Bachelor in Computer Application

(BCAOL)

Project Synopsis

(December 2024)

On

Chat Web Application

To

Indira Gandhi National Open University

(School of Computer and Information Sciences)

(1st Floor, DEC Building, IGNOU, Maidan Garhi, New Delhi -110068.)

Submitted by

Enrollment no. 2250319870

Name: -Hardik Shiromani

Under the guidance of: Versha Matre

SCHOOL OF COMPUTER AND INFORMATION SCIENCES
IGNOU, MAIDAN GARHI, NEW DELHI – 110 068

II. PROFORMA OF BCA PROJECT PROPOSAL (BCSP-064)

Enrolment No.: 2250319870 Regional Centre Code: 15 Study Centre: IGNOU study centre 1500

1. **Name of Student: Hardik Shiromani**

2. **Address of the student: 39/3 Bhagat Singh Marg, Devta colony Ujjain M.P**

3.(a) **E-mail: sangeetashiromani28@gmail.com** 3.(b) **Telephone/ Mobile No.:7648985553**

4. **Title of the Project : ChatWorld The Chat Web App**

5. **Name of Project Guide: Versha Matre** 5.(b) **Designation of Project Guide: Lecturer**

6. **Address of Project Guide: 138,Mangal Murti Krishnaji Nagar,Scheme no.77,behind Mayur Hospital ,Indore,Madhya Pradesh -452001**

7. **Qualification of the Guide*** (Compulsory to Attach bio-data of Guide)

Ph.D.	M.Tech.	B.Tech.	MCA	Any other
☑	☐	☐	☐	☐

***Note :** i. All the above mentioned Degrees must have been awarded in Computer Science/IT only ii. A Guide should not guide more than 8 students of BCAOL at any point of time.*

8. **Industrial / Teaching experience of the Guide (in Years): 14 years**

9. **Software Used for this Project: React js,Node js,Express js,Socket IO,HTML,CSS,Javascript,Bootstrap**

***Note :** 1. Use of Visual Basic and MS-Access as Front End and Back End respectively is forbidden. But, you are permitted to use Visual Basic with other Software. Also, you can use MS-Access with other software.*
2. Use of C or C++ Programming Language for Project Related to Database Management is strictly forbidden.

Signature of the Student
Date: 31/12/2024

Signature of the Guide
Date: 31/12/2024

Important:
1. Attach this Proforma along with Guide"s Biodata and Project Synopsis in the Project Report.
2. Not more than one student is permitted to work on a project.
3. Complete project as per the comments of Synopsis evaluator, then only submit your Project Report.

For Office Use Only

☐ Approved ☐ Not approved

……
Signature, Designation, Stamp of the Project Proposal Evaluator
Date: …….

Suggestions for reformulating the Project:
1.
2.
3.
4.

भारत सरकार
Government of India

वर्षा मात्रे
Versha Matre
जन्म तिथि/DOB: 13/06/1986
महिला/ FEMALE

4494 7474 3294

मेरा आधार, मेरी पहचान

भारतीय विशिष्ट पहचान प्राधिकरण
Unique Identification Authority of India

पता:
आत्मजा: रमेशचंद्र मात्रे, 138, मंगल मूर्ति कृष्णाजी नगर, स्कीम न. 77, मयूर अस्पताल के पीछे, इंदौर, इंदौर, मध्य प्रदेश - 452001

Address:
D/O: Rameshchandra Matre, 138, Mangal Murti Krishnaji Nagar, Scheme No. 77, Behind Mayur Hospital, Indore, Indore, Madhya Pradesh - 452001

QR Code with Photograph

4494 7474 3294

1947 help@uidai.gov.in www.uidai.gov.in

Versha Matre

138, Sch. No.-77 near Mayur
Hospital, Mangal murti Krishnaji
Nagar, Indore MP-452016
Email: versha.matre@gmail.com
Mob: +91-8982702474

CAREER OBJECTIVE

Achieving a challenging position in the organization, where I can exploit my knowledge and skills in advancement of th organization.

EDUCATIONAL RECORD

- **Ph.D (CSE)** (JUNE 2024)from Abdul Kalam University ,Indore
- **M.E (Computer Science & Engineering)**(2012-14) from ,S D Bansal College of Engineering, Indore (M.P.)(RGPV University) 74%

- **B.E (CSE)** (2012) from Chameli Devi School of Engineering, Indore (M.P.) (RGPV University) With Aggregate marks **64.84%**
- Diploma (CSE) (2009) from Govt. Women's Poly. College, Indore with **56.50 %.**
- **10th**from Govt. Girls H.S.School, Barwani (MP Board) with **45%.**

COMPUTER SKILLS

- Professional Skill Set : C, C++, Core Java, .Net, Android, php
- Subjects of Interest: Computer Network.

LANGUAGES KNOWN

- English (read, write & speak)
- Hindi (read, write & speak)

PROJECTS

Title: - "Query through SMS"
Title: - "Compiler Design "
Title: - "Hospital Management"

PERSONAL PROFILE

Name	: Versha Matre
Father's name	: R.C.Matre
Gender	: Female
Date of Birth	: 13 Jun, 1986
Hobbies	: Listening to Music, Reading news paper
Strengths	: Optimistic, Adaptable.

WORK EXPERIENCE

Place : Govt. poly. College, sanawad
Post : Lecturer 14 years

DECLARATION

I hereby declare that the information given above is true to the best of my knowledge and belief.

