Software Engineering Principles

SYST45713

Phase 4

Discord Application

Submitted by:	
Student Name	**Student ID**
SM	
Chris Miele	
MA	

Submitted To: Professor SRB

Table of Contents

Introduction

Discord is a service that provides functions such as VOIP and instant messaging. End users can join public or private servers. Discord connects End-users globally through VoIP and facilitates messaging in a User-Friendly Environment. The application is accessed from their web application or by downloading the application to your smartphone or desktop. End-Users can create an online profile and interact with other users and allow the ability to create a server to host multiple people for voice and text-based communication. Users can also host and watch other community

members' streams.[1]

Business Summary

Current System

The current system that discord employs allows end-users to register an account through their website or application. The credentials are stored in a database, and end-users have the option to add 2FA.[2] The website uses HTTPS, and you can use VOIP and Messaging to communicate with one another. HTTPs encryption is used to secure data coming from the user to the server but is

[1] About Discord | Our Mission and Values
[2] Security, Discord, and You!

unencrypted upon reaching the server. Discord doesn't have a robust spam filter. End-users can create public and private servers. The Public Servers allow anyone to join, while the Private servers require an invite link to connect. Discord allows you to screen share and provide server owners with basic permission controls. Permissions are often bundled with multiple capabilities that administrators would want to be separated.

Proposed System

Our proposed system will implement improvements to the Discord platform focusing on the areas of control and security. The new system will allow for better defined permissions that server administrators can use to customise the configuration for their servers. Implementing a more robust spam filter into the discord environment is imperative to stop user-created bots from infecting users with malware. Spam filtering also stops other users from impersonating staff. The new encryption model will allow secure messaging in transit and disallow third parties from viewing your data. Stricter security policies will strengthen user privacy and protections on the platform. [3]

Control

- Permissions will be updated to allow greater control over user roles
- Separating the abilities of Manage Channels and Manage Roles

Security

- Spam filtering is important to implement into the discord environment as bots try to infect users with malware
- Security is quickly becoming a major deciding factor for end-users, end-to-end encryption will help keep users information secure
- Mandatory two-factor authentication

[3] Understanding Discord — Roles and Permissions | by Lela Benet | Community Builders Blog | Medium

Feasibility Study

Definition

A Feasibility study is held during the initiation phase of the project and is compiled before major expenses and contracts are agreed upon. This study provides a thorough analysis of an idea and is used to determine potential issues. [4]

Upon completion of the study, the results will determine whether the content of the report should be implemented and it will showcase possible alternatives. Management will examine the case through a business strategy lens and will make a "go/no-go" decision.

Purpose

This study serves the purpose of providing upper-management with enough context to determine if the project can be complete while investigating possible alternatives.[4]

Types of Feasibilities

1. **Economic Feasibility**

   ### Definition

   Economic Feasibility is defined as being perturbed about proceeding with the Software Project. The Factors are Value of the Project, Profitability. You can dictate this by performing Cost-Benefit Analysis. [5]

   Cost: $ 61,000

---

[4] B, S R. "Software Engineering Week 2-2." 22 Sept. 2022.

[5] Software Engineering and Testing - B. B. Agarwal, S. P. Tayal, Mahesh Gupta - Google Books

2. Technical Feasibility

Definition

Technical Feasibility is implying the Technologies(Hardware/Software) is sufficient enough to solve the clients problem. The Aspects that define this are Type of Technology, Is the Technology New, Can Said Technology be Acquired.[6]

Tools used: Javascript, Python, React Framework, Elixir, Rust, and C++, VSCode, Firebase, APIS

Hardware used: Employee issued Macbook Pro(M2), Nuc(Mini PC), Cloud Servers(AWS, Azure, Digital Ocean)

Operating systems used: Windows, and Linux, Mac

3. Schedule Feasibility

Definition

The schedule feasibility tries to ascertain the timeframe of a project by looking at a variety of variables that need to be overcome. Some of the most important variables deal with expertise or lack thereof. If tools are available there still must be employees trained or hired to be effective in their use of these tools. These factors must be taken into consideration because they have a direct impact on the projected deadline. The

---

[6] Software Engineering and Testing - B. B. Agarwal, S. P. Tayal, Mahesh Gupta - Google Books

[7] B, S R. “Software Engineering Week 2-2.” 22 Sept. 2022.

repercussions of missing deadlines must also be discussed in this feasibility. [7] **Time**:

The timeline for project delivery is 13 weeks.

4. Operational Feasibility

Definition

Operational feasibility is the measure of a proposed system's ability to address the problems of the current system and all the requirements that have been outlined. Managers and endusers give their thoughts on the presented problems and all the possible alternatives to the proposed system. An evaluation occurs on the viability of the new system and gauges the resistances that may crop up from relevant parties. Another key issue is how changes to the environment of the software also influence the
interactions with the software.[7]

Operational Feasibility Report: The operational feasibility report will be conducted after the completion of this project.

[7] B, S R. “Software Engineering Week 2-2.” 22 Sept. 2022.

Feasibility Report

Management Summary

Discord is a company that focuses on VoIP communications, including text messaging and video calls. The project entails the improvement of key security features that are inadequate in the Discord application. These features include end to end encryption, mandatory two factor authentication, spam filtering and the enhancement of available permissions.

Project Details	
Cost	~~61k~~ Price has been adjusted to 170k
Time	13 weeks
Tools used	☑ Javascript ☐ ~~Python~~ ☑ React Framework ☐ ~~Elixir~~ ☐ ~~Rust~~ ☐ ~~C++~~ ☑ VSCode ☑ Firebase ☑ Signal
Hardware used	Macbook Pro(M2), Nuc(Mini PC), Cloud Servers(AWS, Azure, Digital Ocean), Database
Operating system used	Windows, Linux

Functionality

Control

- Permissions will be updated to allow greater control over user roles
- Separating the abilities of Manage Channels and Manage Roles

Security

- Spam filtering is important to implement into the discord environment as bots try to infect users with malware
- Security is quickly becoming a major deciding factor for end-users, end-to-end encryption will help keep users information secure
- Mandatory two-factor authentication

Scope of the project

Price	170k
Time	13 weeks
Tools used	JavaScript, React Framework, VSCode, Firebase, Signal
Hardware used	• Server • Database
Operating system used	• Windows • Linux
Functionality	• Permissions will be updatedto allow greater control over user roles • Separating the abilities of Manage Channels and Manage Roles • Spam filtering is important to implement into the discord environment as bots try to infect users with malware • Security is quickly becoming a major deciding factor for end-users, end-to-end en cryption will help keep users information secure • Mandatory two-factor authentication
Performance	• Addition of end-to end encryption retains previous functionality and performance • Spam filtering accurately targets spammers with little to no false positives more security with no • Permission controls provide additional performance cost to the
Payment Details	Advance payment of 40%, and after every completed phase another 15% will be paid.
Maintenance	Two years of maintenance coverage

John Smith

Customer Signature

System Analyst Signature

Requirement Elicitation Techniques

Interview

Interviewing is used when requirements for the project are known. For this technique, the interviewer forms questions to ask stakeholders to obtain information. The types of questions to ask play a pivotal role in conducting an interview. It is good practice to always start from simple questions and build up to more advanced questions. Open-ended questions are useful for receiving detailed information whereas close ended allows the ability to obtain clarification and confirmation. The interviewer should avoid ambiguous questions to create a cohesive process for information collection. The basis for conducting these interviews is fairly standard. There should be a setting for the interview to take place, a notable schedule for the duration of the interview. There are various ways of conducting interviews, the interviewer can use point form to jot down quick notes while maintaining eye-contact with the interviewee and still remaining engaged. Once the interview has concluded. The point form notes must be transcribed with further context. The interviewer should also schedule out time towards the end of the interview for confirming their collection of notes. [9]

Questionnaire

There are 3 main types of Questionnaires, close-ended, open-ended and hybrid. The advantage for questionnaires is that data can be collected from large sets of participants and the questionnaire can be administered remotely. On the other hand, Questionnaires do not provide a specific view of a users' needs, the categories of questions provide minimal context and there is rarely room for users to convey their needs.

