A Minor Project Report on

GuffGaff

Submitted in Partial Fulfillment of the Requirements for

the Degree of

Diploma in Information Technology Engineering

under

Council for Technical Education and Vocational Training

(CTEVT)

Submitted by

Krishna Bhandari (2222)

Paban Oli (2227)

Sangam Chaudhary (2238)

Tanka Budha Magar (2242)

Under the Supervision of

Er. Shishir Paudel

Date: 14th December, 2024

Department of Diploma in Information Technology Engineering

SECONDARY SCHOOL

TULSIPUR CENTER

Tulsipur Sub Metropolitan City-6, Dang, Nepal

CERTIFICATE

This is to certify that this report entitled "**GuffGaff**" has been submitted to the Department of Diploma in Information Technology Engineering for the fulfillment of the requirement for the award of the Diploma in Information Technology Engineering has been completed under your supervision. We recommend the report for acceptance and approval.

……

Er. Shishir Paudel

Department of Information Technology Engineering

Secondary School Tulsipur Center

Tulsipur-6, Dang

Date: 2081-09-17

APPROVAL

We recommended this report entitled "**GuffGaff**" submitted by "**Krishna Bhandari (2222), Paban Oli (2227), Sangam Chaudhary (2238), Tanka Magar (2242)"** in partial fulfillment of the requirements for the Diploma in Information Technology Engineering, which has been examined by us and accepted for the award of the Diploma in Information Technology Engineering under Council for Technical Education and Vocational Training (CTEVT).

Panels of Examiners

Name	Signature	Date
1. Er. Shishir Paudel Secondary School Tulsipur Center		
2.		
3.		
4.		

DECLARATION

We, hereby declare that the minor project report entitled "**GuffGaff**" being submitted by us towards the partial fulfillment of the degree of Diploma in Information Technology Engineering in the Department of Information Technology Engineering is a project work carried by us under the supervision of **Er. Shishir Paudel** and have not been submitted anywhere else. We will be solely responsible if any kind of plagiarism is found.

Date: … … … … … … …

Krishna Bhandari (2222)

Paban Oli (2227)

Sangam Chaudhary (2238)

Tanka Budha Magar (2242)

ACKNOWLEDGEMENT

A part from the efforts of all the team members, the section this project report topic depends largely on the encouragement and guidance of our teachers. We take this opportunity to express our gratitude to the teachers who have been instrumental in the approval of this project topic.

We would like to show our greatest appreciation to proof **Er.Shishir Paudel** and other staff members. We cannot think them enough for their tremendous support and help. They motivated and encouraged use very time while selecting the proper project topic. Withouttheir encouraged and guidance, we would not have been able to select the proper topic.

The contribution and support received from all the team members including **Krishna Bhandari (2222), Paban Oli (2227), Sangam Chaudhary (2238) & Tanka Budha Magar (2242)** is vital. The team spirit by all has made a project report work successful.

Thank you all

Krishna Bhandari (*kb5220568@gmail.com*)

Paban Oli *(pabanoli525@gmail.com)*

Sangam Chaudhary (*Sangamtharu603@gmail.com*)

Tanka Budha Magar (*tankastark@gmail.com*)

ABSTRACT

We have developed "GuffGaff", a real-time chat website tailored to provide users with an efficient, seamless, and engaging communication experience. The platform facilitates instant messaging, enabling individuals to connect and collaborate in real time within a virtual community. GuffGaff's design prioritizes responsiveness, user-friendliness, and security, creating an intuitive interface for dynamic interactions.

The platform's architecture is built on HTML, CSS, JavaScript, PHP, and MySQL, ensuring robust and secure data management. Real-time data transmission is achieved using php, allowing users to send and receive messages with minimal latency. The integration of modern frameworks and libraries enhances the responsiveness and performance of the platform, offering a fast and reliable user experience.

Key features of GuffGaff include user registration and login systems to ensure personalized and secure interactions. Each registered user is provided with a unique identifier for identity management. The platform supports real-time one-on-one chats, group conversations, and message synchronization across devices, ensuring that users remain connected without interruptions.

User privacy and data security are central to the platform. Secure username-password authentication and encrypted message storage safeguard personal information and communication. Messages and user details are stored in a MySQL database, ensuring data integrity and accessibility. Administrative tools empower system administrators to manage user activities, monitor content, and address technical issues to maintain system stability.

A feedback mechanism allows users to share their experiences, helping developers refine and improve the platform continually. GuffGaff serves as a versatile and modern communication tool, fostering meaningful connections and building vibrant virtual communities. Designed to meet the demands of real-time communication, GuffGaff provides a secure, efficient, and feature-rich platform for users worldwide.

Table of Contents

List of Figures

List of Table

CHAPTER 1

INTRODUCTION

1.1 Introduction of GuffGaff

In the fast-paced world of digital communication, GuffGaff emerges as a real-time chat website designed to bridge communication gaps and enable instant connections between users. With the growing need for immediate and efficient communication tools, real-time chat platforms have become essential in fostering seamless interactions, collaboration, and social connectivity.

Unlike traditional communication methods that can be slow or asynchronous, real-time chat platforms like GuffGaff offer instantaneous message delivery, allowing conversations to flow naturally and without delay. These platforms serve as virtual meeting points where individuals, groups, and communities can engage in meaningful discussions, share ideas, and collaborate effortlessly, regardless of geographical barriers.

The concept of real-time chat brings people together in a digital space, simulating the immediacy and spontaneity of face-to-face conversations. Whether it's connecting with friends, coordinating projects, or building professional networks, GuffGaff provides a dynamic environment where communication feels personal, direct, and engaging.

What sets real-time chat platforms apart is their versatility. GuffGaff caters to diverse user needs—enabling one-on-one messaging, group conversations, and collaborative discussions in real time. It offers a user-friendly and responsive interface, making it accessible to people from all walks of life. GuffGaff ensures that messages are delivered without delays, fostering smooth and uninterrupted communication.

In today's interconnected world, real-time communication has become an integral part of daily life. GuffGaff stands as a modern communication tool that empowers individuals to stay connected, exchange ideas, and build relationships in a fast, efficient, and secure environment.

1.2 Problem Statement

In the realm of real-time chat platforms, several challenges persist that hinder the smooth flow of communication and limit user satisfaction. One of the primary issues revolves around maintaining a safe and respectful environment where users can communicate freely without encountering inappropriate content, spam, or abusive behavior. As the user base grows, moderating real-time conversations and ensuring platform integrity becomes increasingly challenging.

Another major concern is cybersecurity and data privacy. With real-time exchanges involving sensitive information, there is an ever-growing need for robust security measures to protect user data from breaches, unauthorized access, and misuse. Users must feel confident that their conversations and personal information are safeguarded at all times.

Furthermore, ensuring high user engagement and retention poses a challenge, as chat platforms must continually evolve to meet user expectations. Features such as personalized user experiences, intuitive interfaces, and real-time message delivery need to be optimized to ensure seamless communication. A lack of responsiveness, delayed message synchronization, or complex navigation can lead to frustration and reduced participation.

Additional challenges include:

- Inappropriate content and spam prevention
- Data security and privacy concerns
- User retention and satisfaction
- Ensuring smooth real-time message delivery
- Absence of personalized user experiences
- Technical limitations impacting performance

1.3 Aim and Objective

The primary aim of a real-time chat website, such as "GuffGaff," is to offer a dynamic platform where users can engage in seamless and instant communication. By enabling real-time interactions, the platform fosters an environment for exchanging ideas, sharing information, and building relationships without geographical limitations. The objective extends beyond simple communication, striving to create a vibrant digital community where users feel connected, valued, and engaged.

GuffGaff emphasizes facilitating instant communication by offering tools that ensure quick and smooth message exchanges. It aims to promote collaboration and networking by connecting individuals with shared interests, whether for social or professional purposes. Additionally, the platform seeks to enhance user engagement through intuitive interfaces, interactive features, and personalized user experiences.

A key objective is to ensure accessibility and security, providing users with an intuitive platform that safeguards their data and fosters trust. The platform supports collaboration by incorporating features like group chats and file sharing, making it versatile for both personal and professional interactions. By prioritizing self-expression, privacy, and inclusivity, GuffGaff strives to create a space where users feel a sense of belonging and community.

In conclusion, the overarching aim of GuffGaff is to provide a real-time communication platform that not only facilitates instant conversations but also fosters meaningful connections and collaborations. By addressing user needs for engagement, usability, and security, GuffGaff aspires to enhance digital interactions and create a thriving virtual community.

