### 

### **V CHAT**

#### **A PROJECT REPORT**

##### ***Submitted by***

**Aditya Vishwanathan** (2115000082)

**Abhishek Kushwaha** (2115000022)

***in partial fulfillment for the award of the degree of***

#### **BACHELOR OF ENGINEERING**

**IN**

##### Computer Engineering and Application

**GLA University, Mathura**



##### 

##### November 2023

#### 

#### **BONAFIDE CERTIFICATE**

Certified that this project report **“V CHAT”** is the bonafide work of **“Abhishek Kushwaha”** who carried out the project work under my/our supervision.

|  |  |
| --- | --- |
|  |  |
| **SIGNATURE**    Rohit Agrawal    **HEAD OF THE DEPARTMENT**    Department of CEA | **SIGNATURE**    Ms. Robin Khurana  **SUPERVISOR**    Department of CEA |

Submitted for the project viva-voce examination held on 29th November 2023.

#### **INTERNAL EXAMINER EXTERNAL EXAMINER**

#### **BONAFIDE CERTIFICATE**

Certified that this project report **“V CHAT”** is the bonafide work of **“Aditya Vishwanathan”** who carried out the project work under my/our supervision.

|  |  |
| --- | --- |
|  |  |
| **SIGNATURE**    Rohit Agrawal    **HEAD OF THE DEPARTMENT**    Department of CEA | **SIGNATURE**    Ms. Robin Khurana  **SUPERVISOR**    Department of CEA |

Submitted for the project viva-voce examination held on 29th November 2023.

**INTERNAL EXAMINER EXTERNAL EXAMINER**

#### 

#### **TABLE OF CONTENTS**

List of Figures..................................................................................................... i

List of Tables...................................................................................................... i

**CHAPTER 1 Introduction**

1.1 Overview and Motivation

1.2 Objective .

1.3 Summary of Similar Application

1.4 Timeline

1.5 Organization of the Project

**CHAPTER 2 Software Requirement Analysis**

2.1 Requirement Analysis

2.2 Feasibility Analysis

2.3 Modules Description

2.4 Use-Case Scenarios

**CHAPTER 3 Software Design**

3.1 Data Flow Diagram

3.2 Use Case Diagram

3.3 Signaling, STUN and TURN Servers

**CHAPTER 4 Conclusion & Future Work**

##### 4.1 Conclusion

##### 4.2 Future Work

4.3 Important Links

**Bibliography**

**List of Figures**

**Chapter 3**:

3.1 Use Case

3.2 DFD

3.3 Network Concept

**List of Tables**

**Chapter 1:**

1.4 Gantt Chart

**Chapter 1: Introduction**

**1.1 Motivation and Overview**

In the dynamic landscape of communication technology, the WeChat project stands as a comprehensive communication infrastructure, transcending traditional boundaries. While the Android app serves as the mobile face of WeChat, a crucial component of our ecosystem is the web version built on the MERN stack. This web version, employing ReactJS instead of React Native, plays a pivotal role in facilitating cross-platform communication, providing structured control for developers, and hosting essential features such as developer panels and moderation controls.

The motivation behind WeChat extends beyond the realms of conventional messaging applications. It is rooted in the fusion of innovative technologies like React Native, WebSocket protocol, and WebRTC, not only for the mobile app but also for the web version. The goal is to provide users with a seamless, real-time communication experience that goes beyond text, incorporating multimedia elements and high-quality video calling.

**1.2. Objective**

The primary objective of the WeChat project is to establish a robust communication platform that encompasses both the Android app and the web version. The Android app focuses on real-time messaging and video calling, leveraging technologies like React Native, WebSocket, and WebRTC. Simultaneously, the web version, built on the MERN stack with ReactJS, extends the WeChat ecosystem by providing developer controls, moderation features, and a structured platform for enhanced user management.

1. Real-Time Communication Excellence: WeChat is committed to delivering excellence in real-time communication, ensuring users can exchange a variety of content seamlessly.
2. Seamless Video Calling Integration: WeChat strives to redefine video calling standards on Android devices, providing a high-quality, WebRTC-enabled experience.
3. Cross-Platform Compatibility with React Native and ReactJS: Our objective is to transcend platform limitations by using React Native for the Android app and ReactJS for the web version, offering a consistent user experience across devices.
4. User-Friendly Interface with Expo and MERN: WeChat is dedicated to providing a user-friendly interface, utilizing Expo for the Android app and the MERN stack for the web version.
5. Openness and Privacy: WeChat prioritizes an open yet secure environment, empowering users with control over their data and communication preferences on both mobile and web platforms.
6. Innovation and Adaptability: WeChat aims to stay at the forefront of technology, ensuring adaptability to emerging trends for both the mobile app and web version.