Close-ended questions allow specific responses usually for confirmation. Open-ended questions allow the responder to provide context with their answer, their responses are not specified which allows respondents to generate their own responses. Hybrid Questionnaires combine the two types, open and closed. The questionnaire may start with close ended questions and further progresses to open-ended questions to provide elaboration. [9]

Observation

Observation is an approach to Obtaining information from Clients, Stakeholders, Processes. Once you have Obtained Information (Technical Employee/Employees) you would Audit, Analyze, Make Recommendations about the questions that were answered in the interview process. [9]

B, S R. “Software Engineering Week #3.” 22 Sept. 2022.

An example of using Observation is if a client wants a website to be improved like adding a Search Feature. Once you have Obtained the information you can Analyse it and determine the necessary steps to enhance the website. The following are Advantages and Disadvantages. [10]

Advantages:

Verify Data Is Correct, Assist Interview Team By Asking Follow Up Questions

Disadvantages:

Data Collected has been obtained from a different (Employee)

Surveys

Surveys is a Method to gather data from a massive group of individuals. These Groups can be Categorised as End-Users, Employees, etc. Surveys are usually close ended questions. An example of using Surveys is if you want to gather large amounts of data from End-Users that are using a particular system ex Library System you would want to gather information on Books, Publishers, Features to Add for Library System, etc. The following are Advantages and Disadvantages. [10]

Advantages:

Large Data Set To Use

Disadvantages:

Questions are Fixed

JAD

Joint application development is a paradigm used to gather system requirements. The gathering of requirements occurs throughout several meetings and is derived by a group of participants. A facilitator guides the sessions to improve the quality of ideas and communication between users and developers. The advantages of JAD include faster development times and better requirement gathering through user collaboration leading to more satisfaction with the final result. A facilitator keeps the sessions on track and has productive tasks compared to other approaches. Some disadvantages are found in the informal and open nature of the meetings, which may encourage disagreements and conflicts, leading to poorer quality results (Duggan & Thachenkary, 2004, 399-411).

The JAD process is separated into three distinct phases that need to be completed for the methodology to be successful.

[10]

B, S R. “Software Engineering Week #3.” 22 Sept. 2022.

1. The preparation phase encompasses selecting issues, topics, and participants to include in the session. The participants must be apprised of their inclusion so that they may prepare in advance. A rough agenda must be created to outline the proceedings of the meeting to keep it productive and efficient. Gathering other relevant information is also done in this step, and additionally prepping any other materials such as graphs and charts. The pre-work is an iterative process that continues to refine the itinerary of the sessions. [11]

2. The session leader, also known as a facilitator, leads the work phase. They present the tasks to be discussed and keep conversations on topic. The objective of this phase is to brainstorm and come to a consensus on the requirements. The creation of deliverables happens at the beginning of the session and is refined throughout the discussion. [11]

3. The final phase allows the conclusion of any outstanding issues and questions. [11]

Which and why are you using a specific technique?

The two techniques that we are planning to use for requirement elicitation is interview because it is required for this project and questionnaire. The questionnaire is useful because the company behind Discord has many employees which play to its strengths.
[11]

B, S R. “Software Engineering Week #3.” 22 Sept. 2022.

Interview

Identification of interviewee

Name: Jason Citron

Designation: Discord Head office 444 De Haro Street, San Francisco

Job Description: CEO

Preparation for the interview

Time: 10am(PST)

Date: Wed Sept 28th 2022

Venue: Discord Head office 444 De Haro Street, San Francisco

Duration: 1hr 30 min **Establish**

objectives for the interview

Q#1: What is the purpose of Discord?
Q#2: Who will be using the system?
Q#3: What are your targeted markets?
Q#4: What are some problems with the current system?
Q#5: Does the current system provide enough security for users? **Q#6:** What is the process for responding to security issues?
Q#7: Should each user's data be isolated from the data of other users?
Q#8: What is the potential economic benefit from a successful solution?
Q#9: Are there constraints on execution speed, response time, or throughput?
Q#10: Should Multi Factor Authentication be Mandatory?
Q#11: Should End to End Encryption be implemented in Discord?
Q#12: Does the new system address security concerns?
Q#13: Are there any issues with the proposed system?
Q#14: How easy should it be to port the system from one platform to another?
Q#15: How easy should it be to add features to the system?

Conduct the Interview

Q#1: What is the purpose of Discord?

- Communicate with People around the world through voice and messaging.
- Provide casual environment for friends and colleagues to hang out **Q#2: Who will be using the system?**
- Clients can connect from around the world
- Mostly used by individuals **Q#3: What are your targeted markets?**
- Discord can be used as a chat service and a group call service. The application is in active development with untapped demographics.
- Users who will use nitro features to share and stream on the platform
- Community Building

Q#4: What are some problems with the current system?

- Discord has trouble dealing with spam from malicious users
- Http encryption instead of End to End Encryption
- Multifactor Authentication is optional and left up to the user.

Q#5: Does the current system provide enough security for users?

- Better detection for spam to stop certain attacks (impersonation, links)
- Encryption not high priority, casual atmosphere
- Data has some probability of being compromised **Q#6: What is the process for responding to security issues?**
- Vulnerability reports are usually responded to within 24 hours.
- Severity of issue increases response

Q#7: Should each user's data be isolated from the data of other users?

- Yes Reason we need to stay compliant with certain security frameworks
- Separation means a barrier between data
- Users data is their own

Q#8: What is the potential economic benefit from a successful solution?

- Less likely to see legal action
- Users data is more secure
- Reliability increase in the eyes of the users
- Possibly increase in user retention

Q#9: Are there constraints on execution speed, response time, or throughput?

- Security methods must keep current speed and response time
- VoIP inherently needs to operate smoothly

Q#10: Should Multi Factor Authentication be Mandatory?

- Users are currently in control of this security layer

- Users choose the most efficient method of entry
- Needs to be seamless for user adoption

Q#11: Should End to End Encryption be implemented in Discord?