Ultimately, GuffGaff aspires to provide an all-in-one solution for digital communication that is not only efficient but also enjoyable and meaningful. By addressing the fundamental needs of connectivity, engagement, security, and community-building, the platform aims to enhance the way people interact and collaborate in the digital age

Objective of this project are

Personal objective

- **Improving Technical Skills:** Enhancing proficiency in front-end technologies like HTML, CSS, JavaScript, and Bootstrap through official documentation, tutorial platforms, and textbooks.
- **Learning Backend Development**: Advancing knowledge in server-side technologies such as PHP and MySQL by utilizing resources like W3Schools, Geeks for Geeks, and video tutorials from YouTube channels like Code with Harry.
- **Familiarity with Agile Methodologies**: Gaining hands-on experience with agile frameworks such as Scrum and Kanban, including activities like sprint planning and daily stand-up meetings.
- **Team Collaboration**: Developing interpersonal communication and teamwork skills through active participation in team discussions, collaboration, and progress sharing during agile ceremonies.
- **Time Management**: Building effective time management skills by adhering to project timelines, setting priorities, and meeting deadlines during development phases.

1.4 Features

The features are as follows:

- **User Authentication**: Enables users to sign up and log in securely to access the chat platform.
- **Real-Time Messaging**: Facilitates instant text messaging with read receipts and typing indicators.
- **Group Chats**: Allows users to create and participate in group conversations with customizable settings.
- **Multimedia Sharing**: Supports sharing of images, videos, and files during conversations.

- **Searchable Message History**: Enables users to search for messages, keywords, or specific users within chat threads.
- **Personalized Profiles**: Users can set profile pictures and update personal information.
- **Feedback System**: Provides an option for users to share their experiences and report issues directly within the app.
- **Enhanced Privacy**: End-to-end encryption and secure data storage to protect user conversations.

1.5 Significance of Study

Real-time chat websites like GuffGaff hold immense significance in today's fast-paced digital world. Unlike forums, which primarily focus on asynchronous discussions, real-time chat platforms enable instantaneous communication, fostering more dynamic and engaging interactions. This immediacy is particularly beneficial in scenarios that demand quick responses, such as collaborative work, customer support, or personal conversations. By providing tools for real-time messaging, multimedia sharing, and group chats, platforms like GuffGaff enhance the quality and efficiency of communication.

Real-time chat systems contribute to building trust and engagement by creating an environment where users feel connected and heard. The ability to exchange messages, share experiences, and resolve issues instantly enhances user satisfaction and strengthens relationships. Advanced features like end-to-end encryption, secure logins, and privacy-focused designs ensure user data is protected, building confidence and trust in the platform.

The combination of connectivity, speed, and security makes real-time chat websites indispensable for both personal and professional communication. These platforms are especially valuable in fostering teamwork and collaboration, allowing users to brainstorm, coordinate tasks, and achieve shared goals seamlessly. Like forums, real-time chat systems contribute significantly to the broader digital ecosystem by empowering users, enriching their experiences, and fostering community engagement.

1.6 Project Justification

The GuffGaff forum is an innovative platform designed to address the growing need for an inclusive and user-friendly space where individuals and communities can engage in meaningful discussions, share ideas, and exchange knowledge. In an era of increasing digital communication, the absence of a dedicated, well-structured, and accessible forum has left a gap in collaborative interactions.

Benefits to the Community

The benefits of GuffGaff to the community are as follows:

- **Enhanced Communication:** GuffGaff is all about connecting people. Whether you're looking to share ideas, explore common interests, or just have meaningful conversations, it provides the perfect space to communicate effectively with like-minded individuals.
- **Knowledge Sharing:** Imagine a platform where everyone has something valuable to share. GuffGaff makes this a reality by letting users exchange ideas, insights, and expertise, creating a rich environment for learning and collaboration.
- **Community Building:** GuffGaff isn’t just a forum; it’s a place to belong. It’s designed to bring people together, helping build strong and vibrant communities where everyone feels welcome.
- **Problem Solving:** Got a question or a problem? The GuffGaff community is there to help. Members can find advice, solutions, and support for all sorts of challenges in a positive and collaborative atmosphere.

1.7 Scopes and Limitations

The Scopes GuffGaff are as follows:

- **Diverse Topics:** GuffGaff is designed to accommodate discussions across a wide range of topics, making it a versatile platform for all types of users.
- **Real-Time Interaction:** With features like live commenting and instant notifications, users can engage in seamless, real-time conversations.

- **Secure User Experience:** The platform incorporates robust security features such as encrypted login, data privacy measures, and moderated content to ensure a safe environment.
- **Customizable Profiles:** Users can personalize their profiles to express individuality and make meaningful connections.
- **Scalable Design:** GuffGaff is built with scalability in mind, ensuring it can handle growing user numbers and new feature additions.
- **Accessibility:** The website is designed to be mobile-friendly, ensuring users can participate in discussions from any device.

The Limitations GuffGaff are as follows:

- **Internet Dependency:** The platform requires a stable internet connection, limiting access for users in low-connectivity areas.
- **Moderation Challenges:** Managing inappropriate content or spam might become more challenging as the user base grows.
- **Feature Scope:** While the platform includes core functionalities, some advanced features, like AI-driven moderation or extensive analytics, might not be available initially.
- **User Engagement:** Success heavily relies on active user participation; limited engagement could hinder the platform's growth and vibrancy.

1.8 Advantages and Disadvantages

The advantages GuffGaff are as follows:

Easy to Use: GuffGaff is designed to be simple and user-friendly, so anyone can join and start participating without any hassle.

Instant Connections: With real-time messaging and updates, users can stay engaged and respond quickly during discussions.

Diverse and Inclusive: The platform welcomes people from all walks of life, creating a space where everyone can share their ideas and perspectives.

Personalized Experience: Users can customize their profiles, avatars, and even choose light or dark themes to match their preferences.

Safe and Secure: With strong security measures like encrypted logins and active moderation, GuffGaff ensures a safe environment for everyone.

Built to Grow: The platform can handle more users and features as the community grows, making it future-proof.

Access Anytime, Anywhere: Whether you're on your phone, tablet, or computer, GuffGaff works seamlessly across all devices.

Learn and Share: It's the perfect place to exchange ideas, solve problems, and learn from others in a collaborative environment.

Supportive Community: Users can seek advice, share experiences, or simply find encouragement from like-minded people.

The disadvantages GuffGaff are as follows:

Internet Dependency: Users need a stable internet connection to access the platform, which may limit usability in areas with poor connectivity.

Moderation Challenges: As the user base grows, maintaining effective moderation to control spam and inappropriate content can become increasingly difficult.

Privacy Concerns: Despite strong security measures, there is always a risk of data breaches or unauthorized access in any online platform.

User Engagement Reliance: The success of GuffGaff depends heavily on active participation; a lack of engagement could impact the platform's vibrancy.

Resource Demands: Expanding the platform to accommodate more users and features requires significant resources and ongoing investments.

Competition: GuffGaff faces competition from well-established forums and social media platforms, which might make attracting users more challenging.

Learning Curve: Some users, especially those unfamiliar with digital platforms, may face difficulty navigating or using advanced features

CHAPTER 2

LITERATURE STUDY/REVIEW

2.1 Project Elaboration

GuffGaff is the real-time chat feature of the ourDiscuss forum, designed to make communication faster, easier, and more engaging. It allows users to connect through one-on-one chats or group conversations, making it perfect for both personal and community interactions. With real-time messaging, users can instantly send and receive messages, while features like typing indicators and online status add a dynamic and interactive touch to conversations.

GuffGaff also supports sharing multimedia content like images, videos, and documents, making discussions richer and more expressive. Users can customize their experience with options like avatars, emojis, and GIFs, adding personality to their chats. Notifications ensure that no message or update is missed, keeping users connected at all times.

To maintain a safe and secure environment, GuffGaff uses end-to-end encryption for privacy and provides features like blocking and reporting. It's designed to bring people closer, making discussions more interactive and meaningful while complementing the overall forum experience. GuffGaff isn't just a chat feature.

2.2 GuffGaff Literature Study

The study of real-time chat systems like GuffGaff focuses on key aspects such as low latency, user-friendly interfaces, and robust security. Popular platforms like WhatsApp and Discord highlight the importance of features like typing indicators, read receipts, and multimedia sharing for enhancing user experience. Personalization options such as avatars, emojis, and themes improve engagement, while a well-designed notification system ensures users stay informed without feeling overwhelmed.

From a technical perspective, scalable backends, PHP technology for instant messaging, and end-to-end encryption are crucial for reliability and privacy. By

incorporating these insights, GuffGaff aims to deliver a modern, secure, and engaging real-time chat experience.

