained
1	Exercise	5	
2	Viva	5	
Total		10	

Staff Signature with date

Aim

To provide the details of architectural design/framework/implementation

Team Members:

S No	Register No	Name	Role
1	RA2011003011112	SAI PRASAD	Rep/Member
2	RA2011003011100	BHOGI SARANYA	Member
3	RA2011003011092	KRISHNA PRAKASH	Member

< Provide the details of architectural design/framework/implementation with screenshots - Minimum three modules to be completed (excluding login page) use of software on their choice to implement>

Full documentation with the coding

1.HOME PAGE

```

import UIKit
import FirebaseAuth
import JGProgressHUD

/// Controller that shows list of conversations
final class ConversationsViewController: UIViewController {

    private let spinner = JGProgressHUD(style: .dark)

    private var conversations = [Conversation]()

    private let tableView: UITableView = {
        let table = UITableView()
        table.isHidden = true
        table.register(ConversationTableViewCell.self,
                      forCellReuseIdentifier: ConversationTableViewCell.identifier)
        return table
    }()
```

```

private let noConversationsLabel: UILabel = {
    let label = UILabel()
    label.text = "No Conversations!"
    label.textAlignment = .center
    label.textColor = .gray
    label.font = .systemFont(ofSize: 21, weight: .medium)
    label.isHidden = true
    return label
}()

private var loginObserver: NSObjectProtocol?

override func viewDidLoad() {
    super.viewDidLoad()
    navigationItem.rightBarButtonItem = UIBarButtonItem(barButtonSystemItem:
.compose,
                                                target: self,
                                                action: #selector(didTapComposeButton)))
    view.addSubview(tableView)
    view.addSubview(noConversationsLabel)
    setupTableView()
    startListeningForCOnversations()

    loginObserver = NotificationCenter.default.addObserver(forName:
.didLogInNotification, object: nil, queue: .main, using: { [weak self] _ in
        guard let strongSelf = self else {
            return
        }

        strongSelf.startListeningForCOnversations()
    })
}

private func startListeningForCOnversations() {
    guard let email = UserDefaults.standard.value(forKey: "email") as? String else {
        return
    }

    if let observer = loginObserver {
        NotificationCenter.default.removeObserver(observer)
    }

    print("starting conversation fetch...")

    let safeEmail = DatabaseManager.safeEmail(emailAddress: email)

    DatabaseManager.shared.getAllConversations(for: safeEmail, completion: { [weak self]
result in
        switch result {
        case .success(let conversations):
            print("successfully got conversation models")
        }
    })
}

```

```

        guard !conversations.isEmpty else {
            self?.tableView.isHidden = true
            self?.noConversationsLabel.isHidden = false
            return
        }
        self?.noConversationsLabel.isHidden = true
        self?.tableView.isHidden = false
        self?.conversations = conversations

        DispatchQueue.main.async {
            self?.tableView.reloadData()
        }
    case .failure(let error):
        self?.tableView.isHidden = true
        self?.noConversationsLabel.isHidden = false
        print("failed to get convos: \(error)")
    }
})
}

@objc private func didTapComposeButton() {
    let vc = NewConversationViewController()
    vc.completion = { [weak self] result in
        guard let strongSelf = self else {
            return
        }

        let currentConversations = strongSelf.conversations

        if let targetConversation = currentConversations.first(where: {
            $0.otherUserEmail == DatabaseManager.safeEmail(emailAddress: result.email)
        }) {
            let vc = ChatViewController(with: targetConversation.otherUserEmail, id:
targetConversation.id)
            vc.isNewConversation = false
            vc.title = targetConversation.name
            vc.navigationItem.largeTitleDisplayStyle = .never
            strongSelf.navigationController?.pushViewController(vc, animated: true)
        } else {
            strongSelf.createNewConversation(result: result)
        }
    }
    let navVC = UINavigationController(rootViewController: vc)
    present(navVC, animated: true)
}

private func createNewConversation(result: SearchResult) {
    let name = result.name
    let email = DatabaseManager.safeEmail(emailAddress: result.email)
}