- E2E encryption is one of the best ways to secure data
- Casualness does not mean less security
- Instills feeling of security and trust from users **Q#12: Does the new system address security concerns?**
- Addresses the security holes of the current system
- Gives tools to server admins to further control over their users to limit spamming
- Provides mandatory methods of account security
- Provides private communications between users on the platform **Q#13: Are there any issues with the proposed system?**
- Users may be opposed to changes in 2FA policies
- Some of the ease of use might change
- New system might take end users

Q#14: How easy should it be to port the system from one platform to another? ● Must be easy as application already exists on multiple platforms

- Windows, MacOS, Android, iOS, iPadOS, Linux, and in web browsers
- Security must be seamlessly integrated

Q#15: How easy should it be to add features to the system?

- Modularity is key
- Security challenges change quickly
- New malicious attacks can occur

Q#16 (Branching from #5): Why is the current security policy changing?

- Previous security policy based on a more casual atmosphere
- Newer VoIP applications coming with stronger security guidelines, competition
- Foster more trust and favorability from users

Q#17 (Branching from #14): What are the priority platforms for the application?

- Preferably system is ready for all platforms
- Windows, Browser, Android, iOS take priority

Q#18 (Branching): Are there any other security issues that have been overlooked?

- Currently all issues are being addressed

Document the interview

Discord provides a casual environment for users from around the world to message, talk and stream with each other. The company sees the application as a hang out for groups of friends and colleagues to get together. The security practices were created to provide an easy entry into the application lending itself to this time of atmosphere. These practices are currently undergoing changes to better match competition. The current software finds it difficult dealing with spam which is used to impersonate staff from Discord and other companies. The current encryption method is HTTP and is not entirely secure against more intelligent attacks. Benefits of fixing these issues are less legal issues, less employee time spent responding to security threats and possible monetary gain from an increased user base.

The new system hopes to alleviate these security flaws by implementing the aforementioned endto-end encryption. This new method of security for communication also positively impacts the perception of the company and user trust. The user data must be kept separate to stay compliant with frameworks. The new system must not impact the response time and speed of the current implementation as the VoIP feature relies on a level of consistency. The Discord application also has many platforms that it is servicing which includes Apple, Microsoft. Android and Linux devices. Therefore any implementations made will need to run on all mentioned platforms for full availability.

Other issues to monitor will be users' response to changes made to two factor authentication policy and other security changes. The added irritation to the login process may impact the happiness of customers. Performance as previously stated must remain at a consistent level so that communication remains clear and efficient. Modularity will also need to be considered as new possible forms of malicious attack will require changes to security policies and modules.

Evaluation of the interview

The interview was successful, all objectives that were created were met within the interview process.

Response #1 - Filled by Johna X. Yue

Responses cannot be edited

Discord App

Department

Data Platform Engineering

Position in Company

Data Scientist

Date

MM DD YYYY

10 / 28 / 2022

Questionnaire

Do you use Discord on your personal devices?

- (●) Yes
- ( ) No

What Are Your Thoughts About Security In The Company?

- ( ) Strongly Agree
- (●) Agree
- ( ) Disagree
- ( ) Strongly Disagree

Do You Like The User Interface?

- (●) Yes
- ( ) No

Does Your Department Have a Written Policy/Procedural Manual?

- (●) Yes
- ( ) No

Are you satisfied with the way security vulnerabilities are disclosed?