2.3 Similar Project

2.3.1 WhatsApp

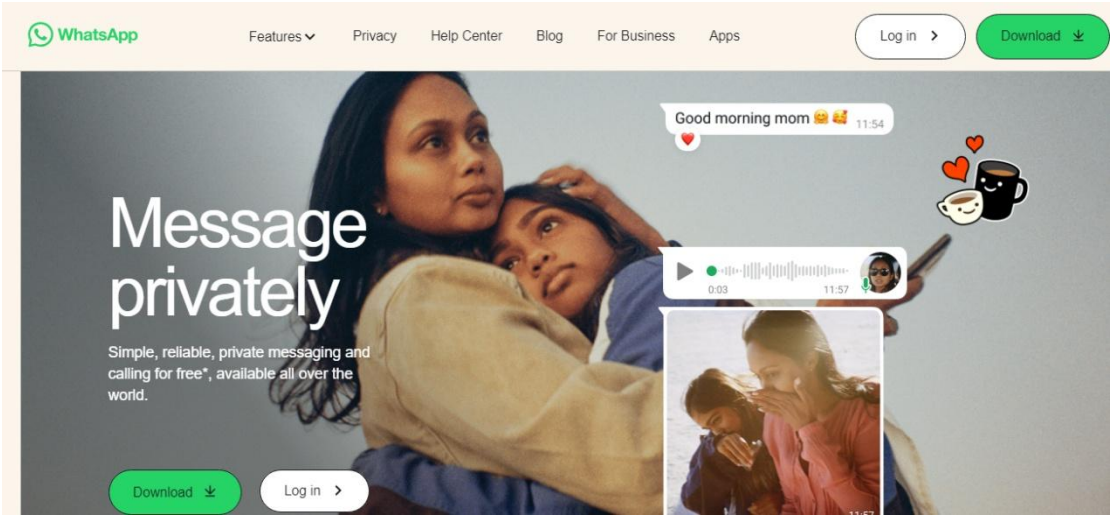

Figure 1: WhatsApp Home Page

WhatsApp was founded in February 2009 by Brian Acton and Jan Koum, former employees of Yahoo! A month earlier, after Koum purchased an iPhone, he and Acton decided to create an app for the App Store. Initially, the app was designed to display statuses in the phone's Contacts menu, indicating if someone was at work or on a call.Their discussions often took place at the home of Jan Koum's Russian friend, Alex Fishman. They realized they needed an iPhone developer to move the idea forward. Fishman found a Russian developer, Igor Solomennikov, on RentACoder.com and introduced him to Koum.

Koum named the app "WhatsApp" to sound like "what's up." On February 24, 2009, he incorporated WhatsApp Inc. in California. However, when the early versions of WhatsApp kept crashing, Koum considered abandoning the project. Acton encouraged him to wait a "few more months." In June 2009, the app had only been downloaded by a handful of Fishman's Russian-speaking friends. Around this time, Apple launched push notifications, which allowed users to be notified even when the app wasn't in use. Koum updated WhatsApp to notify users when someone in their

network changed their status. Surprisingly, users began using this feature to send quirky statuses like “I woke up late” or “I’m on my way.” Fishman noted, “At some point, it sort of became instant messaging.”

In August 2009, WhatsApp 2.0 was released for iPhone, introducing a dedicated messaging feature. This caused the number of active users to surge to 250,000. Although Acton was working on a different startup idea, he decided to join WhatsApp. In October 2009, Acton convinced five former Yahoo! colleagues to invest $250,000 in seed funding. Acton officially joined WhatsApp on November 1. By February 2013, WhatsApp had 200 million active users and 50 employees. Sequoia Capital invested an additional $50 million, valuing WhatsApp at $1.5 billion. Around the same time, WhatsApp acquired Santa Clara-based startup SkyMobius, the developers of Vtok, a video and voice calling app. In a December 2013 blog post, WhatsApp announced it had reached 400 million active monthly users. The year ended with $148 million in expenses, including $138 million in losses.

2.3.3 Instagram

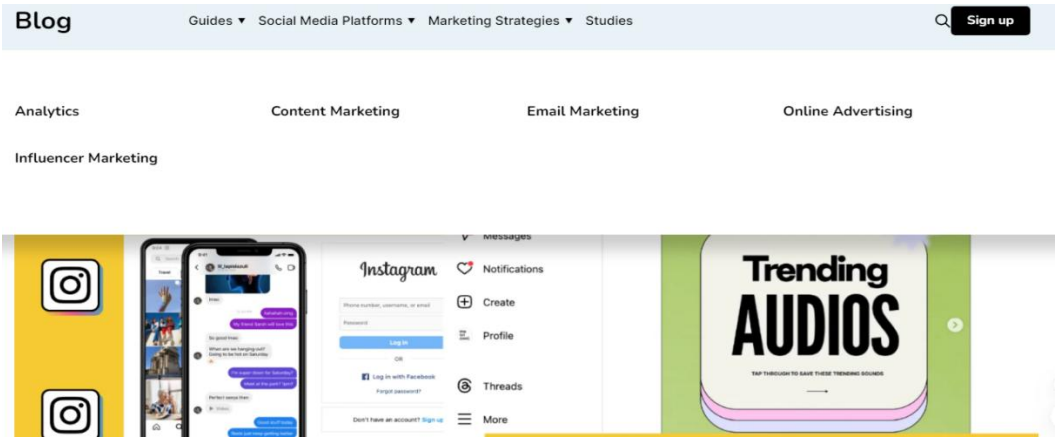

Figure 2: Instagram Home Page

Instagram originated with Stanford graduate Kevin Systrom, who had previously worked at Google. His initial creation was "Burbn," named after his interest in whiskey and bourbon. Inspired by the popularity of Foursquare and other location-based platforms, Burbn allowed users to post check-ins along with photos, though photos were not yet a common feature of social media posting.

Systrom secured venture funding for Burbn and recruited fellow Stanford graduate Mike Krieger, who had worked on the social media platform Meebo. Together, they reworked the concept to focus on photos taken on mobile devices and renamed it "Instagram." Systrom and Krieger emphasized minimalism in their prototype, focusing on images (with an option to add filters), comments, and “liking” features.

The duo decided to forgo a web version and instead focused on an iOS app to leverage the advanced photographic capabilities of the iPhone 4. Within a few months, they finalized the app and posted the platform's first photos in July 2010. Instagram was released to the public on Apple’s App Store on October 6, 2010, and it reached 25,000 users on its first day.

As Instagram’s popularity grew rapidly, the company attracted interest from various investors and potential buyers. In April 2012, 18 months after its launch, Instagram was acquired by Facebook for $1 billion in cash and stock. The acquisition occurred about a month before Facebook’s initial public offering (IP).

2.3.3 Facebook Messenger

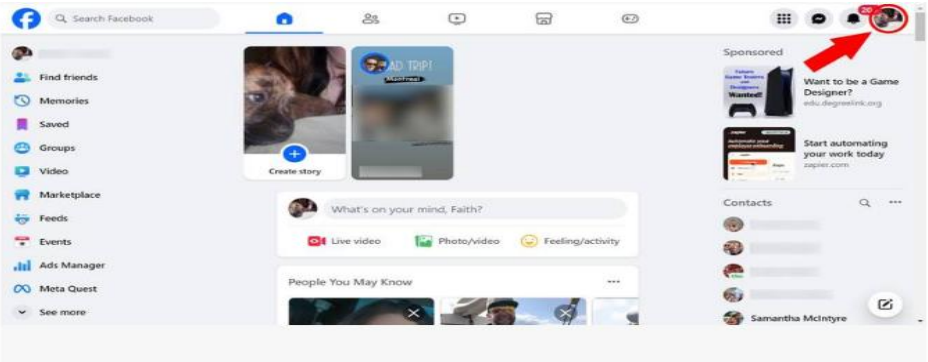

Figure 3: Facebook Messenger Home Page

Facebook Messenger is an American messaging app and platform developed by Facebook, Inc. Initially launched as "Facebook Chat" in 2008, the company revamped its messaging service in 2010. Later, standalone iOS and Android apps were released in August 2011, along with standalone Facebook Portal hardware for Messenger-based calling in Q4 2018.Subsequently, Facebook introduced a dedicated website interface, "Messenger.com," and separated the messaging functionality from the main

Facebook app. This allowed users to either use the web interface or download one of the standalone apps.

In April 2020, Facebook officially released the desktop version of Messenger, supported on Windows 10 and macOS. These are distributed via the Microsoft Store and the App Store, respectively.

Through Messenger, users can send messages, share photos, videos, stickers, audio, and files, react to messages, and interact with bots. The platform also supports voice and video calling.

CHAPTER 3

METHODOLOGY

3.1 Overview

A project management methodology is a set of tools, guidelines, and principles that help organize projects in a way that optimizes efficiency and performance. It provides a repeatable series of steps and principles to manage a project, ensuring that it is completed successfully and within the given constraints of time, budget, and scope.

In essence, a methodology is a collection of methods, practices, processes, techniques, procedures, and rules that guide project management. It outlines a specific approach to managing a project, including the activities, deliverables, and timelines. A methodology can be tailored to suit the specific needs of a project, industry, or organization.

3.2 Possible Methodologies for this Project

When planning and executing a project, choosing the right development methodology is crucial for success. Here are brief descriptions of four popular project methodologies: Waterfall, Prototype, Spiral, and Agile.

a. Waterfall Model

The Waterfall model is a linear and sequential approach to software development. It progresses through a series of distinct phases, each one starting only after the previous phase is complete.