Place: Indore

Versha Matre

Title of the Project

ChatWorld (The chat web application)

Table of Contents

Introduction

In today's world of advanced communication, instant messaging applications have become a cornerstone in connecting people and streamlining communication. Traditional methods of communication often suffer from delays, limited accessibility, and inefficiency in handling data. Managing user interactions, message storage, and secure communication through conventional systems can be challenging, with issues such as lack of security, inconsistency, and reliability. These challenges increase the probability of data loss, resulting in compromised user experiences.

To address these issues, we developed a modern web-based Chat Application that ensures seamless, real-time communication for its users. This application leverages a robust architecture designed to provide secure, reliable, and efficient messaging features. By overcoming the limitations of traditional systems, this chat application facilitates instant communication, user authentication, and message encryption.

The proposed Chat Application enables users to send messages, share files, and create group conversations. It operates on an Internet-based platform, ensuring access across devices via a web interface. The system ensures data integrity and high performance by incorporating advanced backend technologies like Node.js and databases such as MongoDB/MySQL. This application provides a secure, scalable, and user-friendly solution for both personal and professional communication needs.

The application also features admin tools for managing users and monitoring conversations, ensuring a secure and controlled environment. It is accessible

through a web interface, making it compatible across various devices, and can be scaled for both small groups and large organizations.

Objectives: -

Facilitate Real-Time Communication
Enable users to exchange messages instantly, ensuring seamless and uninterrupted communication.

Ensure Data Security and Privacy
Protect user information and messages through robust encryption and secure authentication mechanisms.

Provide Multi-Platform Compatibility:-Design the web application to be accessible across various devices and browsers, offering a consistent user experience.

Offer User Personalization
Allow users to create personalized profiles with options to upload photos, set bios, and customize themes.

Foster User Engagement
Support features like group chats, media sharing, and interactive chat options to enhance the communication experience.

Optimize Performance in Low-Network Conditions
Ensure reliable message delivery and syncing, even in areas with weak network connectivity.

Enhance Accessibility
Create a responsive and intuitive interface that is easy to navigate, catering to users of all technical skill levels.

Implement Advanced Privacy Features
Include tools such as password-protected access, incognito modes, and hidden chats to ensure user activity remains private.

Tools/Platform of Project Hardware and Software Requirement Specifications

Hardware Requirements: -

- **Processor: - i3 or higher**
- **RAM: - 4GB minimum**
- **Storage: - 256 SSD or higher**
- **Operating System: - Windows 11**
- **Keyboard**
- **Mouse**

Software Requirements: -

The software requirements deal with defining software resource requirements and prerequisites that needs to be installed on a computer to provide optimal functioning of an application. Here are the key requirements for making software application:

Operating System: Windows, Linux or macOS

Nodejs: The latest version must be installed on the development machine

1. **Package Manager**: NPM (Node Package Manager) comes with Node.js and is used to install project dependencies and libraries for managing software.
2. **Web Browser:** The latest versions of Chrome, Firefox, Safari, or Edge for testing and debugging purposes.
3. **Visual Studio Code:** To write the code
4. **MySQL:** The latest must be installed in the machine

Tools/Platform / Services

For the Development of the chat application the Tools and Platforms can be used: -

- **React: -** A widely used Front end Framework to develop Dynamic UI user interfaces
- **HTML/CSS/JavaScript:** These are the three core technologies used in web development. HTML is the backbone which gives structure and content to webpage. CSS is used for styling and formatting content. JavaScript is dynamic and high-level programming language.
- **Node.js:** It is an open-source, cross-platform runtime environment that allows developers to execute JavaScript code outside of a web browser. It will as a backend
- **Express.js:** A web application framework for Node.js that helps in building scalable web applications through APIs.
- **Socket IO:** Socket.IO is a popular JavaScript library used for real-time, bidirectional communication between web clients (like browsers) and servers. It's often used to build real-time applications such as chat apps, online games, and live notifications.
- **Bootstrap:** It is used for styling, open-source front-end development framework for the creation of websites and web apps.
- **Firebase:** It is used for the User Authentication through mobile number or email ID.

Project Category

The project falls under the Web Application Development category because it involves building a chat application that operates on the web.

DFD (Data Flow Diagram)

A data flow diagram is a graphical representation that depicts information flow and transforms that are applied as data move from input from output. The basic form of a data flow diagram may be used to represent a system or software at any level of abstraction. It may be partitions into levels that represent increasing information flow and functional details. Therefore, the DFD's provide a mechanism for functions; modelling as well as information flow modeling.

Data flow diagram is portioned into levels like 0,1,2 and so on. 0 level DFD's also called a functional system model or a context level that represent the entire software elements as a single bubble with input and output data indicated by incoming and outgoing arrows respectively some guide line for creating DFD's A few simple guidelines can add immeasurably during derivation of data flow Diagram.

1. The Level 0 DFD's Should depict the software system as a single bubble.
2. Primary Inputs and outputs should be carefully noted.
3. Refinement should be beginning by isolating candidate process data object and 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.
6. One bubble at a time should be refined.