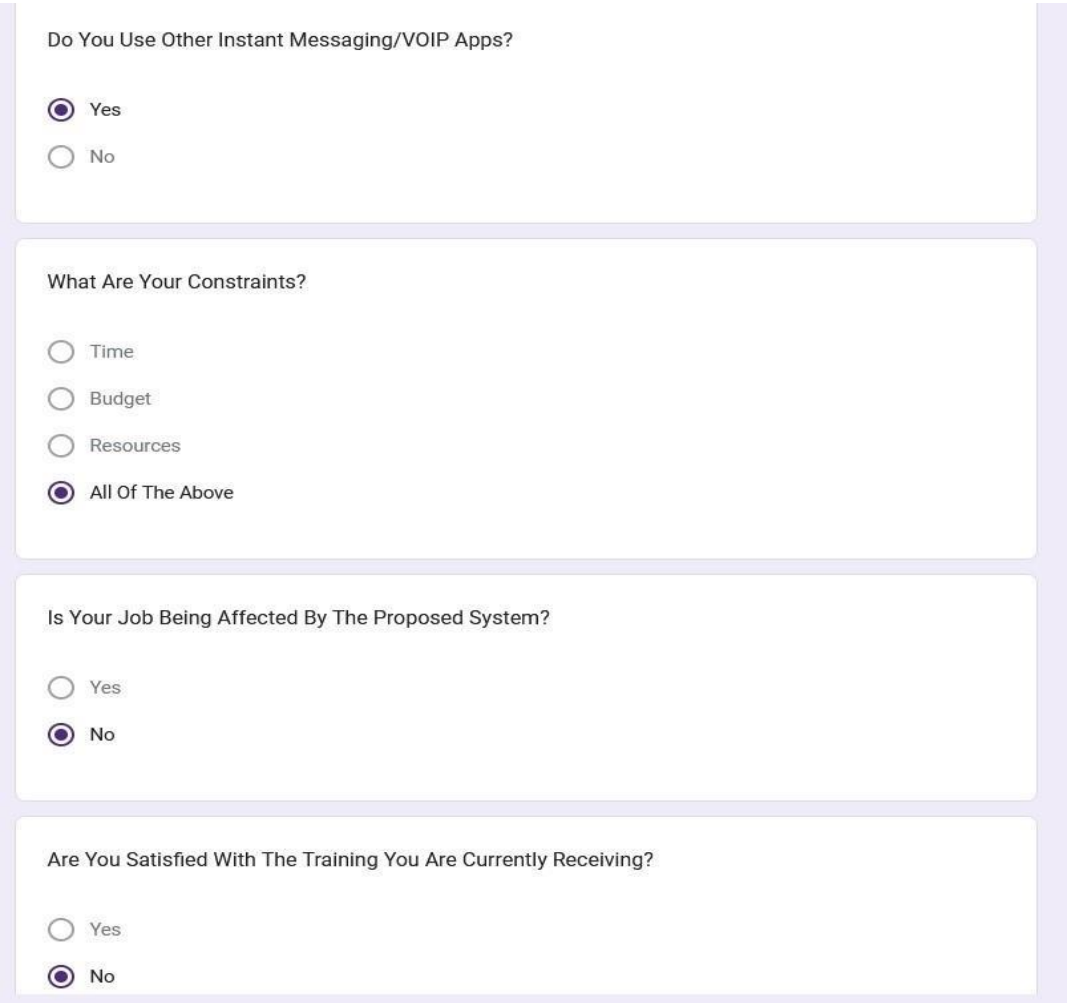
- (●) Yes
- ( ) No

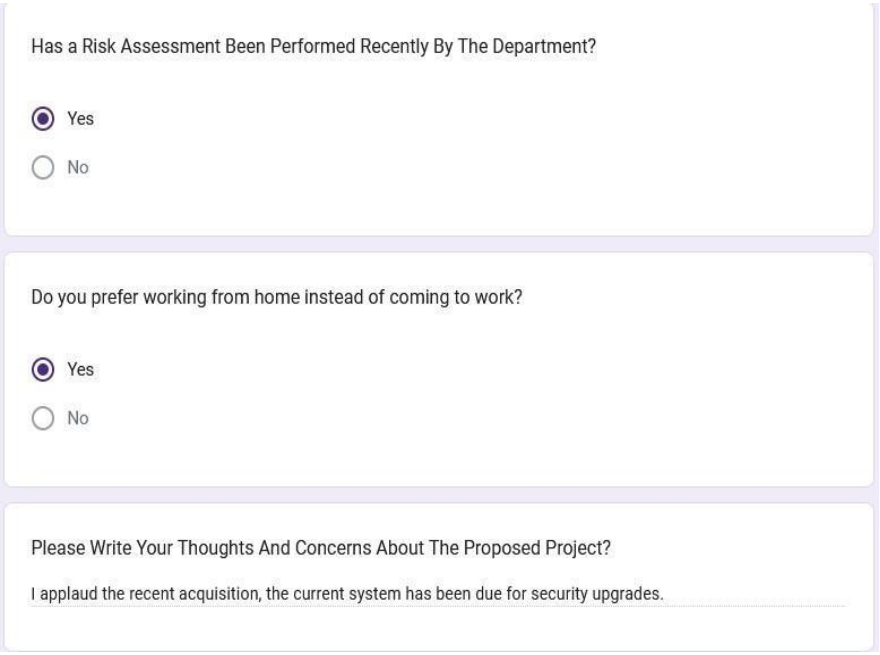
Do You Like How The Permissions Are Laid Out?

- ( ) Yes
- (●) No

How Many Times Have You Used Discord In The Past Month?

- ( ) None
- ( ) Once
- (●) More Than One





Response #2 - Filled by Arthur Horie

Responses cannot be edited

Discord App

Department

IT

Position in Company

IT Technican

Date

MM DD YYYY

10 / 02 / 2022

Questionnaire

Do you use Discord on your personal devices?

- (●) Yes
- ( ) No

What Are Your Thoughts About Security In The Company?

- ( ) Strongly Agree
- (●) Agree
- ( ) Disagree
- ( ) Strongly Disagree

Do You Like The User Interface?

- (●) Yes
- ( ) No

Do you find purpose in your work?

- (●) Yes
- ( ) No

Does Your Department Have a Written Policy/Procedural Manual?

- (●) Yes
- ( ) No

Are you satisfied with the way security vulnerabilities are disclosed?

- ( ) Yes
- (●) No

Do You Like How The Permissions Are Laid Out?

- (●) Yes
- ( ) No

How Many Times Have You Used Discord In The Past Month?

- ( ) None
- ( ) Once
- (●) More Than One

Do You Use Other Instant Messaging/VOIP Apps?

- ○ Yes
- ◉ No

What Are Your Constraints?

- ◉ Time
- ○ Budget
- ○ Resources
- ○ All Of The Above

Is Your Job Being Affected By The Proposed System?

- ○ Yes
- ◉ No

Are You Satisfied With The Training You Are Currently Receiving?

- ◉ Yes
- ○ No

Has a Risk Assessment Been Performed Recently By The Department?

- ◉ Yes
- ○ No

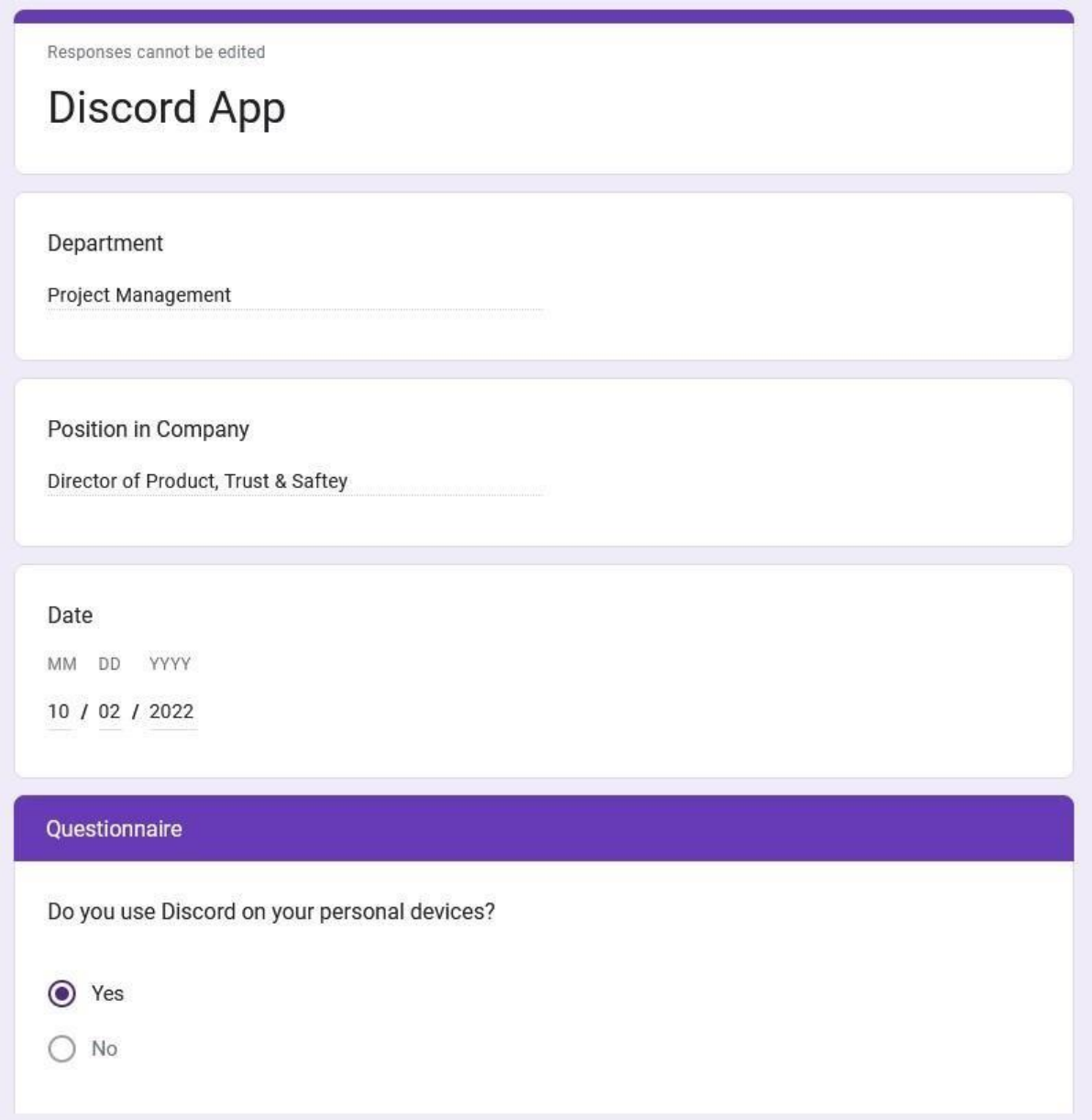
Do you prefer working from home instead of coming to work?

- ◉ Yes
- ○ No

Please Write Your Thoughts And Concerns About The Proposed Project?

I have no concerns.