Advantages

- Simple and easy to understand and manage.
- Well-suited for projects with well-defined requirements.

Disadvantages:

- Inflexible to changes once the project is underway. dissatisfaction.
- Often, clients only see the product at the end, leading to possible

dissatisfaction.

b. Prototype Model

The Prototype model involves building a working prototype of the software to understandthe requirements better and refine them through user feedback.

Advantages:

- Allows users to interact with a working version of the software early in the process.
- Helps identify and resolve issues early.

Disadvantages:

- Can lead to scope creep if the prototype evolves significantly from initialexpectations.
- Requires more user involvement, which can be resource-intensive.

c. Agile Model:

The Agile model is a flexible and iterative approach to software development and project management. It emphasizes collaboration, customer feedback, and incremental delivery.

Advantages:

- Easy to manage
- Little or no planning required
- Gives flexibility to developers

Disadvantages:

- Not suitable for handling complex dependencies
- More risk of sustainability, maintainability, and extensibility
- It is not useful for small development projects.

d. **Spiral Model:**

The Spiral Model is a Software Development Life Cycle (SDLC) model that provides a systematic and iterative approach to software development. In its diagrammatic representation, looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a phase of the software development process.

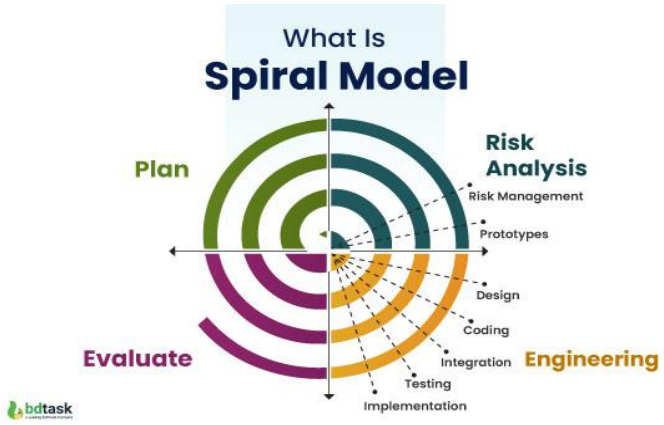

Figure 4: Spiral Model

The Spiral Model is a risk-driven model, meaning that the focus is on managing risk through multiple iterations of the software development process. It consists of the following phases:

a. **Objectives Defined:** In first phase of the spiral model, we clarify what the project aims to achieve, including functional and non-functional requirements.

b. **Risk Analysis:** In the risk analysis phase, the risks associated with the project are identified and evaluated.

c. **Engineering:** In the engineering phase, the software is developed based on the requirements gathered in the previous iteration.

d. **Evaluation:** In the evaluation phase, the software is evaluated to determine if it meets the customer's requirements and if it is of high quality.

e. **Planning:** The next iteration of the spiral begins with a new planning phase, based on the results of the evaluation.

The Spiral Model is often used for complex and large software development projects, as it allows for a more flexible and adaptable approach to software development.

3.2 Reason to select Spiral methodology

We chose the Spiral Model for building GuffGaff, our real-time chat website, because it's perfect for complex projects. It lets us develop step by step, tackling challenges like handling many users, ensuring fast performance, and keeping data secure.

Each cycle helps us spot and fix issues early, making sure everything runs smoothly. It's also great for adding new features based on user feedback, like improving messaging or adding cool extras like voice and video calling. The Spiral Model's flexibility means we can easily adapt to new ideas or changes, keeping GuffGaff modern and user-focused.

Characteristics of Spiral Methodology

- Built step by step through repeated cycles.
- Focuses on identifying and managing risks early.
- Flexible, allowing changes as the project evolves.
- Creates prototypes to gather feedback.
- Regular user feedback ensures the product meets needs.
- Ideal for large, complex projects.
- Clear documentation and planning in every phase.

Advantages of the Spiral Model

Below are some advantages of the Spiral Model:

a. **Risk Handling:** The projects with many unknown risks that occur as the development proceeds, in that case, Spiral Model is the best development model to follow due to the risk analysis and risk handling at every phase.

b. **Good for large projects:** It is recommended to use the Spiral Model in large and complex projects.

c. **Flexibility in Requirements:** Change requests in the Requirements at a later phase can be incorporated accurately by using this model.

d. **Customer Satisfaction:** Customers can see the development of the product at the early phase of the software development and thus, they habituated with the system by using it before completion of the total product.

Disadvantages of the Spiral Model

Below are some main disadvantages of the spiral model:

a. **Complex:** The Spiral Model is much more complex than other SDLC models.

b. **Expensive:** Spiral Model is not suitable for small projects as it is expensive.

c. **Too much dependability on Risk Analysis:** The successful completion of the project is very much dependent on Risk Analysis.

3.3 Reason not to Select Other Available Methodologies

a. Waterfall Methodology:

- **Lack of Flexibility**: The Waterfall model is linear and rigid, making it difficult to adapt to changing requirements. Once a phase is completed, reversing decisions or modifyingscope can be costly.
- **Late Feedback**: Since testing and validation occur at the end of the development process, issues may not be identified until it's too late, leading to costly rework.
- **Limited Collaboration**: The sequential nature of Waterfall often results in limited collaboration between teams. This can create communication barriers and misalignment.
- **Longer Time to Market**: With Waterfall, the entire project is completed before delivery, which can result in longer timeframes to release a product.

b. Agile Methodology:

- **Complexity:** Agile involves frequent iterations and adjustments, which can make coordination tricky. It requires good team communication and flexibility to handle changes.
- **Cost:** While Agile can save on costs with faster delivery, it may increase resource usage due to regular meetings, feedback sessions, and continuous involvement from stakeholders.
- **Overhead:** The need for constant reviews and updates can slow things down, especially if the team is not well-organized or if requirements keep changing.

c. Prototyping Methodology:

- **Time-Consuming:** Prototyping can be time-consuming, especially if the prototypesare elaborate or undergo multiple iterations.
- **Resource-Intensive:** Creating prototypes requires dedicated resources, includingdesigners, developers, and testers.
- **Scope Creep**: With frequent iterations and feedback, there's a risk of scope creep, where additional features or changes are continually introduced.

CHAPTER 4

TOOLS AND TECHNOLOGY

4.1 Tools and Technology Tools

Tools

Tools refers to a broad category of objects, devices, instruments, or software programs that are designed to assist in performing a particular task or achieving a specific objective. Tools can be physical objects that you hold and manipulate, such as hammers, screwdrivers, or wrenches, or they can be digital tools like software applications or computer programs.

Technology

Technology refers to the application of scientific knowledge, techniques, skills, and tools to create, modify, or improve processes, systems, products, or services in order to solve problems, achieve specific goals, or fulfill human needs.

4.1.1 Frontend technology

Front-end technology refers to the set of technologies and tools used to create and implement the user interface (UI) and user experience (UX) of a website or web application. It primarily focuses on the client-side development, where the user interacts with the application directly.

Frontend Technology are as follows:

HTML

Hyper Text Mark-up Language (HTML) is one of the most popular languages to design Web Pages and their content. HTML uses different tags, elements, images and some latest components to make Web Pages more attractive and user-friendly. With the help of some CSS, the look and feel of the web page can be easily changed. It should be considered an important source in the World Wide Web to structure the web page properly.

CSS

CSS is used with HTML to create and format content structure. It is responsible for colors, font properties, text alignments, background images, graphics, tables, etc. It provides the positioning of various elements with the values being fixed, absolute, and relative. CSS is designed to enable the separation of presentation and content, including layout, colors, &fonts. CSS operates on a "cascading" principle, meaning that styles can be applied to elements at different levels, and they can inherit properties from parent elements or override inherited styles.

JavaScript

JavaScript is a lightweight, cross-platform, single-threaded, and interpreted compiled programming language. It is also known as the scripting language for webpages. It is well-known for the development of web pages, and many non-browser environments also use it. JavaScript is a weakly typed language (dynamically typed). JavaScript can be used for Client-side developments as well as Server-side developments. JavaScript is both an imperative and declarative type of language. JavaScript contains a standard library of objects, like Array, Date, and Math, and a core set of language elements like operators, control structures, and statements.

4.1.2 Backend Technology

Backend technologies refer to the tools, frameworks and languages used to build and maintain the backend systems of applications. They are a set of server-side technologies that make up the architecture and functionality of the application. Backend technologies are responsible for handling tasks such as, data storage, server-side logic, application programming interface, authentication and security, robust, deployment, error handling.

Backend Technology are as follows:

PHP

PHP, an acronym for Hypertext Preprocessor, is a server-side scripting language and is one of the most widely used programming languages for backend web development. We can validate this statement by letting you know that platforms like Wikipedia, WordPress, Facebook, and many others are relying on PHP. This particular language is preferred for web development because of various prominent reasons such as cross- platform compatibility, OOPs features, easy integration with HTML, CSS, JavaScript, etc., huge community support, better flexibility & security, and many more. In addition, the language is quite easy to learn and use. Furthermore, there are various renowned PHP frameworks out there such as LA ravel, Symphony, Code Igniter, etc. that you can consider.