```

```

// check in database if conversation with these two users exists
// if it does, reuse conversation id
// otherwise use existing code

DatabaseManager.shared.conversationExists(iwth: email, completion: { [weak self]
result in
    guard let strongSelf = self else {
        return
    }
    switch result {
    case .success(let conversationId):
        let vc = ChatViewController(with: email, id: conversationId)
        vc.isNewConversation = false
        vc.title = name
        vc.navigationItem.largeTitleDisplayStyle = .never
        strongSelf.navigationController?.pushViewController(vc, animated: true)
    case .failure(_):
        let vc = ChatViewController(with: email, id: nil)
        vc.isNewConversation = true
        vc.title = name
        vc.navigationItem.largeTitleDisplayStyle = .never
        strongSelf.navigationController?.pushViewController(vc, animated: true)
    }
    })
}

override func viewDidLoad() {
    super.viewDidLoad()
    tableView.frame = view.bounds
    noConversationsLabel.frame = CGRect(x: 10,
                                         y: (view.height-100)/2,
                                         width: view.width-20,
                                         height: 100)
}

override func viewWillAppear(_ animated: Bool) {
    super.viewWillAppear(animated)
    validateAuth()
}

private func validateAuth() {
    if FirebaseAuth.Auth.auth().currentUser == nil {
        let vc = LoginViewController()
        let nav = UINavigationController(rootViewController: vc)
        nav.modalPresentationStyle = .fullScreen
        present(nav, animated: false)
    }
}

private func setupTableView() {
    tableView.delegate = self
}

```

```

        tableView.dataSource = self
    }

}

extension ConversationsViewController: UITableViewDelegate, UITableViewDataSource {

    func tableView(_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {
        return conversations.count
    }

    func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {
        let model = conversations[indexPath.row]
        let cell = tableView.dequeueReusableCell(withIdentifier: ConversationTableViewCell.identifier,
                                                for: indexPath) as! ConversationTableViewCell
        cell.configure(with: model)
        return cell
    }

    func tableView(_ tableView: UITableView, didSelectRowAt indexPath: IndexPath) {
        tableView.deselectRow(at: indexPath, animated: true)
        let model = conversations[indexPath.row]
        openConversation(model)
    }

    func openConversation(_ model: Conversation) {
        let vc = ChatViewController(with: model.otherUserEmail, id: model.id)
        vc.title = model.name
        vc.navigationItem.largeTitleDisplayMode = .never
        navigationController?.pushViewController(vc, animated: true)
    }

    func tableView(_ tableView: UITableView, heightForRowAt indexPath: IndexPath) -> CGFloat {
        return 120
    }

    func tableView(_ tableView: UITableView, editingStyleForRowAt indexPath: IndexPath) -> UITableViewCell.EditingStyle {
        return .delete
    }

    func tableView(_ tableView: UITableView, commit editingStyle: UITableViewCell.EditingStyle, forRowAt indexPath: IndexPath) {
        if editingStyle == .delete {
            // begin delete
            let conversationId = conversations[indexPath.row].id
            tableView.beginUpdates()
            self.conversations.remove(at: indexPath.row)
        }
    }
}

```

```

        tableView.deleteRows(at: [indexPath], with: .left)

        DatabaseManager.shared.deleteConversation(conversationId: conversationId,
completion: { success in
    if !success {
        // add model and row back and show error alert
    }
})

        tableView.endUpdates()
    }
}
}