**1.3 Summary of Similar Applications**

In the vibrant landscape of mobile messaging applications, WeChat distinguishes itself by combining the strengths of existing platforms with a focus on real-time communication, seamless video calling, cross-platform compatibility, user-friendly interfaces, openness, privacy, and continuous innovation. As we survey the current market, several notable applications serve as benchmarks for comparison, each with its unique features, strengths, and limitations.

WhatsApp: Globally recognized, WhatsApp is a widely used messaging application celebrated for its simplicity and end-to-end encryption. While boasting a substantial user base, concerns arise regarding limited customization options and the necessity for phone number registration.

Telegram: Standing out with features such as secret chats, self-destructing messages, and a highly customizable interface, Telegram caters to a niche audience with a focus on security and privacy. However, its user base may be more limited compared to mainstream alternatives like WhatsApp.

**WeChat's Differentiators**: Building upon the strengths and learning from the limitations of existing applications, WeChat aims to differentiate itself through a combination of real-time communication excellence, seamless video calling integration, cross-platform compatibility, user-friendly interfaces, openness, privacy, and continuous innovation. By strategically addressing user needs and incorporating cutting-edge technologies, WeChat seeks to carve its niche in the competitive realm of mobile messaging applications, providing a distinctive and unparalleled experience on both Android and web platforms.

**1.4 Timeline**

The project will proceed through the following phases:

1. **Research and Learning (2 weeks):** Understanding React Native, ReactJS, WebSocket, and WebRTC.
2. **Android App Development (2 weeks):** Implementing real-time messaging and video calling features.
3. **Web Version Development (2 weeks):** Building the web version with MERN stack, emphasizing developer controls and moderation features.
4. **Integration and Testing (2 weeks):** Ensuring seamless cross-platform compatibility and rigorous testing.
5. **Documentation and Finalization (2 weeks):** Preparing the project report and finalizing the deliverables.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task / Period (in Weeks) | 1-2 | 3-4 | 5-6 | 7-8 | 9-10 |
| **Research and Learning** |  |  |  |  |  |
| **Android App Development** |  |  |  |  |  |
| **Web Development** |  |  |  |  |  |
| **Integration and Testing** |  |  |  |  |  |
| **Documentation** |  |  |  |  |  |

**Chapter 2: Software Requirement Analysis**

**2.1 Requirement Analysis**

The WeChat project's foundation lies in a thorough analysis of software requirements, encompassing both the Android app and the web version. The functional and non-functional requirements outlined below are essential for the successful development of this comprehensive communication infrastructure.

**Functional Requirements**

1. Real-Time Messaging:

- Users should send and receive text messages in real-time on both mobile and web platforms.

- Multimedia attachments, including images and videos, should be supported seamlessly.

2. Video Calling:

- WeChat must support high-quality video calling functionality using WebRTC on both mobile and web platforms.

- Users should initiate video calls seamlessly within the application on both platforms.

3. Cross-Platform Compatibility:

- The application should be compatible with various Android devices, ensuring a consistent user experience.

- Cross-platform compatibility must extend to the web version built on the MERN stack.

4. User Authentication:

- Secure user authentication mechanisms should be implemented to protect user data on both mobile and web platforms.

5. Contact Management:

- Users should manage contacts, create groups, and navigate through their contact list on both mobile and web platforms.

6. User Profile Management:

- WeChat must allow users to set and manage their profiles, including profile pictures and status messages on both mobile and web platforms.

**Non-Functional Requirements**

1. Performance:

- The application should offer smooth performance, even under high user load on both mobile and web platforms.

- Messages and calls should be delivered with minimal latency on both platforms.

2. Security:

- End-to-end encryption should be implemented for all messages and video calls on both mobile and web platforms.

- Secure protocols must protect user privacy on both platforms.

3. Scalability:

- The system should be scalable to accommodate an increasing number of users over time on both platforms.

4. Usability:

- The user interface should be intuitive and user-friendly, catering to users of varying technical proficiency on both platforms.

- Accessibility features should be considered to ensure inclusivity on both platforms.

**2.2 Feasibility Analysis**

**Technical Feasibility**

The technical feasibility of WeChat involves an assessment of whether the chosen technologies, including React Native, WebRTC, and Expo for the Android app, and the MERN stack with ReactJS for the web version, can effectively meet the project's objectives on both platforms.