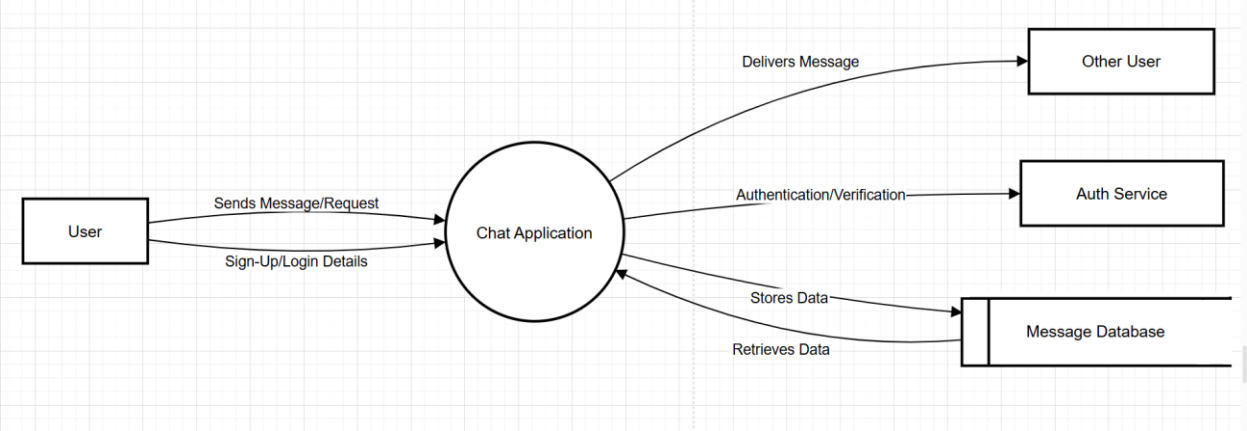
All symbols and their indication

External entity

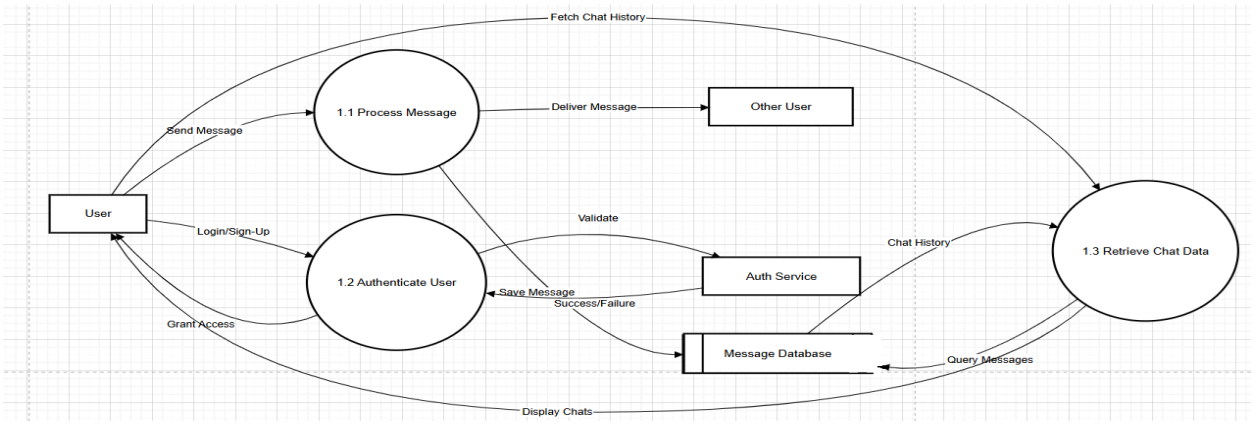
Process

dataflow

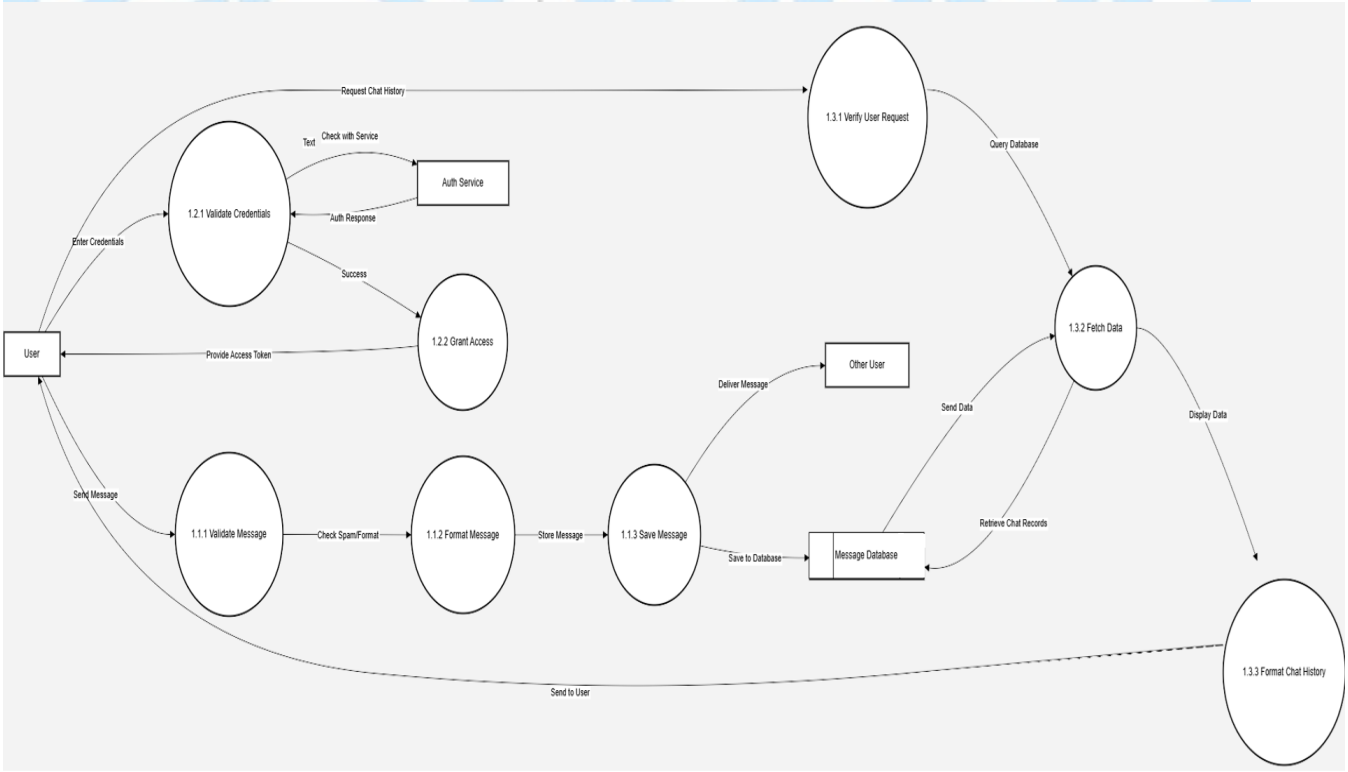
0 level DFD: -



1level DFD


Fetch Chat History
1.1 Process Message
Deliver Message
Other User
Send Message
User
Login/Sign-Up
Validate
1.2 Authenticate User
Auth Service
Chat History
1.3 Retrieve Chat Data
Save Message
Success/Failure
Grant Access
Message Database
Query Messages
Display Chats

2 level DFD


Request Chat History
1.3.1 Verify User Request
Query Database
Text
Check with Service
Auth Service
1.2.1 Validate Credentials
Auth Response
Enter Credentials
Success
1.3.2 Fetch Data
User
Provide Access Token
1.2.2 Grant Access
Other User
Deliver Message
Send Data
Display Data
Send Message
1.1.1 Validate Message
Check Spam/Format
1.1.2 Format Message
Store Message
1.1.3 Save Message
Save to Database
Message Database
Retrieve Chat Records
1.3.3 Format Chat History
Send to User

Entity relationships Diagram (ERD)

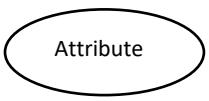