```

2. SIGN IN

```

//  

// LoginViewController.swift  

// Messenger  

//  

// Created by Sai Prasad Gudari on 14/05/22.  

//  

import UIKit  

class LoginViewController: UIViewController {  

    private let scrollView: UIScrollView = {  

        let scrollView = UIScrollView()  

        scrollView.clipsToBounds = true  

        return scrollView
    }()  

    private let emailField: UITextField = {  

        let field = UITextField()  

        field.autocapitalizationType = .none  

        field.autocorrectionType = .no  

        field.returnKeyType = .continue  

        field.layer.cornerRadius = 12  

        field.layer.borderWidth = 1  

        field.layer.borderColor = UIColor.lightGray.cgColor  

        field.placeholder = "Email Address .."  

        field.leftView = UIView(frame: CGRect(x: 0, y: 0, width: 5, height: 0))  

        field.leftViewMode = .always  

        field.backgroundColor = .white  

        return field
    }()  

}
```



```

loginButton.addTarget(self,action: #selector(loginButtonTapped),for: .touchUpInside)
emailField.delegate = self
passwordField.delegate = self
// Add subviews
view.addSubview(scrollView)
scrollView.addSubview(imageView)
scrollView.addSubview(emailField)
scrollView.addSubview(passwordField)
scrollView.addSubview(loginButton)

}

override func viewDidLoad() {
    super.viewDidLoad()
    scrollView.frame = view.bounds
    let size = scrollView.width/3
    imageView.frame = CGRect(x: (view.width-size)/2, y: 20, width: size, height: size)

    emailField.frame = CGRect(x: 30, y: imageView.bottom+10, width: scrollView.width-60,
height: 52)

    passwordField.frame = CGRect(x: 30, y: emailField.bottom+10, width: scrollView.width-
60, height: 52)

    loginButton.frame = CGRect(x: 30, y: passwordField.bottom+10, width:
scrollView.width-60, height: 52)
}

@objc private func loginButtonTapped(){
    emailField.resignFirstResponder()
    passwordField.resignFirstResponder()

    guard let email = emailField.text,let password = passwordField.text,
    !email.isEmpty,!password.isEmpty,password.count >= 6 else{
        alertUserLoginError()
        return
    }
    // Firebase Log in
}
func alertUserLoginError(){
    let alert = UIAlertController(title: "Woops",message: "Please enter all information to log
in.",preferredStyle: .alert)

```

```

        alert.addAction(UIAlertAction(title:"Dismiss",style: .cancel,handler:nil))
        present(alert,animated: true)
    }
    @objc private func didTapRegister(){
        let vc = RegisterViewController()
        vc.title = "Create Account"
        navigationController?.pushViewController(vc, animated: true)
    }
}

extension LoginViewController: UITextFieldDelegate{
    func textFieldShouldReturn(_ textField: UITextField) -> Bool {

        if textField == emailField{
            passwordField.becomeFirstResponder()
        }else if textField == passwordField{
            loginButtonTapped()
        }

        return true
    }
}

```

3.SIGN UP

```

import UIKit
import FirebaseAuth
import JGProgressHUD

final class RegisterViewController: UIViewController {

    private let spinner = JGProgressHUD(style: .dark)

    private let scrollView: UIScrollView = {
        let scrollView = UIScrollView()
        scrollView.clipsToBounds = true
        return scrollView
    }()