Response #3 - Filled by Elbert Belone



Responses cannot be edited

Discord App

Department

Project Management

Position in Company

Director of Product, Trust & Saftey

Date

MM DD YYYY

10 / 02 / 2022

Questionnaire

Do you use Discord on your personal devices?

- (●) Yes
- ( ) No

What Are Your Thoughts About Security In The Company?

- (●) Strongly Agree
- ( ) Agree
- ( ) Disagree
- ( ) Strongly Disagree

Do You Like The User Interface?

- (●) Yes
- ( ) No

Do you find purpose in your work?

- (●) Yes
- ( ) No

Does Your Department Have a Written Policy/Procedural Manual?

- (●) Yes
- ( ) No

Are you satisfied with the way security vulnerabilities are disclosed?

- (●) Yes
- ( ) No

Do You Like How The Permissions Are Laid Out?

- ( ) Yes
- (●) No

How Many Times Have You Used Discord In The Past Month?

- ( ) None
- ( ) Once
- (●) More Than One

Do You Use Other Instant Messaging/VOIP Apps?

- (●) Yes
- ( ) No

What Are Your Constraints?

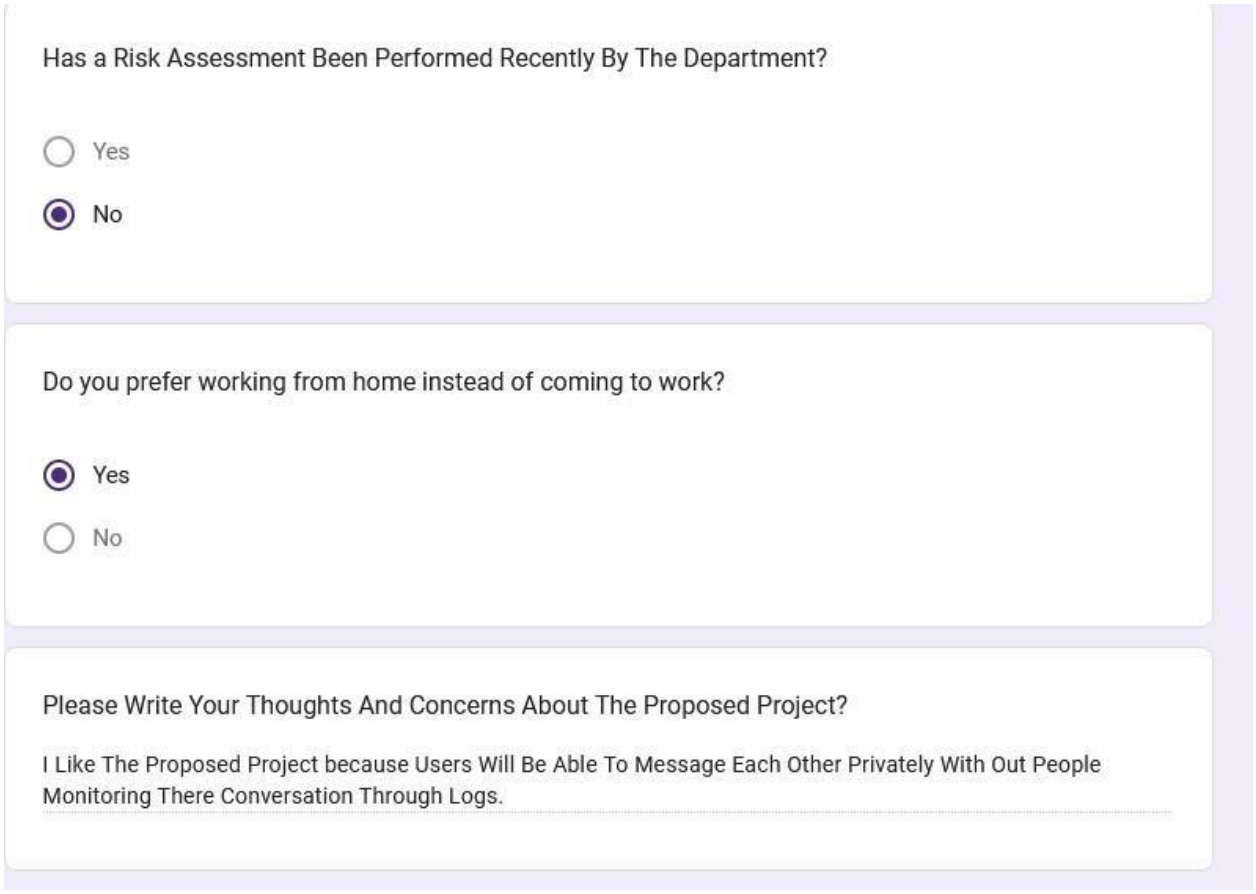
- ( ) Time
- (●) Budget
- ( ) Resources
- ( ) All Of The Above

Is Your Job Being Affected By The Proposed System?

- ( ) Yes
- (●) No

Are You Satisfied With The Training You Are Currently Receiving?

- (●) Yes
- ( ) No



Response #4 - Filled by Wesley Wellson

Responses cannot be edited

Discord App

Department

Talent

Position in Company

Vice President Of People

Date

MM DD YYYY

10 / 02 / 2022

Questionnaire

Do you use Discord on your personal devices?

- (●) Yes
- ( ) No

What Are Your Thoughts About Security In The Company?

- ○ Strongly Agree
- ◉ Agree
- ○ Disagree
- ○ Strongly Disagree

Do You Like The User Interface?

- ◉ Yes
- ○ No

Do you find purpose in your work?

- ◉ Yes
- ○ No

Does Your Department Have a Written Policy/Procedural Manual?

- ◉ Yes
- ○ No

Are you satisfied with the way security vulnerabilities are disclosed?

- ○ Yes
- ◉ No

Do You Like How The Permissions Are Laid Out?