An Entity-Relationship Diagram (ER Diagram) is a high-level, logical data model used to describe and visualize the structure of a database. It is an abstraction that illustrates how data is organized and how relationships between different pieces of data are structured within a system.

ER diagrams serve as a blueprint for database design and are particularly useful for analyzing data requirements and designing databases systematically.

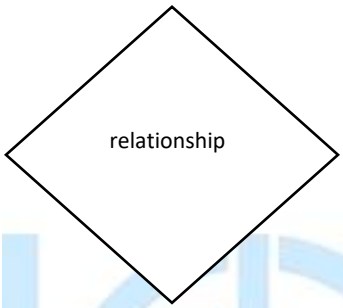
Components of the ER Diagram: -

1.Entity: Entity is a fundamental concept that represents a real-world object, person, place, event, or concept about which data can be collected and stored in a database. Entities are the "nouns" of the system being modeled, and they form the backbone of an ER diagram. The entities represent with a rectangle.

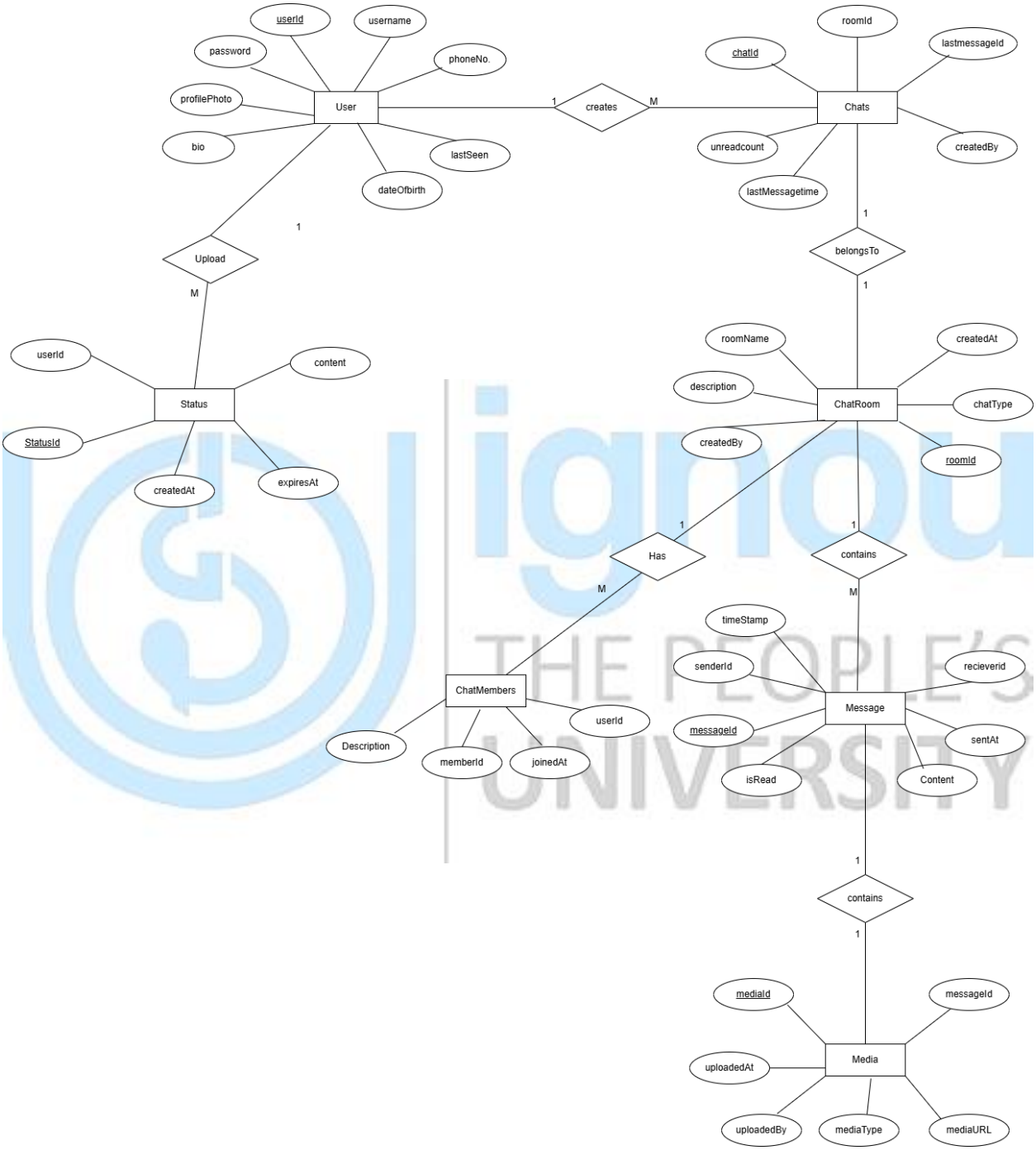
2.Attribute: An attribute is a characteristic or property of an entity or relationship in an Entity-Relationship Diagram (ERD). It defines specific details or information about the entity or relationship it describes. Attributes store the actual data values that define or identify an entity or relationship in a database. It is represented by eclipse.