```

```
private let imageView: UIImageView = {
    let imageView = UIImageView()
    imageView.image = UIImage(systemName: "person.circle")
    imageView.tintColor = .gray
    imageView.contentMode = .scaleAspectFit
    imageView.layer.masksToBounds = true
    imageView.layer.borderWidth = 2
    imageView.layer.borderColor = UIColor.lightGray.cgColor
    return imageView
}
```

```
private let firstNameField: UITextField = {
    let field = UITextField()
    field.autocapitalizationType = .none
    field.autocorrectionType = .no
    field.returnKeyType = .continue
    field.layer.cornerRadius = 12
    field.layer.borderWidth = 1
    field.layer.borderColor = UIColor.lightGray.cgColor
    field.placeholder = "First Name..."
    field.leftView = UIView(frame: CGRect(x: 0, y: 0, width: 5, height: 0))
    field.leftViewMode = .always
    field.backgroundColor = .secondarySystemBackground
    return field
}
```

```
private let lastNameField: UITextField = {
    let field = UITextField()
    field.autocapitalizationType = .none
    field.autocorrectionType = .no
    field.returnKeyType = .continue
    field.layer.cornerRadius = 12
    field.layer.borderWidth = 1
```

```
        field.layer.borderColor = UIColor.lightGray.cgColor
        field.placeholder = "Last Name..."
        field.leftView = UIView(frame: CGRect(x: 0, y: 0, width: 5, height: 0))
        field.leftViewMode = .always
        field.backgroundColor = .secondarySystemBackground
        return field
    }()
```

```
private let emailField: UITextField = {
    let field = UITextField()
    field.autocapitalizationType = .none
    field.autocorrectionType = .no
    field.returnKeyType = .continue
    field.layer.cornerRadius = 12
    field.layer.borderWidth = 1
    field.layer.borderColor = UIColor.lightGray.cgColor
    field.placeholder = "Email Address..."
    field.leftView = UIView(frame: CGRect(x: 0, y: 0, width: 5, height: 0))
    field.leftViewMode = .always
    field.backgroundColor = .secondarySystemBackground
    return field
}()
```

```
private let passwordField: UITextField = {
    let field = UITextField()
    field.autocapitalizationType = .none
    field.autocorrectionType = .no
    field.returnKeyType = .done
    field.layer.cornerRadius = 12
    field.layer.borderWidth = 1
    field.layer.borderColor = UIColor.lightGray.cgColor
    field.placeholder = "Password..."
    field.leftView = UIView(frame: CGRect(x: 0, y: 0, width: 5, height: 0))
    field.leftViewMode = .always
```

```
        field.backgroundColor = .secondarySystemBackground
        field.isSecureTextEntry = true
        return field
    }()

private let registerButton: UIButton = {
    let button = UIButton()
    button.setTitle("Register", for: .normal)
    button.backgroundColor = .systemGreen
    button.setTitleColor(.white, for: .normal)
    button.layer.cornerRadius = 12
    button.layer.masksToBounds = true
    button.titleLabel?.font = .systemFont(ofSize: 20, weight: .bold)
    return button
}()

override func viewDidLoad() {
    super.viewDidLoad()
    title = "Log In"
    view.backgroundColor = .systemBackground

    navigationItem.rightBarButtonItem = UIBarButtonItem(title: "Register",
                                                    style: .done,
                                                    target: self,
                                                    action: #selector(didTapRegister))

    registerButton.addTarget(self,
                           action: #selector(registerButtonTapped),
                           for: .touchUpInside)

    emailField.delegate = self
    passwordField.delegate = self

    // Add subviews
}
```



```
        width: scrollView.width-60,
        height: 52)

lastNameField.frame = CGRect(x: 30,
                             y: firstNameField.bottom+10,
                             width: scrollView.width-60,
                             height: 52)

emailField.frame = CGRect(x: 30,
                          y: lastNameField.bottom+10,
                          width: scrollView.width-60,
                          height: 52)

passwordField.frame = CGRect(x: 30,
                             y: emailField.bottom+10,
                             width: scrollView.width-60,
                             height: 52)

registerButton.frame = CGRect(x: 30,
                             y: passwordField.bottom+10,
                             width: scrollView.width-60,
                             height: 52)
```

```
}
```

```
@objc private func registerButtonTapped() {
    emailField.resignFirstResponder()
    passwordField.resignFirstResponder()
    firstNameField.resignFirstResponder()
    lastNameField.resignFirstResponder()
```

```
guard let firstName = firstNameField.text,
      let lastName = lastNameField.text,
      let email = emailField.text,
      let password = passwordField.text,
      !email.isEmpty,
      !password.isEmpty,
      !firstName.isEmpty,
```

```
!lastName.isEmpty,  
password.count >= 6 else {  
    alertUserLoginError()  
    return  
}  
  
spinner.show(in: view)  
  
// Firebase Log In  
  
DatabaseManager.shared.userExists(with: email, completion: { [weak self] exists in  
    guard let strongSelf = self else {  
        return  
    }  
  
    DispatchQueue.main.async {  
        strongSelf.spinner.dismiss()  
    }  
  
    guard !exists else {  
        // user already exists  
        strongSelf.alertUserLoginError(message: "Looks like a user account for that email  
address already exists.")  
        return  
    }  
  
    FirebaseAuth.Auth.auth().createUser(withEmail: email, password: password,  
completion: { authResult, error in  
        guard authResult != nil, error == nil else {  
            print("Error creating user")  
            return  
        }  
  
        UserDefaults.standard.setValue(email, forKey: "email")  
    }  
}
```

```

UserDefaults.standard.setValue("\(firstName) \(lastName)", forKey: "name")

let chatUser = ChatAppUser(firstName: firstName,
                           lastName: lastName,
                           emailAddress: email)

DatabaseManager.shared.insertUser(with: chatUser, completion: { success in
    if success {
        // upload image
        guard let image = strongSelf.imageView.image,
              let data = image.pngData() else {
            return
        }

        let filename = chatUser.profilePictureFileName
        StorageManager.shared.uploadProfilePicture(with: data, fileName: filename,
                                                completion: { result in
            switch result {
                case .success(let downloadUrl):
                    UserDefaults.standard.set(downloadUrl, forKey: "profile_picture_url")
                    print(downloadUrl)
                case .failure(let error):
                    print("Storage manager error: \(error)")
            }
        })
    }
})

strongSelf.navigationController?.dismiss(animated: true, completion: nil)
})

})
}
}

func alertUserLoginError(message: String = "Please enter all information to create a new account.") {