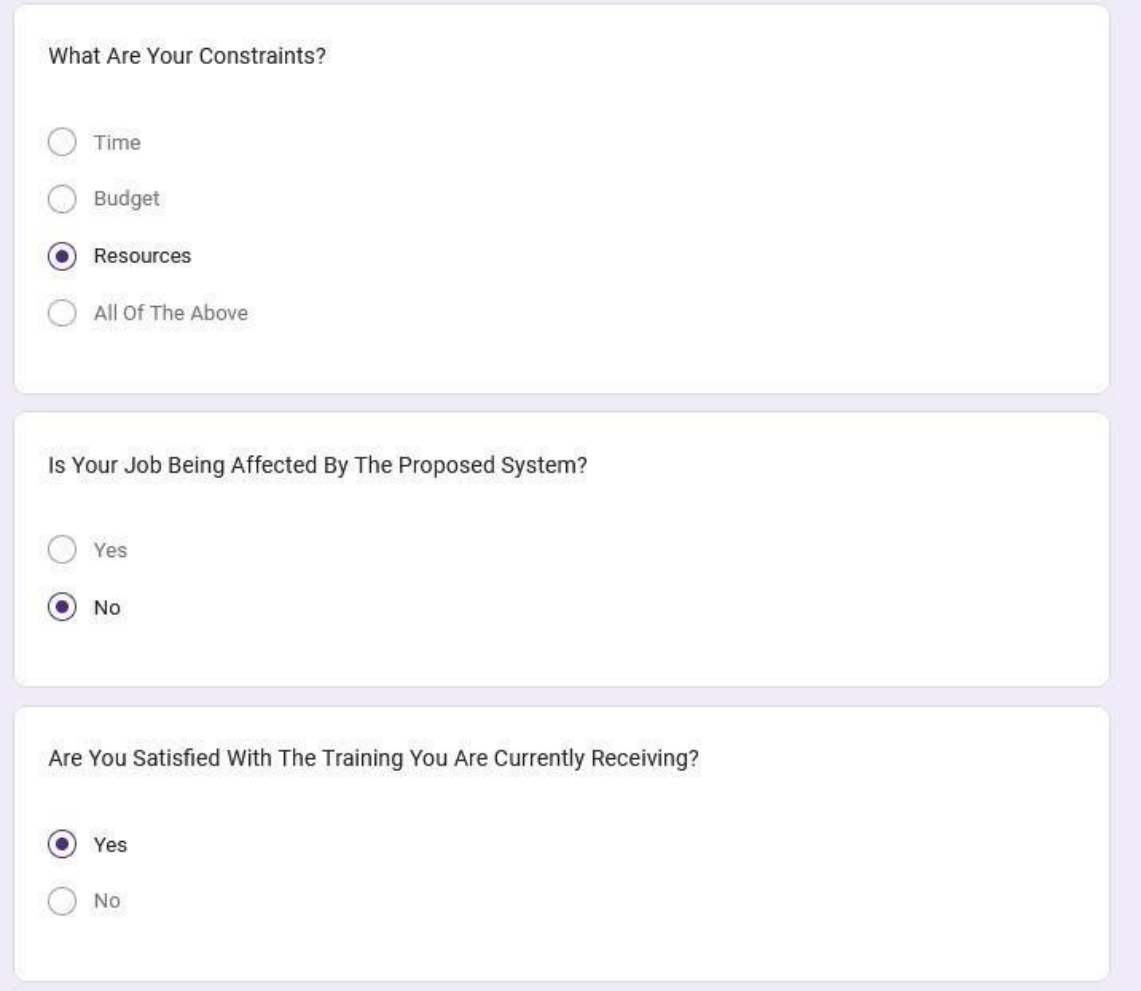
- ○ Yes
- ◉ No

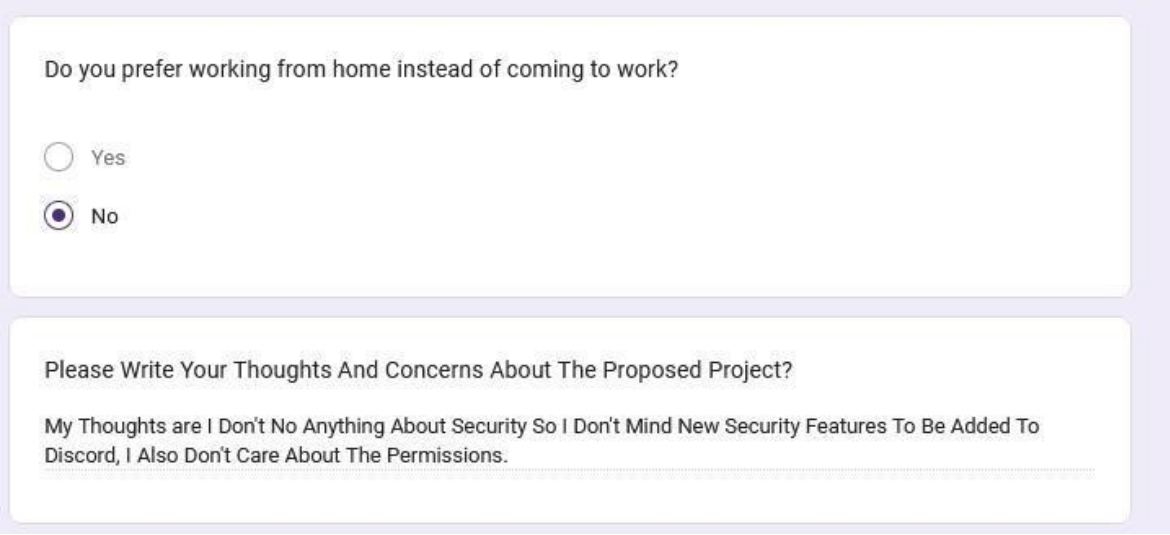
How Many Times Have You Used Discord In The Past Month?

- ○ None
- ○ Once
- ◉ More Than One

Do You Use Other Instant Messaging/VOIP Apps?

- ◉ Yes
- ○ No





Response #5 - Filled by Wyatt Gregorash

Responses cannot be edited

Discord App

Department

Infrastructure

Position in Company

STAFF SECURITY SOFTWARE ENGINEER, DETECTION AND RESPONSE

Date

MM DD YYYY

10 / 02 / 2022

Questionnaire

Do you use Discord on your personal devices?

- (●) Yes
- ( ) No

What Are Your Thoughts About Security In The Company?

- (●) Strongly Agree
- ( ) Agree
- ( ) Disagree
- ( ) Strongly Disagree

Do You Like The User Interface?

- (●) Yes
- ( ) No

Do you find purpose in your work?

- (●) Yes
- ( ) No

Does Your Department Have a Written Policy/Procedural Manual?

- (●) Yes
- ( ) No

Are you satisfied with the way security vulnerabilities are disclosed?

- (●) Yes
- ( ) No

Do You Like How The Permissions Are Laid Out?

- (●) Yes
- ( ) No

How Many Times Have You Used Discord In The Past Month?

- ( ) None
- ( ) Once
- (●) More Than One

Do You Use Other Instant Messaging/VOIP Apps?

- ( ) Yes
- (●) No

What Are Your Constraints?