3. Relationship: Relationship in an Entity-Relationship Diagram (ERD) represents the logical association or connection between two or more entities. It models how entities interact with one another in the system being designed. Relationships are fundamental to understanding the structure and behavior of a database. It is represented by diagonal.



ER Diagram: -


userId
username
password
phoneNo.
profilePhoto
User
bio
lastSeen
dateOfbirth
1
creates
M
roomId
lastmessageId
chatId
Chats
unreadcount
createdBy
lastMessagetime
1
belongsTo
1
Upload
M
userId
content
Status
StatusId
createdAt
expiresAt
roomName
createdAt
description
ChatRoom
chatType
createdBy
roomId
1
Has
1
contains
M
M
timeStamp
senderId
recieverid
ChatMembers
Message
userId
messageId
sentAt
Description
memberId
joinedAt
isRead
Content
1
contains
1
mediaId
messageId
uploadedAt
Media
uploadedBy
mediaType
mediaURL

Database Design: -

Table: User

Field	Data Type	Constraint
userId	Varchar (255)	Unique ,Primary
username	VARCHAR (255)	NotNull
phoneNumber	VARCHAR (255)	UNIQUE, NOT NULL
password	VARCHAR (255)	NOT NULL
profilePhoto	VARCHAR (255)	NULL
bio	TEXT	NULL
dateOfbirth	DATE	NULL
lastSeen	DateTime	NULL

Table: Chatroom

Field	Data Type	Constraint
roomId	Int (10)	Primary Key
roomName	Varchar (255)	NotNull
createdAt	Datetime	NotNull
chatType	ENUM	NotNull
description	TEXT	Null
createdBy	Int	NotNull
createdAt	DateTIme	NotNull

Table: ChatMembers

Field	Data Type	Constraint
Description	Text	Null
memberId	Int (10)	Primary Key
userId	Int (10)	Foreign Key References User(userId)

joinedAt	DateTime	Null

Table: Message

Field	Data Type	Constraint
messageId	Int (10)	Primary key
senderId	Int (10)	FOREIGN KEY REFERENCES User(userId)
recieverId	Int (10)	FOREIGN KEY REFERENCES User(userId)
content	TEXT	NOT Null
sentAt	DateTime	NOT Null
isRead	BOOLEAN	Default False
type	Varchar(10)	Optional
roomId	Int (10)	FOREIGN KEY REFERENCES User(userId)

Table: ChatsTable

Field	Data Type	Constraint
chatId	Int (10)	Unique
roomId	Int (10)	Foreign Key References ChatRoom(roomId)
lastMessageId	Int (10)	Foreign Key References Message(messageId)
lastMessagetime	DateTime	NotNull
unreadCount	Int	Optional

Table: Status

Field	Data Type	Constraint
statusId	Int(10)	Primary key

userId	**Int(10)**	**Foreign Key References User (userId)**
content	**Text**	**Null**
createdAt	**Datetime**	**Not Null**
expiresAt	**Datetime**	**Null**

Table: Media

Field	Data Type	Constraint
mediaId	**Int (10)**	**Primary Key**
messageId	**Int (10)**	**Foreign Key References Message(messageId)**
mediaURL	**Varchar (255)**	**NOT NULL**
fileSize	**Int (10)**	**Null**
mediaType	**Enum ('image', 'video', 'audio', 'file')**	**NOT NULL**
uploadedAt	**Datetime**	**NOT NULL**
uploadedBy	**Datetime**	**Not Null**

Number of Modules and their Description:

1.User Registration and Authentication: -It allows the user to sign in before creating the user profile on the application. It handles user sign in and sign-up processes using mobile number.

It includes processes: -

- User registration and Login.
- Password reset/recovery via email or OTP.

3.**Chat Module (Individual and Group Chats)**: - It allow user to manage their chats and handles the core messaging functionality for both individual users and groups. It includes: -

3.1 Individual Chat:-

- Real -time one to one messaging.
- Message delivery Status (sent, delivered, read).
- Media Sharing (images, videos, documents, audio).
- Message timestamps.

3.2 Group Chat :-

- Create and manage chat groups.
- Add or remove members from groups
- Assign roles (admin/manager/members) within groups.
- Real -time group messaging with delivery /read receipts.
- Group notification for events.
- Media sharing and collaborative interactions within the group.

Advanced Features:-

- Pin important messages in group chats for quick reference.
- Change group name and profile picture (admin privilege).

- Configure group privacy (invite-only, public, etc.).