**Operational Feasibility**

Operational feasibility focuses on evaluating whether the WeChat application, encompassing both mobile and web versions, can be seamlessly integrated into users' daily routines, considering potential disruptions and user training requirements.

**Economic Feasibility**

Economic feasibility assesses the financial viability of the WeChat project, encompassing both the Android app and the web version. This includes estimating development costs, potential revenue streams, and return on investment.

**2.3 Modules Description**

1. User Authentication Module: This module is responsible for authenticating users securely, ensuring the confidentiality of user credentials, and protecting against unauthorized access on both mobile and web platforms.
2. Messaging Module: The messaging module facilitates real-time text communication between users, supporting the exchange of multimedia content on both mobile and web platforms.
3. Video Calling Module: Utilizing WebRTC, this module enables users to initiate and participate in high-quality video calls within the application on both mobile and web platforms.
4. Contact Management Module: The contact management module allows users to organize and manage their contacts, create groups, and easily navigate through their contact list on both mobile and web platforms.
5. User Profile Management Module: This module empowers users to set and manage their profiles, including profile pictures and status messages, on both mobile and web platforms.

**2.4 Use Case Scenarios**

Scenario 1: Sending a Text Message

- Actor: User

- Preconditions: User is logged.

- Flow:

1. User selects a contact.

2. User types and sends a text message.

3. System delivers the message to the recipient on both platforms.

Scenario 2: Initiating a Video Call

- Actor: User

- Preconditions: User is logged in.

- Flow:

1. User selects a contact.

2. User initiates a video call.

3. System establishes a secure video connection on both platforms.

Scenario 3: Managing Contacts

- Actor: User

- Preconditions: User is logged in.

- Flow:

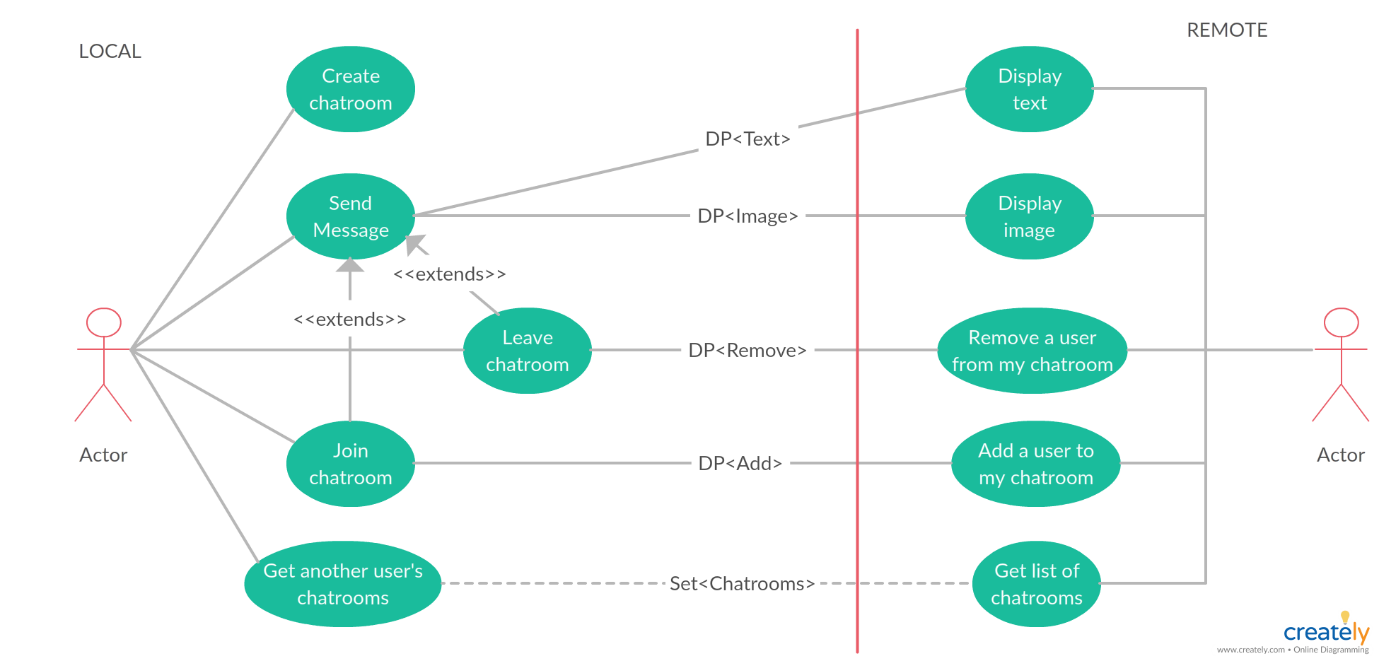
1. User navigates to the contact management section.