4.1.3 Designing Tools

MS Word

Microsoft is a graphical word processing program that users can type with. It is made by the computer company Microsoft. The purpose of the MS Word is to allow the users to type and save documents. Similar to other word processors, it has helpful tools to make documents. Microsoft word, often called as word, carries with it a lot of benefits. Let’s discuss it in a little bit more detail. It's the most popular word processing program in the world. One of the most obvious benefits is its availability. Almost all windows users installMicrosoft suit in their computer system.

4.1.4 Code Editor

Visual Studio Code: Visual Studio Code Though there are many text editors, visual studio text editor is used for development of the system. It is one of the most popular text editors for developers which are open-source software with various features. This text editor doesn’t crash while opening large files as it is stable. It is being continuously developed. With each release of new versions Visual studio text 1.89.1 there are new features.

Canva: Canva is a popular online graphic design platform that allows users to create stunning visuals with ease. It provides a user-friendly interface with drag-and-drop features, making it accessible to both beginners and professionals. Canva offers various templates for social media posts, presentations, posters, business cards, and more. With a vast library of fonts, images, and design elements, users can customize their designs effortlessly.

4.1.5 Server

Xampp: Xampp is an easy to install Apache distribution for Windows, Mac OS X, Linux and Solaris. The package includes the Apache web server, MySQL, PHP, Perl, a FTP server and PHP My Admin. It has been designed to be the easiest way to install and run a development server. XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself.

4.1.6 Databases

MySQL: MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Wideness's daughter My, and "SQL", the acronym for Structured Query Language. A relational database organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language that programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

4.1.7 Browser

Google Chrome: Google Chrome is a web browser developed by Google. It was first released in 2008 for Microsoft Windows, built with free software components from Apple WebKit and Mozilla Firefox. Versions were later released for Linux, macOS, iOS, and also for Android, where it is the default browser. The browser is also the main component of ChromeOS, where it serves as the platform for web applications.

Most of Chrome's source code comes from Google's free and open-source software project Chromium, but Chrome is licensed as proprietary freeware. WebKit was the original rendering engine, but Google eventually forked it to create the Blink engine; all Chrome variants except iOS used Blink as of 2017.

As of April 2024, Stat Counter estimates that Chrome has a 65% worldwide browser marketshare (after peaking at 72.38% in November 2018) on personal computers (PC), is most used on tablets and is also dominant on smartphones. With a market share of 65% across all platforms combined. Chrome offers a clean and intuitive user interface, making it easy to navigate and use. It features a minimalistic design with a combined search and address bar, known as the Omni box, where users can enter URLs or search queries.

4.2 Time Schedule

A time schedule is a document that outlines a plan for how time will be allocated and managed during a specific period. It includes a list of tasks, deadlines, and the estimated time required for each task. A time schedule helps individuals and teams to prioritize tasks, coordinate their efforts, and manage their time effectively.

10 November 2024 – 31 March 2025	Weeks	Task						
		Proposal Writing	Requirement Analysis	Design	Coding	Testing	Implementation	Evaluate
	W 01							
	W 02							
	W 03							
	W 04							
	W 05							
	W 06							
	W 07							
	W 08							
	W 09							
	W 10							
	W 11							
	W 12							
	W 13							
	W 14							
	W 15							
	W 16							
	W 17							
	W 18							

Table 1: Time Schedule

4.3 ER Diagram

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

Used different symbol of ER Diagram:

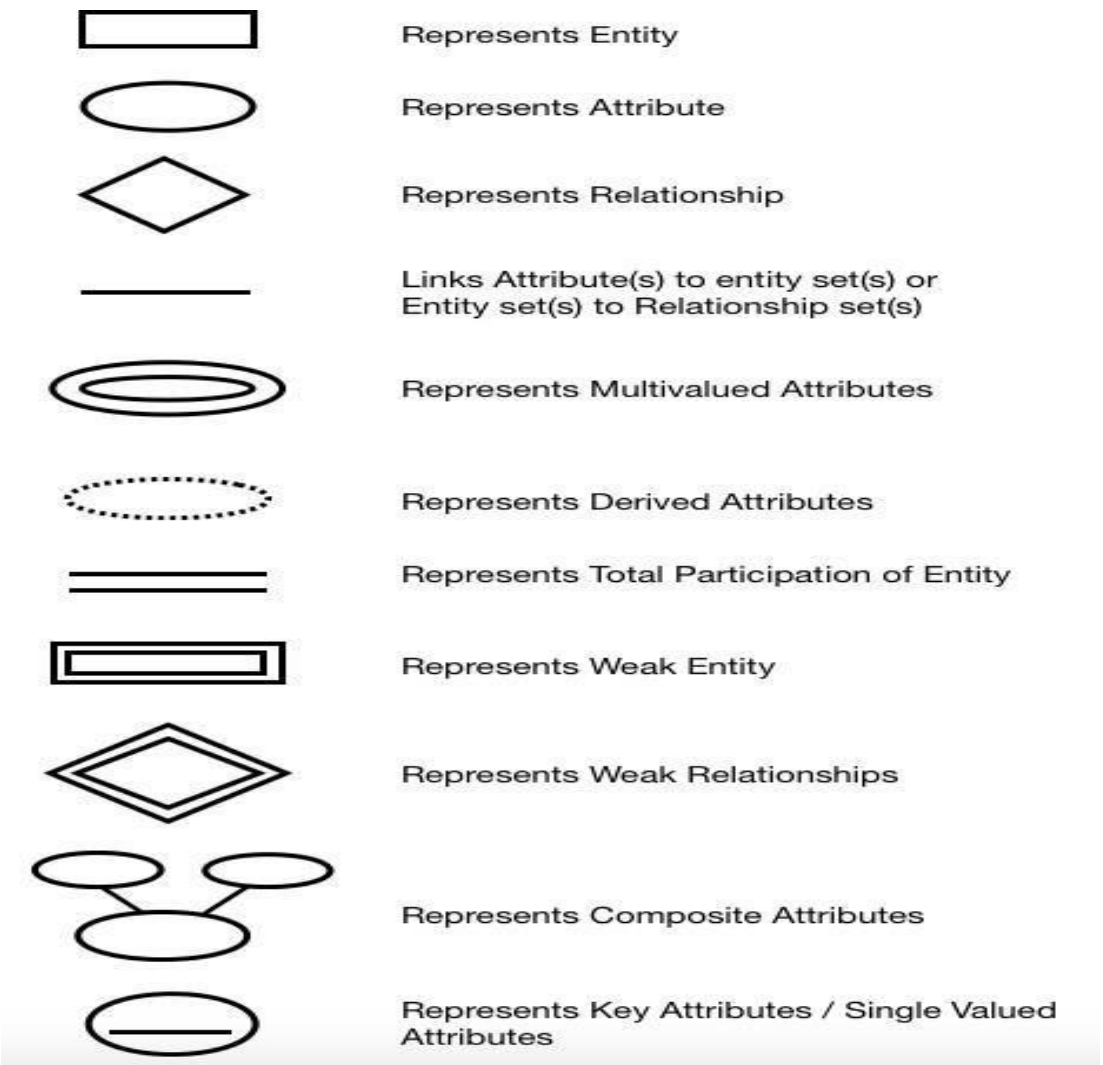

Figure 5: Symbol of ER Diagram

ER Diagram of GuffGaff:

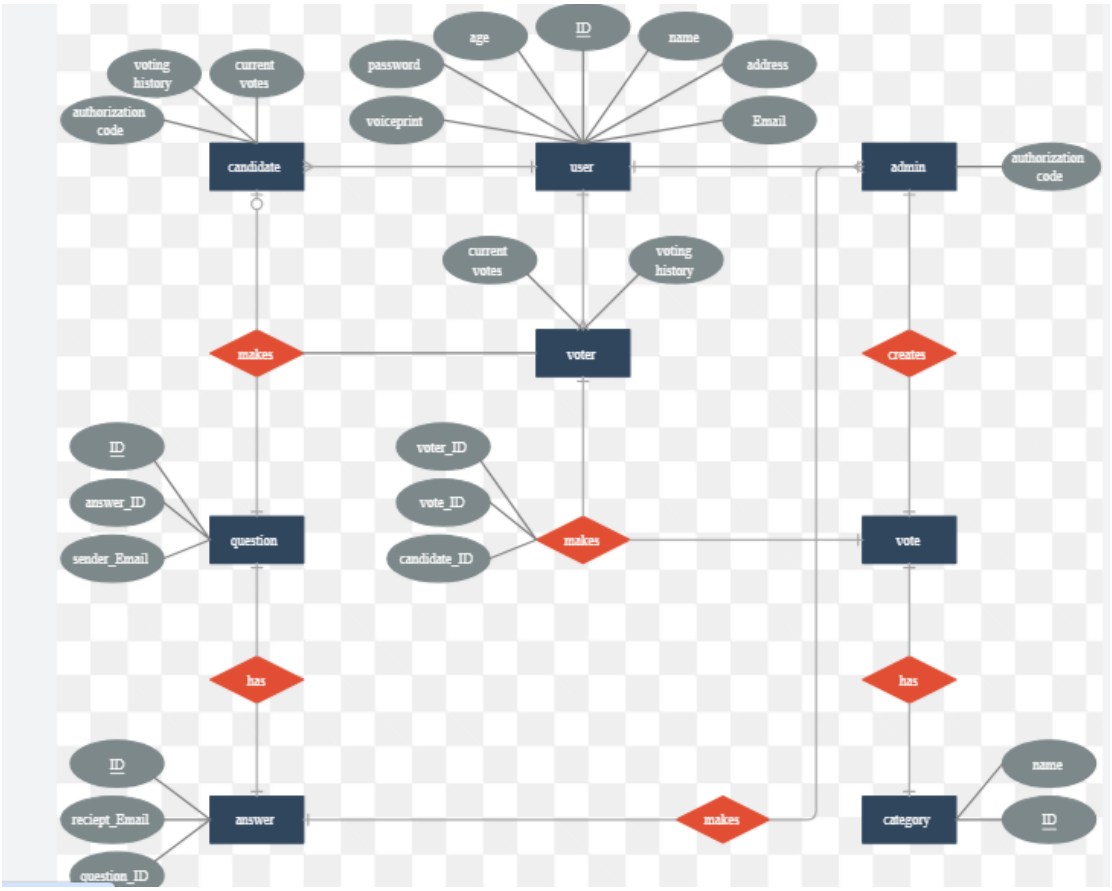

Figure 6: ER Diagram of GuffGaff

CHAPTER 5

PROJECT BOUNDARIES, RESOURCE, RISK

5.1 Boundaries for the project

Project boundaries define the scope and limits of a project, clearly outlining what is included and excluded. This helps set clear expectations, ensuring everyone understands the project's deliverables and objectives. By specifying inclusions, such as specific tasks and goals, and exclusions, such as activities or features that are not part of the project, it helps prevent scope creep and keeps the project focused. Additionally, project boundaries consider constraints like budget, time, and resources, and outline assumptions made during planning, ensuring the project stays on track and within its defined parameters.

5.2 Resource

Human Resources:

- Project Manager
- Developers
- Designers
- Testers/QA Analysts
- Content Creators/Moderators

Financial Resources:

- Budget
- Funding
- Cost Management

Physical Resources:

- Hardware (Computers, servers, and other physical technology)
- Office Space
- Supplies

Information Resources:

- Project Documentation
- Knowledge Base
- Data

Technological Resources:

- Software Tools (IDEs, version control systems, and other development tools)
- Frameworks and Libraries (Technologies and code libraries)
- Testing Tools

5.3 Risk

When developing a real-time chat application like GuffGaff, several risks may arise that can impact its success. Below are key risks and mitigation strategies:

Technical Risks

1. **Integration Issues:**
   - GuffGaff might not integrate properly with user authentication, group chat, or media-sharing features.
   - Solution: Conduct thorough testing and use standardized APIs for seamless integration.
2. **Performance Bottlenecks:**
   - As user activity increases, GuffGaff might experience slow message delivery, lag in real-time updates, or server crashes.
   - Solution: Implement load testing, code optimization, caching mechanisms, and scalable cloud-based infrastructure.
3. **Security Vulnerabilities:**
   - Risks like SQL injection, XSS attacks, unauthorized access, and data breaches can affect user data privacy.

- Solution: Regular security audits, end-to-end encryption, and implementing secure authentication methods (OAuth, JWT, 2FA, OTP verification).

Resource Risks

4. **Human Resource Availability:**
   - Key developers or designers may become unavailable, delaying progress.
   - Solution: Cross-train team members and maintain a backup plan for critical roles.
5. **Budget Overruns:**
   - Unexpected expenses for server costs, third-party integrations (Agora for video/audio calls), or storage requirements may arise.
   - Solution: Monitor expenses closely and set aside a contingency budget.

Scope and Planning Risks

6. **Scope Creep:**
   - New feature requests such as custom emoji packs, message reactions, or additional admin controls may extend timelines.
   - Solution: Define a clear project roadmap and manage feature requests through a structured change process.
7. **Unclear Requirements:**
   - Miscommunication about real-time features, offline message handling, or notification systems could lead to unexpected rework.
   - Solution: Ensure early and frequent engagement with stakeholders to refine requirements.

User Adoption Risks

8. **Low User Engagement:**
   - If GuffGaff fails to attract users, it could lead to poor adoption.
   - Solution: Improve UI/UX, ensure fast performance, promote user-friendly onboarding, and implement push notifications to increase engagement.

9. **Negative User Behavior:**
    - Spam messages, inappropriate content, and harassment can degrade the user experience.
    - Solution: Implement AI-powered moderation, keyword filtering, user reporting, and blocking features.

Project Management Risks

10. **Timeline Delays:**
    - Development, testing, or unexpected technical issues may delay the project.
    - Solution: Use Agile development methodologies with iterative releases to manage progress effectively.
11. **Dependency Risks:**
    - GuffGaff relies on third-party services like Agora (voice/video calls) and Supabase (user authentication/storage), which might change or become unavailable.
    - Solution: Prepare backup solutions and contingency plans for critical dependencies.

External Risks

12. **Regulatory Changes:**
    - New data privacy laws (e.g., GDPR, CCPA) may impact how user messages and personal data are stored and processed.
    - Solution: Stay updated on regulations and ensure compliance with data protection laws.
13. **Market Changes:**
    - If competitors release similar features or shift user expectations, GuffGaff might lose relevance.
    - Solution: Continuously monitor market trends and adapt the platform accordingly.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 Summary

GuffGaff is a real-time chat application that enables users to exchange messages instantly, conduct group chats, and share multimedia content. This application offers various features such as one-on-one and group chat, emoji and GIF support, image/video sharing, file attachments, audio messages, typing indicators, online/offline status, profile customization, light/dark mode, and search functionality**.**

One of the primary features of this app is real-time messaging, which allows users to communicate instantly and engage in live conversations. It includes **s**ecure JWT login, password encryption, OTP email verification, and the ability to block/unblock users. Additionally, the app features an admin management system, which enables administrators to monitor user activity, manage abuse reports, and efficiently control the platform.

The development of this application required essential resources, including human resources (developers, designers, testers, and administrators), financial resources (server hosting, development costs, maintenance), physical resources (hardware and network infrastructure), and technological resources (React.js, Node.js, MongoDB, Supabase, Redux, Agora API, etc.).

6.2 Conclusion

The GuffGaff chat application project has been successfully completed. It is a multi-functional and user-friendly chat system designed to provide a seamless real-time messaging experience, voice and video calling, and secure communication. With its simple interface, easy usability, and advanced security features, the platform ensures an excellent user experience.

To achieve successful implementation, effective management of human, financial, physical, and technological resources was necessary. Potential risks such as technical issues, data security challenges, and user engagement were

addressed with a proper risk management plan, ensuring GuffGaff remains stable and reliable.

In the future, plans include integrating an AI-based recommendation system**,** automated message response system, and enhanced encryption technologies to improve security and user experience. While some features were not implemented due to time constraints, upcoming updates aim to refine and enhance GuffGaff into an even more user-friendly and advanced chat platform.