4 Status Module

The Status Module allows users to share temporary updates (text, images, or videos) with their contacts or groups. These statuses are time-bound, typically expiring after 24 hours. The module improves user engagement by providing a platform for quick, dynamic sharing.

5.**Contact Management modules**: -It allows the user to see their contacts on the app and start chat with them. It includes

- Viewing a list of active users.
- The list the list of contacts from user's devices.

5. **Notifications**: -The Notifications Module ensures users stay informed about important events in real time, such as new messages, friend requests, or group activity. It improves user engagement and enhances the overall app experience.
   It includes: -

5.1 Real-time Notifications: -

- Alert for new Messages (both individual or group)
- Updates on group activity, such as new members or role changes.

5.2 Customizable Notification Settings:-

- Users can enable/disable notifications for:
- All chats or specific chats/groups.

5.3 Sounds and vibrations for messages:-

- Push notifications or in-app alerts.
- Do Not Disturb (DND) mode: Silences all notifications for a specific time.

5.4 Actionable Notifications:- The Users can reply to messages directly from the notification (push or in-app). It also have Quick actions like marking messages as read or muting chats/groups from the notification.

6.Search Module: - The Search Module provides users with a powerful and efficient way to find specific content within the app, such as users, chats, or messages. It simplifies navigation and enhances user experience by offering quick and organized search results.

It includes features: -

6.1 User Search: - It allows the user to search and start chat with another user with their Usernames.

6.2 Chat Search: - It allows the search for active or archived chats by their contact names and group name and directly open the chat from the search results.

6.3Search History: - It Saves Recent Searches for Quick Access and allows users to clear their search history.

7.Settings: - The Settings Module allows users to customize their app experience, manage account preferences, and control privacy and notification settings. It acts as a centralized hub for user configuration.

It includes features: -

7.1 Account Settings:-

- Profile Management: - Allows the user to update profile Picture, Username, bio or Email And edit personal details like date of birth or phone number.
- Change Password: - Option to reset or update the account password.

7.2 Account Security:-

- Enable Two-Factor Authentication (2FA) and View active sessions and logout remotely.

- Delete/Deactivate Account: Temporary deactivation or permanent deletion options.

7.3 Privacy Control :-

- Choose who can see the user's profile Picture, status, or last seen (Everyone, Contacts Only, Nobody).
- Enable/disable read receipts.
- Block/unblock specific users.
- Toggle the display of Online or Last Seen.

7.4 Chat Theme customization: -

- Light and dark.
- Adjust font style and size for better readability.

7.5 Storage Usage:-

- Display storage consumed by chats, groups, and media.
- Auto-download settings for images, videos, and documents (Wi-Fi only, data, or never).
- Clear Chat Data.
- Delete specific chat histories or clear all conversations

7.6 Backup Options:-

- Enable cloud or local backup for messages and media.
- Set automatic backup schedules (daily, weekly, or manual).

8.Help and Support: -

- Access FAQs and troubleshooting guides.
- Contact customer support via email or in-app chat.
- Provide feedback or report bugs directly from the app.

9.App Version and Updates

- Display current app version.
- Check for updates manually if auto-update isn't enabled.

Process Logic

1. **User Registration Process :-**This is the first page and module of the application the user interacts with it and starts the registration process. Steps:

   Input Collection:

   - User enters a mobile number.

   Mobile Number Validation:

   - Check if the entered mobile number is in a valid format.
   - Query the database to ensure the number is not already registered.

   OTP Generation and Verification:

   - Generate a one-time password (OTP) and send it to the user's mobile number via SMS.
   - Prompt the user to enter the OTP.
   - Validate the OTP within a time limit (e.g., 5 minutes).

   Account Creation:

   - If OTP is valid, save the user details (mobile number, hashed password, etc.) in the database.
   - Then ask the user to create a unique Username and Password.
   - After it Return a success registration message to user And Now the user can login on app.

1.1User Login Process: After the successful registration the user sees a form Of the user name and password where user can enters his unique username and password.

1.3 Password Validation: If the password is invalid then return an error message .

1.4. Password Reset/Recovery Process: User selects Forgot Password and an OTP sends on his number that he registered. It Generate a one-time password (OTP) and send it via SMS. If OTP is valid, allow the user to enter a new password. Hash the new password and update it in the database. And return a success message.

2. **User Profile Management:** After the successful login the user sees the page of the adding a profile picture, bio, and date of birth and email id (for chat backups)
3. **Chat Module (Individual and Group):-**this is the main page of the chat application where user can create and manage chats with individual and groups.

3.1 Process of Individual Chat

Real-Time Messaging:

Logic:

- Establish a socket connection (e.g., using WebSocket or libraries like Socket.IO).
- When a user sends a message, it is transmitted to the server and forwarded to the recipient.

Key Steps:

- Sender types and sends a message.
- Server verifies the recipient's online status.
- If the recipient is online:
- Deliver the message in real-time.
- Send a "delivered" status back to the sender.
- If the recipient is offline:
- Save the message in the database for delivery once the recipient comes online.

Message Delivery Status:

- Sent: Message successfully sent to the server.
- Delivered: Message received by the recipient's device.
- Read: Recipient opens and reads the message.

- Use event listeners to update the message status in real-time.

Media Sharing:

Logic:

- Users upload media (images, videos, audio, documents) to the server.
- The server generates a secure URL for the media file and sends it to the recipient.

Steps:

- User selects a media file and uploads it.
- Server stores the file securely (e.g., in cloud storage).
- Server sends the file URL to the recipient.
- Ensure media is compressed or optimized to save bandwidth.

Timestamps:

- Include the time when the message was sent, delivered, and read.
- Maintain consistency using a standardized format (e.g., ISO 8601).