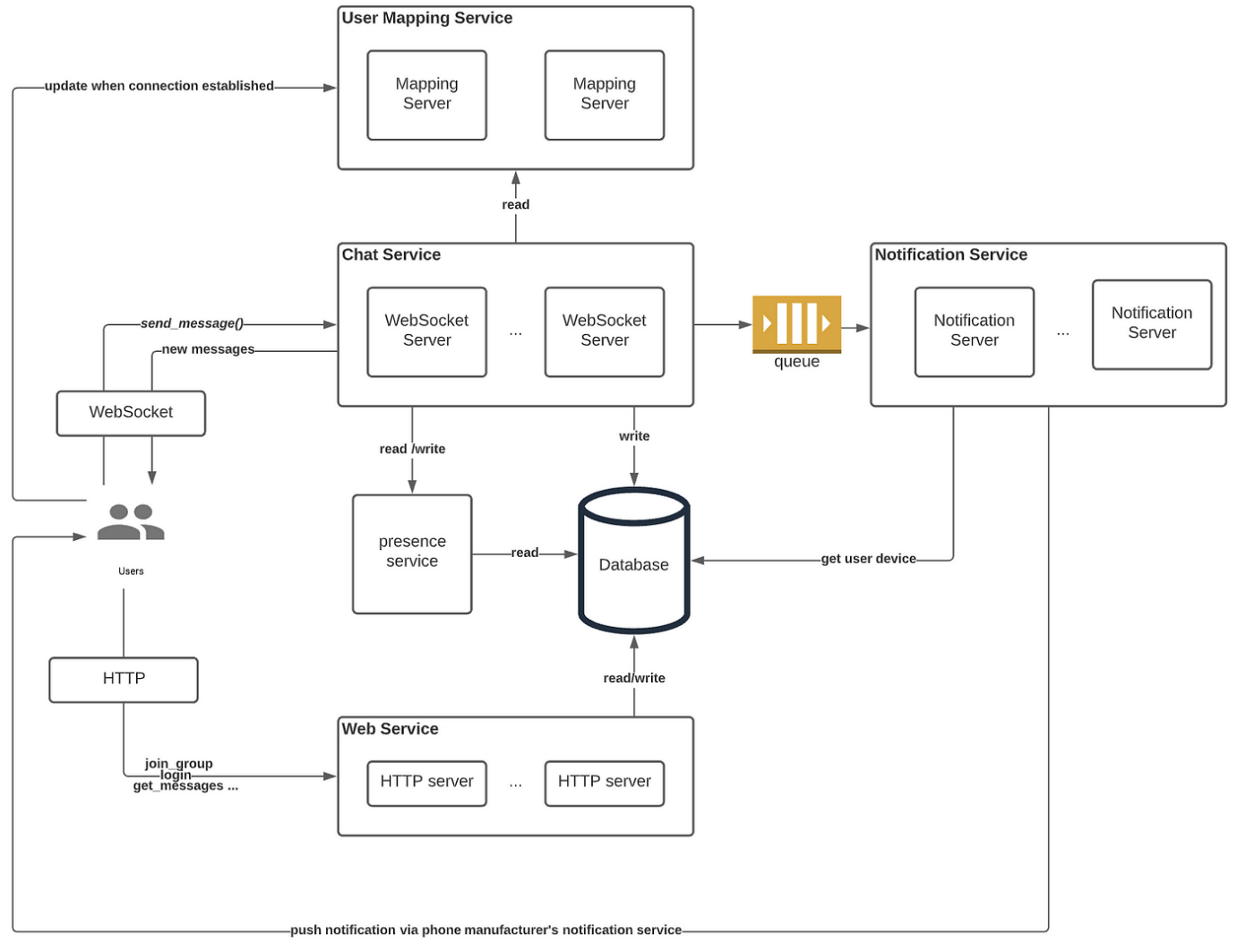
2. User creates a new contact group.