Appendix A: References

- Wikipedia [Online] Available at:

  https://en.wikipedia.org/wiki/WhatsApp

  [Accessed at 26 January, 2025]

- Wikipedia [Online] Available at:

  https://en.wikipedia.org/wiki/Telegram(software)

  [Accessed at 26 January, 2025]

- Wikipedia [Online] Available at:

  https://en.wikipedia.org/wiki/FacebookMessenger

  [Accessed at 26 January, 2025]

- JavaPoint [Online] Available at:

  https://www.javatpoint.com/chat-application-in-node-js

  [Accessed at 8 February, 2025]

- Geeks for Geeks [Online] Available at:

  https://www.geeksforgeeks.org/websockets-in-node-js/

  [Accessed at 26 February, 2025]

- Wikipedia [Online] Available at:

  https://en.wikipedia.org/wiki/Google_Chrome

  [Accessed at 2 March, 2025]

- Lucid chart [Online] Available at:

  https://www.lucidchart.com/pages/er-diagrams

  [Accessed at 5 March, 2025]

Appendix B: Project Screenshots

a. Home Page

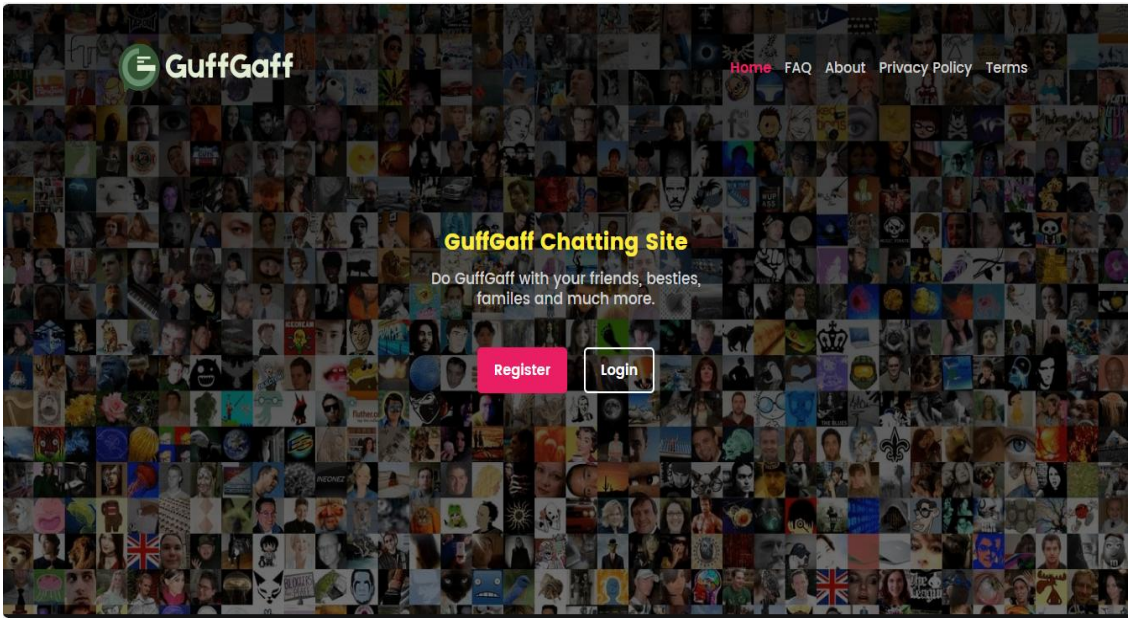

Figure 7: Home Page

b. About Page