3.2 Process of Group Chat:-

Create and Manage Groups:

Logic:

- Users can create a group by providing a group name, description, and profile picture.
- Assign roles (e.g., admin, manager, or member) during group creation or later.

Steps:

- Creator sends a request to create a group.
- Server generates a unique group ID and stores group details.
- Admins can invite or remove members as needed.

Real-Time Messaging:

- Messages sent in a group are broadcast to all group members.
- Delivery and read receipts are handled for each group member individually.
- Use socket connections to broadcast updates in real-time.

Roles and Permissions:

Admin Privileges:

- Add/remove members.
- Change group name and profile picture.
- Pin messages for everyone in the group.

Member Privileges:

- Send/receive messages and media.
- Role-based permissions ensure proper management.

Group Notifications:

- Notify all group members about events like:
- New member added or existing member removed.
- Group name/profile picture updated.
- Notifications can be sent via push notifications or in-chat messages.

Media Sharing:

- Similar to individual chats, but media is visible to all group members.
- Media files are uploaded to the server, and links are shared within the group chat.

3.3 Advanced Features

Pin Important Messages:

Logic:

- Allow group admins to mark certain messages as "pinned."
- Store pinned messages in a separate database field for quick retrieval.

Use Case:

- Highlight meeting schedules, important updates, or shared documents.

Steps:

- Admin selects a message and marks it as pinned.
- Message is moved to the top of the group chat under a "Pinned Messages" section.

3.4 Change Group Name and Profile Picture:

- **Logic:**
  - Group admins can edit group details.
  - Changes are synced with all group members.
- **Steps:**
  - Admin initiates the change request.
  - Server updates the group details in the database.
  - Notify all group members about the change.

3.5 Configure Group Privacy:

- **Logic:**
  - Group creators can select privacy settings:
  - Invite-link: Members can join only through an invitation link from an admin.
  - Public: Any user can find and join the group.
  - Store privacy settings in the database.
- **Steps:**

- During group creation, privacy settings are configured.
- Based on the settings:
- Public groups are displayed in group search results.
- Invite-only groups require an admin to approve joining requests.

4. **Contact Management modules**: - with this module the user can view the contacts from his device available on this app . The user can see the username profilePhoto and bio of the user from his contacts. When the user clicks on their name then he can start conversation with them.

5. **Notifications:- It will use to receive notification of replies to the user and also in group.**

Process Description:

5.1Alerts for New Messages:

For both individual and group chats:

- When a new message is received:
- The server sends a push notification to the recipient(s) using services like Firebase Cloud Messaging (FCM) or Apple Push Notification Service (APNS).
- Notification includes the sender's name, message preview, and timestamp.
- The user can reply on the notifications and send it to receiver..

If the app is in use:

Trigger an in-app alert using WebSocket for real-time updates.

5.2 Updates on Group Activities:

- **For events like:**
  - New members joining a group.
  - Admin role changes or group name updates.

- Server sends a notification to all group members to inform them of the event.

6.Search module:- this module will represented as the search bar above the chats section on the app . with the help of search the user can the search and find people with their username . the user can also find the groups and the files uploaded on the group or on a specific chat. the search by the user are saved in the search suggestions for the quick access.

7.Settings: This tab will represented as three dots on the top right corner of the chat tab. This module will used to change and customize the user and privacy settings and also multiple settings. It includes 6 sections:-

7.1Account Settings:-

Profile Management: User can update his profile details by uploads a new profile picture, updates their username, bio, email, or other personal details. System updates the User Table with new values .

Account Security: User can enable Two factor Authentication and manage, delete, view active sessions.

Process of 2FA:

- User enables 2FA and chooses a method (e.g., OTP via email or SMS).
- On login, the system generates and sends an OTP to the user.
- User must enter the OTP to access their account.
- OTP validation occurs in the backend.

7.2 Chat Theme: This tab contains only two theme for the app light and dark which the user can set from this option.

7.3 manage the fonts: This option helps the user the change the font size of the whole application according to his readability . this contains a slider for the increasing and decreasing the font size.

7.4 Storage Usage : this tab helps the user to view and analyse and manage his storage stats and helps to manage the app storage.

- **Process:**
  1. Storage usage for media files is calculated:
  2. Users can clear specific chats.
  3. Enable and disable Auto download settings of media in individual and group chats .

7.5 Backup Options:-the user can do backups of the chats and media files and adding an email id to it.after Registering and email id the user can set a backup timer in a week or in a month

8.Status: This tab is used to upload and share videos,photos, images and text to the contacts of user for 24 hours only.

8.1Adding a Status

- **Steps**:
  1. **Input**: User uploads content (text, image, video).
  2. **Validation**:
     - Check file size and format for images/videos.
     - Restrict text length to a predefined limit.
  3. **Save to Database**:
     - Save the content with a timestamp in the Status table.

8,2Viewing a Status

Steps:

1. **Fetch Status**: Retrieve statuses from the Status table for contacts or groups visible to the user.

2 **Display Order**: Sort statuses by createdAt timestamp.

3 Update View Count: Increment the view count for the status being watched

8.3Deleting a Status

Steps:

1. **Identify Status**: User selects the status to delete.
2. **Automatic Expiration:**A scheduled task runs periodically (e.g., hourly or daily) to remove expired statuses.

Testing

Black box testing: It is a software testing technique used to evaluate the functionality of a system without requiring any knowledge of its internal code, architecture, or logic. The primary focus is on examining the software's external behavior based on its requirements and specifications.