3. User adds and organizes contacts within the group on both platforms

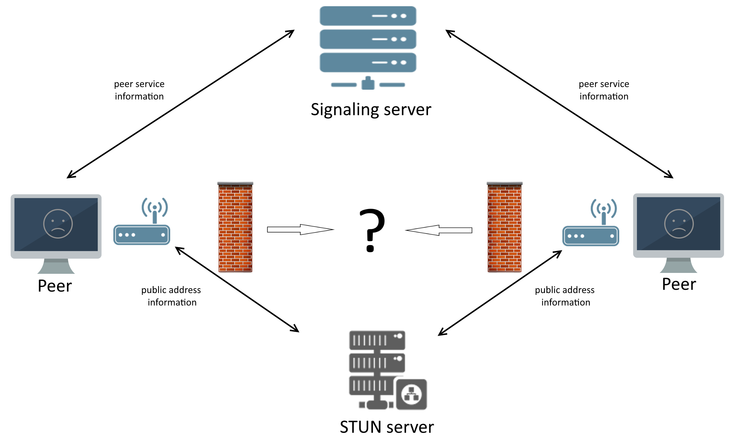
This chapter establishes the WeChat project as a comprehensive communication infrastructure, detailing the software requirements for both the Android app and the web version. The subsequent chapters will delve into the detailed implementation of these requirements, providing a comprehensive view of the development process on both mobile and web platforms.

**Chapter 3: Software Design and Flow**

**3.1 Use Case:****3.2 DFD:**



**3.3 Signaling, STUN and TURN servers**



#### 

**Chapter 4: Conclusion and Future Work**

* 1. **Conclusion**

The WeChat project has traversed a significant development journey, aiming to redefine the landscape of mobile messaging applications by providing a comprehensive communication infrastructure. Throughout the implementation process, we anticipate achieving the following outcomes:

**Expected Results/Outcome:**

1. Seamless Cross-Platform Communication: The integration of the Android app and web version is expected to provide users with a seamless cross-platform communication experience, allowing for real-time messaging and high-quality video calling.

2. Enhanced User Control: The web version's developer controls and moderation features are anticipated to empower users with more control over their communication environment, fostering a structured and secure platform.

**4.2 Future Work**

**Way Ahead**:

1. Optimization for Diverse Devices: Further optimization for a broader range of Android devices and web browsers will be undertaken to ensure a consistent experience for all users.

2. Enhanced Security Measures: Future iterations will focus on implementing additional security measures, ensuring end-to-end encryption and privacy features are robustly maintained.

**Required Modifications in the Solution**:

1. User Interface Refinement: Continuous efforts will be directed towards refining the user interface based on user feedback to enhance usability and aesthetics.

2. Integration with Additional Platforms: Exploring the possibility of extending the WeChat ecosystem to additional platforms, such as iOS, will be considered to broaden the user base.

**Change in Approach:**

1. Adoption of Emerging Technologies: As technology evolves, the project will stay adaptive by considering the integration of emerging technologies that can further enhance user experience and communication features.

**Suggestions for Extending the Solution**:

1. Collaborative Features: Future work may explore the inclusion of collaborative features, such as document sharing and collaborative editing, to transform WeChat into a versatile communication and productivity tool.

2. AI Integration: The integration of artificial intelligence for features like smart replies and content recommendations can add an innovative dimension to the user experience.

In conclusion, the WeChat project is poised to deliver a dynamic and user-centric communication infrastructure. While expecting positive outcomes, the project acknowledges potential deviations, and future work is outlined to address these, ensuring a continuous evolution towards excellence in mobile and web-based communication.

**Important Links:**

**GitHub Repositories:**

Web Version: <https://github.com/LogiqueClergyman/VChat>

Android App: <https://github.com/abhishek20111/A_chat>

**Video Intro:**

<https://clipchamp.com/watch/oB2QTPf9tA8>

Alternative Link: <https://drive.google.com/file/d/1AAtiLeMmW9TLLpubALJhX37V2mavotBW/view?usp=sharing>

**PowerPoint Presentation:**

<https://docs.google.com/presentation/d/1zVdXnsbEYTyZ881Z5FhxuJBKfSFGQTSk/edit?usp=sharing&ouid=118179297043032880851&rtpof=true&sd=true>

**Bibliography:**

NodeJS Docs: <https://nodejs.org/en/docs>

ReactJS Docs: <https://react.dev/learn>

MongoDB Docs: <https://www.mongodb.com/docs/>

React Native Docs: <https://reactnative.dev/docs/getting-started>

Expo Docs: <https://docs.expo.dev/>

Firebase Docs: <https://firebase.google.com/docs>

WebSocket MDN Docs: <https://developer.mozilla.org/en-S/docs/Web/API/WebSockets_API>

Socket.io Docs: <https://socket.io/docs/v4/>

WebRTC Docs: <https://developer.mozilla.org/en-US/docs/Web/API/WebRTC_API>

WebRTC Library Docs: <https://webrtc.org/getting-started/overview>