```

```
let alert = UIAlertController(title: "Woops",
                            message: message,
                            preferredStyle: .alert)
alert.addAction(UIAlertAction(title:"Dismiss",
                            style: .cancel, handler: nil))
present(alert, animated: true)
}

@objc private func didTapRegister() {
let vc = RegisterViewController()
vc.title = "Create Account"
navigationController?.pushViewController(vc, animated: true)
}

extension RegisterViewController: UITextFieldDelegate {

func textFieldShouldReturn(_ textField: UITextField) -> Bool {

if textField == emailField {
    passwordField.becomeFirstResponder()
}
else if textField == passwordField {
    registerButtonTapped()
}

return true
}

}

extension RegisterViewController: UIImagePickerControllerDelegate,
UINavigationControllerDelegate {
```

```
func presentPhotoActionSheet() {  
    let actionSheet = UIAlertController(title: "Profile Picture",  
                                       message: "How would you like to select a picture?",  
                                       preferredStyle: .actionSheet)  
    actionSheet.addAction(UIAlertAction(title: "Cancel",  
                                       style: .cancel,  
                                       handler: nil))  
    actionSheet.addAction(UIAlertAction(title: "Take Photo",  
                                       style: .default,  
                                       handler: { [weak self] _ in  
                                           self?.presentCamera()  
                                       }))  
    actionSheet.addAction(UIAlertAction(title: "Chose Photo",  
                                       style: .default,  
                                       handler: { [weak self] _ in  
                                           self?.presentPhotoPicker()  
                                       }))  
    present(actionSheet, animated: true)  
}  
  
func presentCamera() {  
    let vc = UIImagePickerController()  
    vc.sourceType = .camera  
    vc.delegate = self  
    vc.allowsEditing = true  
    present(vc, animated: true)  
}
```

```
func presentPhotoPicker() {
    let vc = UIImagePickerController()
    vc.sourceType = .photoLibrary
    vc.delegate = self
    vc.allowsEditing = true
    present(vc, animated: true)
}

func imagePickerController(_ picker: UIImagePickerController,
didFinishPickingMediaWithInfo info: [UIImagePickerController.InfoKey : Any]) {
    picker.dismiss(animated: true, completion: nil)
    guard let selectedImage = info[UIImagePickerController.InfoKey.editedImage] as?
    UIImage else {
        return
    }

    self.imageView.image = selectedImage
}

func imagePickerControllerDidCancel(_ picker: UIImagePickerController) {
    picker.dismiss(animated: true, completion: nil)
}

}
```

11:46



Phone Number or Email

Password

LOG IN

CREATE NEW ACCOUNT

You will not be creating a Facebook profile

FORGOT PASSWORD



Result:

Thus, the details of architectural design/framework/implementation along with the screenshots were provided.