In black box testing, testers provide various inputs to the application and analyze the outputs to verify that the system performs as expected. This method treats the software as a "black box," meaning that the inner workings of the system are hidden from the tester. Instead, testing is conducted purely based on the system's interface and functional specifications.

White Box Testing: White box testing for a chat app involves examining the internal code structure, logic, and algorithms to ensure that all components of the app work as intended. Since the focus is on the internal workings of the application, we evaluate modules like message transmission, database queries, authentication, and encryption.

It allows testers to verify:

- All paths through the code are executed correctly.
- Each line of code performs its intended function.
- Data flows smoothly and accurately through the system.

White box testing is often used to:

7 Detect logical errors in the code.
8 Optimize the performance of algorithms.
9 Identify vulnerabilities, such as security flaws, early in development.

Integration Testing

Integration testing focuses on evaluating how well different modules or components of an application work together. integration testing ensures that individual components such as message sending, user authentication, database interaction, and notification systems properly interact with each other and work as expected when combined.

Benefits:

1. **Detects Issues Between Modules:**
   Verifies that different component (e.g., user authentication, messaging, notifications) work together as expected, ensuring smooth interaction.
2. **Improves System Reliability:**
   Ensures that the combined functionality of multiple components doesn't result in issues such as data inconsistencies, broken features, or crashes.
3. **Validates End-to-End Scenarios:**
   It allows for testing user journeys, ensuring that real-life scenarios (like logging in, sending messages, receiving notifications) function correctly when integrated.
4. **Ensures Data Consistency:**
   Verifies that data is stored, retrieved, and synchronized correctly across multiple modules, such as databases, message storage, and notifications.

Types of Report Generation

Performance Reports

- Response Time: Average time taken for messages to be delivered.
- Server Downtime: Details about app or server outages.
- API Usage: Metrics for backend API performance and usage.

User Interaction Reports

- Chat Frequency: Average number of chats per user.
- Most Active Users: Users with the highest activity levels.
- Friend/Contact Requests: Number of sent and accepted requests.

Security and Compliance Reports

- Login Attempts: Successful and failed login attempts by users.
- Blocked Users: List of users who have been blocked for inappropriate behavior.
- Message Encryption Logs: Details on encryption status for messages.

Feedback and Support Reports

- Customer Support Logs: Types and frequency of user queries or complaints.
- Feedback Analysis: Common issues or feature requests submitted by users.

Future scope and further enhancement of the project

The chat world has a vast scope which encompass various features, functionalities, and use cases.

Customizable UI:

Theme Selection: Allows users to choose predefined theme and also provide an option for custom theme creation by selecting background colors, font styles, and chat bubble designs.

Chat Bubble Customization: Allow users to adjust the shape, size, and colors of chat bubbles .Enable different styles for sent and received messages (e.g., rounded, square, or shadowed).

Voice and Video Calls:

Real-Time Voice and Video Communication

- Allow users to initiate high-quality voice and video calls directly from the chat interface.

Group Call Functionality

- Enable group voice and video calls with participants, ideal for team meetings, family gatherings, or virtual events.

Advanced AI and Automation

- **AI-Powered Chatbots:** Integrate AI-driven chatbots for customer support, FAQs, or personalized interactions.
- **Natural Language Processing (NLP):** Enable features like sentiment analysis, language translation, and context-based message suggestions.

- **Smart Replies:** Suggest quick responses based on the context of the conversation.

Security and Privacy Upgrades

- **Zero-Knowledge Encryption:** Ensure that even the app developers cannot access user messages.
- **Privacy-Centric Features:** Introduce self-destructing messages, private chat modes, or anonymous chatting.

Validation to be performed

1.Mobile Number Validation: The mobile number plays and important role in user verification So it should be in 10 digits format and it should be unique and not already be registered on the database.

2.Username validation: The username of a user should be unique and it's not repeatable the user can edit it after login but user can't make 2 usernames on one number.

3.User Privacy and Consent Validation: Ensure that users agree to terms of service and privacy policies before creating an account. Give users control over what personal data they share and let them opt-out of certain features if desired.

4. Session Management Validation: Limit the number of devices that can be logged into a single account at the same time, or allow users to log out from all devices.

Limitations for the project

1.Network Connectivity Issues: A Poor or intermittent internet connectivity can disrupt chat communication, especially on mobile devices. Slow or unreliable network conditions can prevent messages from being delivered instantly, leading to a bad user experience.

2.Legal and Regulatory Compliance: different regions have different laws governing data privacy, user consent, and content moderation, and the chat app must comply with these regulations.

3.Handling Large Attachments: Users often send large files like images, videos, and documents. Handling these files can be challenging for storage, upload speed, and app performance. For this there is must be a limit of the file size such as 100 MB only can be on chat.

4.Scability Limitations: As more users join the app and the volume of messages increases, the app must handle higher traffic and data storage. This becomes challenging for both the backend infrastructure and database.

5. Security Vulnerabilities: Chat apps are vulnerable to various security threats such as data breaches, hacking attempts, or unauthorized access. To solve this, implement end-to-end encryption (E2EE) for private chats, secure APIs with OAuth 2.0, hash passwords using secure algorithms like bcrypt, and ensure secure transmission with HTTPS. Also, monitor for potential vulnerabilities and conduct regular security audits.

Bibliography

Books:-

1. IGNOU blocks of System Analysis And Design
2. IGNOU blocks of Introduction To Database Management Systems.
3. Web Development with Node and Express(Author :Ethan Brown)

Websites:

1. https://react.dev/
2. https://firebase.google.com/
3. https://nodejs.org/en
4. https://socket.io/

Are you doing this project for any Industry/Client? Mention Yes/No. If Yes, Mention the Name and Address of the Industry or Client

Ans. NO