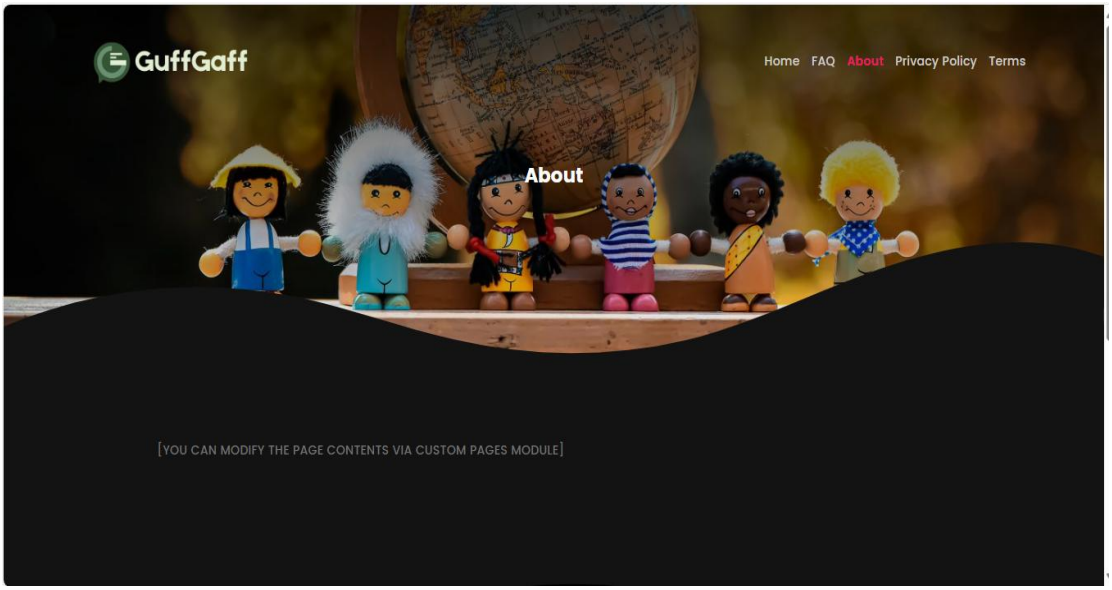

Figure 8: About Page

c. Signup Modal

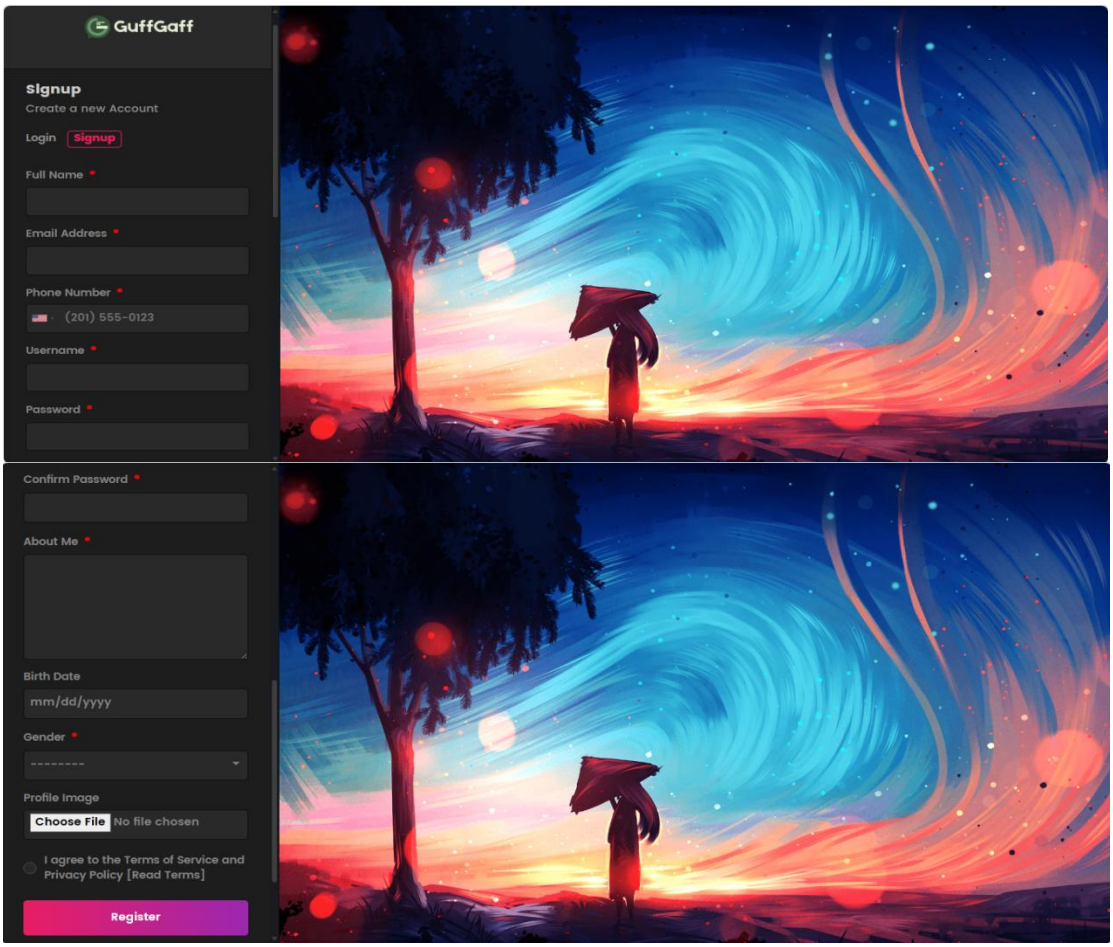

Figure 9: Signup Model

d. Login Modal

Figure 10: Login Modal

e. Help page

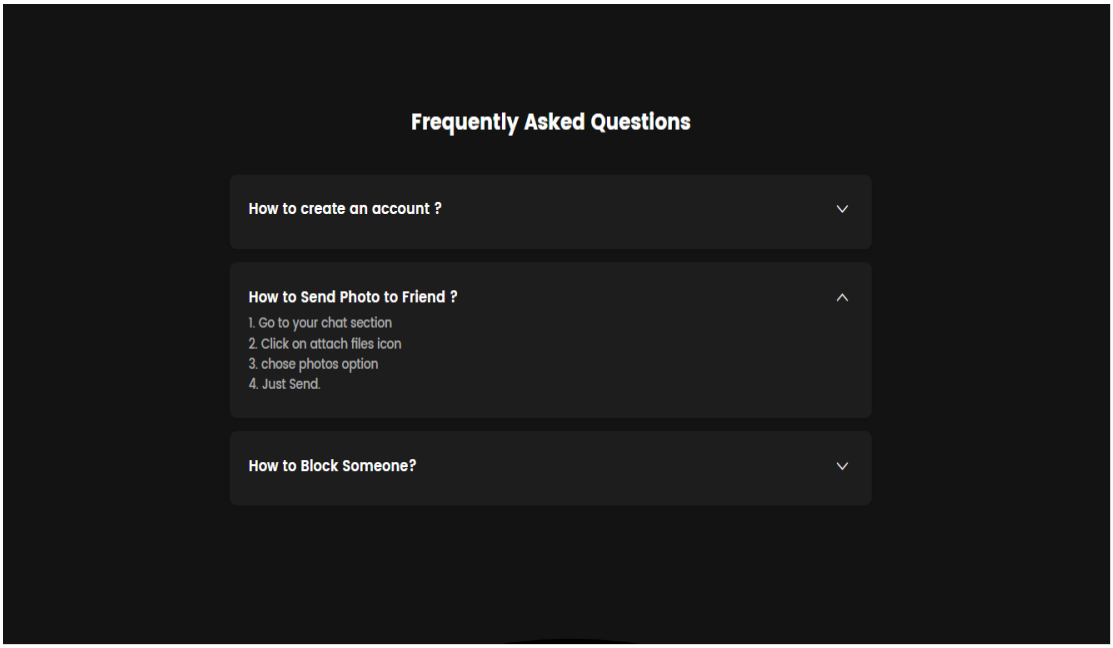

Figure 11: Help Page

f. User profile page

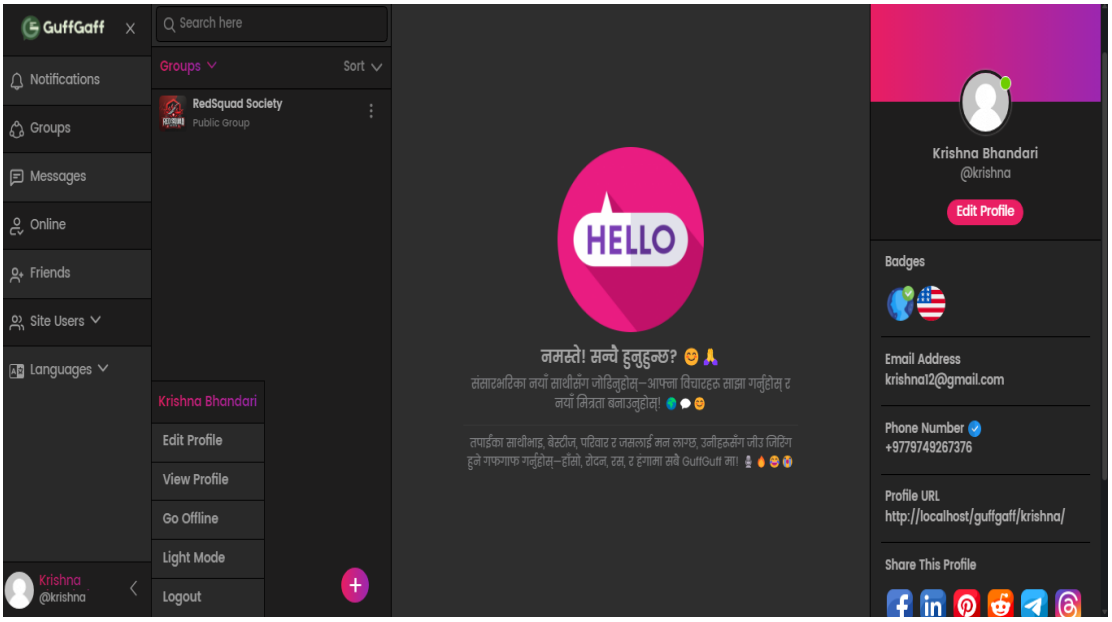

Figure 12: User profile page

g. Messaging (Messaging both friends)

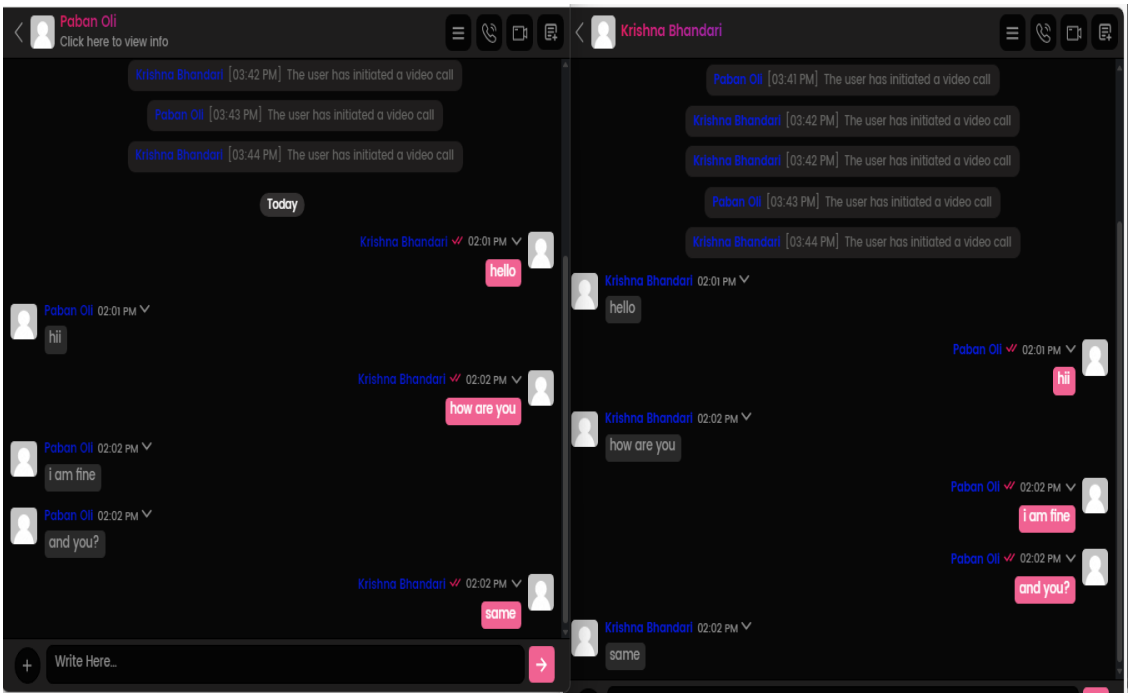

Figure 13: Messaging both friends

h. Calling page

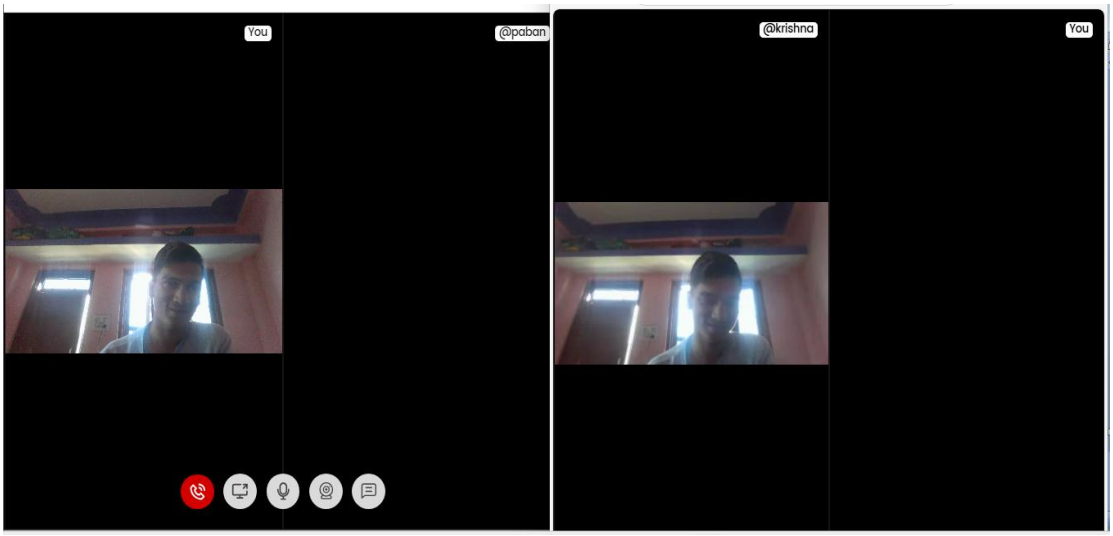

Figure 14: Calling page

Appendix C: Team members and its roles

S.N.	Name of Team Members	Roles
1	Krishna Bhandari	Research, Proposal, Coding, Documentation, Testing, Designing Gather Requirement.
2	Paban Oli	Gather Requirements, Documentation, Testing,Coding.
3	Sangam Chaudhary	Information gathering, Proposal, Documentation,Testing.
4	Tanka Budha Magar	Research, Gather Requirements, Coding, Designing, Documentation, Testing.

Table 2: Team Members