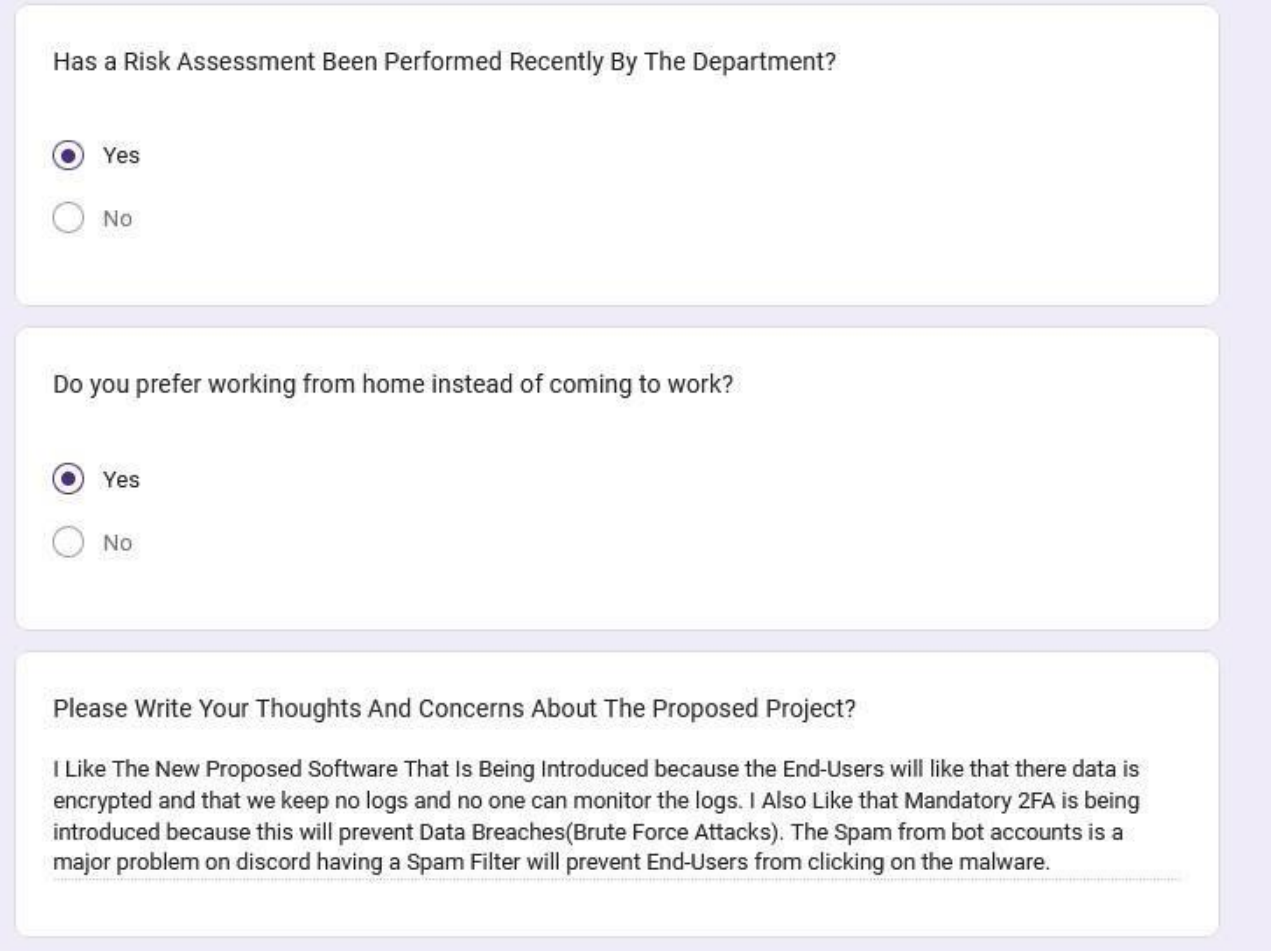
- ○ Time
- ○ Budget
- ◉ Resources
- ○ All Of The Above

Is Your Job Being Affected By The Proposed System?

- ○ Yes
- ◉ No

Are You Satisfied With The Training You Are Currently Receiving?

- ◉ Yes
- ○ No



Software Process Models

Waterfall

The waterfall process model is an older model. The model is used when the requirements are known. Also, the requirements that the client gives you are firm meaning the requirements do not get adjusted during the software development cycle.[8]

The phases of the Waterfall model are:

- Requirements analysis and definition
- System and software design
- Implementation and unit testing

[8] B, S R. "Software Engineering Week #4

-2 4 Oct. 2022.

- Integration and system testing

- Operation and maintenance

Prototyping

The prototyping process model is a fast development of a system. The model focuses on assisting clients and the software team with grasping the requirements for the system. Also, with Prototyping you are lessening the chance of problems occurring during the development of the system.[9]

The Prototyping Process are:

- Establish Prototype Objective - Prototyping Plan
- Define Prototype Functionality - Outline Definition
- Develop Prototype - Executable Prototype
- Evaluate Prototype - Evaluation Report

Prototyping two approaches in Software Process:

- Evolutionary Prototyping
- Throw-away Prototyping

Evolutionary Prototyping is when a mock-up is created. The functions of the software are then polished through different stages up until the final system.

Throw-away Prototyping is a hands-on approach to the development of the software system.

Using this way helps find out requirement issues with the software system and get rid of them.

After this system is Finished being developed you apply other development processes to it. **V**

Model

The V-Model functions in a way where processes execute in sequential manner forming a V-shape. This model is also known as the verification and validation model. The processes base themselves on association of a testing phase for each correlating development stage. Development of each step is associated with the testing phase. Next phases start only after completion of the previous phase. There are relatively an equal amount of pros and cons for the use of a V-Model. It is advantageous

[9] B, S R. "Software Engineering Week #4

-2 4 Oct. 2022.

because there is an emphasis on planning for verification and validation of the product in the early stages of product development, while ensuring that each deliverable is testable. On the other hand, the V-Model does not easily handle concurrent events, nor is it good at handling deliverables in reference to phases or iterations. At the same

time, there is no analysis for risk within this model.[10]

Incremental

This model is also known as the Successive version model. Within this model, requirements are broken down into modules that can be incrementally developed and delivered. The module's requirements are ranked and delivered with increments. This allows the developmental team to construct high-risk functions first while providing an operational product with each release. Initially, a simple functional system is implemented with basic features, the initial simple system is then delivered to the customer. Upon completion of the initial working system, many iterations and versions are implemented and delivered to the customer until a desired system is implemented.
[11]**Spiral**

The spiral model is the only process model that incorporates risk evaluation and management. Each risk is identified and categorised from highest to lowest risk upon which a prototype is created starting from the most high-risk. Each prototype is developed using the waterfall model and if a risk evaluates as successful then the next risk is assessed. If any of the risks turn out to be unsolvable then the project is cancelled. Rapid prototyping allows constant feedback and impossible projects can be terminated at relatively little cost. This process model is perfect for medium to high risk projects where requirements aren’t clear and changes are to be expected.

Unified Process

This process model is based on the contributions to Uniform Modeling Language which emphasises visually designing systems. UP can be broken down into three perspectives: (a) a dynamic perspective showing phases of the possess model over time, (b) a static perspective showing process activities, and {c} a practice perspective that suggests good practice. Unlike the standard 6-7 phases that SDLC has, the unified process model separates into four phases:

[10] B, S R “Software Engineering Week #4-2 4 Oct. 2022.

[11] B, S R. “Software Engineering Week #4

-2 4 Oct. 2022.

Inception, Elaboration, Construction and Transition. Construction uniquely is made up of the design, implementation,and testing phases commonly found in other process models.

Every phase has the opportunity to include many different disciplines such as design, testing and project management. The unified process model is an iterative and incremental approach allowing phase repetition which culminates in a prototype at the end of each cycle. Requirements may change between cycles but remain fixed within the development process of a

cycle.[12]

B, S R. “Software Engineering Week #5 4 Oct. 2022.

[12]

Agile Development

Definition

Agile methodology was created to address overhead issues and the increased risk of failure for projects that have extended development periods. The main goal for Agile Development is to develop systems quickly so that a project may be iteratively produced. Another pillar of agile is the people over plan approach which leads to better communication and customer satisfaction.

Benefits

There are many benefits when using agile development which include having a product ready at the end of every sprint and increment. It integrates changes made during the process which makes the overall flexibility of the project much greater. Agile development also promotes better and consistent communication with the customer. This framework leads to the product being produced quicker which means that it has a greater chance of success. There are many different models based on the agile methodology which can be chosen to fit the project.[12]

Scrum

Definition

Scrum is based on the agile methodology meaning it incorporates the incremental approach to project management in the form of ready software every two to four weeks referred to as a sprint.

[12] B, S R. “Software Engineering Week #5 4 Oct. 2022.

B, S R. “Software Engineering Week #5 4 Oct. 2022.

The Scrum process involves self-organising teams which take customer requirements and use them to produce a product backlog. Each sprint is designated a sprint goal which is used by the

team to build their sprint backlog.[18]

[18]

Stages

Sprint Planning Meeting

This first part of this stage involves all participants of the project including the scrum master, scrum team, product owner. This is where the product backlog is generated and the first sprint goal is chosen. The second part involves only the scrum master and team where the sprint goal is used to select the requirements in the form of a sprint backlog.

Sprint

B, S R. “Software Engineering Week #5 4 Oct. 2022.

In Scrum a sprint denotes a four week iteration that is completed when product functionality is achieved. During this process the team is isolated and only the team is allowed to change anything on the sprint backlog. Each sprint cycle begins with a daily scrum meeting.

Daily Scrum

A daily meeting between the scrum team and scrum master that tries to get a review of what each member has worked on yesterday and what they plan to work on presently. This process is a way for the team to make commitments to each other and the scrum master to drive personal responsibility.

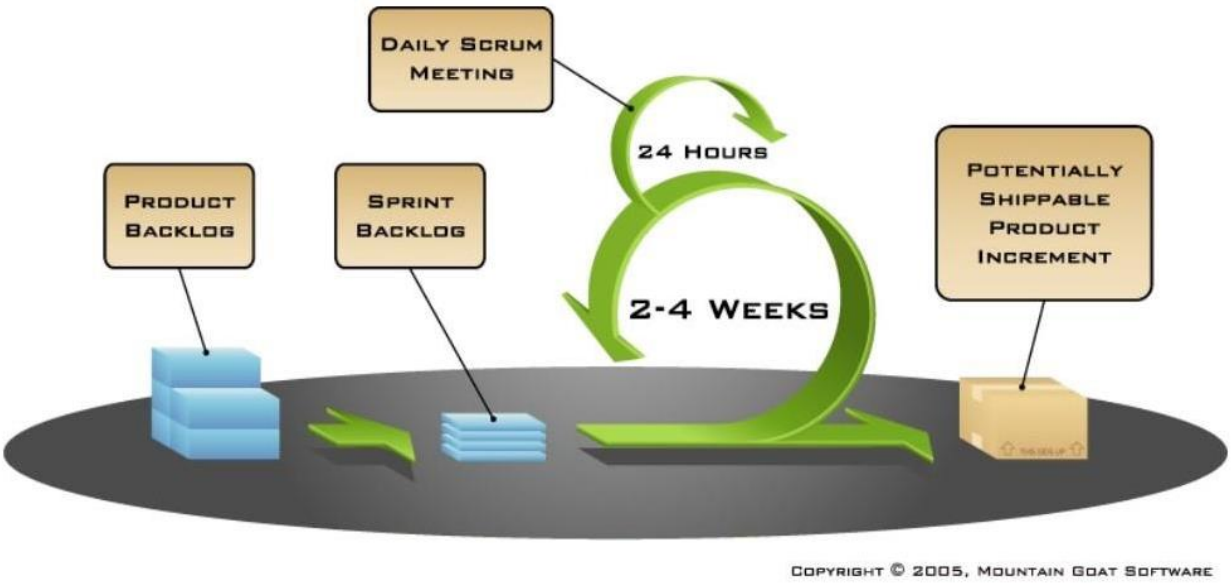
Sprint Review Meeting

Each sprint ends with a sprint review meeting which includes customers, product owner, management and other related individuals. The team makes a presentation detailing progress made on the product. After the closing of the sprint review meeting a retrospective meeting occurs between the scrum team where feedback is exchanged.[19]

19

Scrum Diagram[20]

B, S R. “Software Engineering Week #5 4 Oct. 2022.



Prof. Syed Raza Bashir

20

B, S R. “Software Engineering Week #5 4 Oct. 2022.

Product Backlog

Item #	Description
Very High	
-	**Enforce Mandatory Multifactor Authentication**
1	Setup In browser
2	Setup In application
3	Create UI for 2FA input(Browser/Application)
4	New User Signs Up/Login
5	Current User Logins
6	Select 2FA Option(New User/Current)
7	Description what 2FA Method you want(Sms, Authenticator)
8	Receive 2FA Authentication
9	Generate Backup Key/Code
-	**End to End Encryption**

10	Create Server
11	Use Chat Protocol
12	Key generation
13	Verify Keys
14	Process connection
15	Terminate connection
16	Verify Protocol Compatibility Ios, Devices(Android, Linux, Windows)
High	
-	**Spam filtering**

17	Time Between Messages
18	Algorithm Characterise Message
19	Custom language filters
20	Limit file sharing
21	Change direct messaging permissions

-	**Permission controls**
22	Separate manage roles into specific permissions
23	Separate manage channels into specific permissions
24	Supplement current default permissions
25	Create necessary UI elements for new permissions
26	Show only generated links
Medium	
27	Check to see if Mandatory two factor authentication is enabled/working
28	Verify implementation of end to end encryption
29	Examine to see if permission controls are operating as intended
30	Synchronising communication between users while using end to end encryption.
31	Inspect to see if spam filtering is enabled/working
Low	

32	Monitor performance impacts.
33	Optimise Response time of application.
34	Making sure the Certificate of the website is up to date and valid.
35	Improve employee security training.
36	Ensure consistency across new UI elements

Scrum

Scrum Sprint #1 - Sprint Goal

Setup and Enforce Mandatory MultiFactor Authentication

Item #	**Description**
-	**Enforce Mandatory Multifactor Authentication**
1	Setup In Browser
2	Setup In application
3	Create UI for 2FA input
4	New User Signs Up/Login
5	Current User Logins
6	Select 2FA Option(New User/Current)

7.	Description what 2FA Method you want(Sms, Authenticator)
8	Receive 2FA Authentication
9	Generate Backup Key/Code
27	Check to see if Mandatory two factor authentication is enabled/working

Product Backlog (After sprint 1)

Item #	Description
Very High	
-	**Enforce Mandatory Multifactor Authentication**
1	~~Setup In browser~~
2	~~Setup In application~~
3	~~Create UI for 2FA input(Browser/Application)~~
4	~~New User Signs Up/Login~~
5	~~Current User Logins~~
6	~~Select 2FA Option(New User/Current)~~

7	Description what 2FA Method you want(Sms, Authenticator)
8	Receive 2FA Authentication
9	Generate Backup Key/Code
-	**End to End Encryption**
10	Create Server
11	Use Chat Protocol
12	Key generation
13	Verify Keys
14	Process connection
15	Terminate connection
16	Verify Protocol Compatibility Devices(Android, Ios, Linux, Windows)
High	
-	**Spam filtering**

17	Time Between Messages
18	Algorithm Characterise Message
19	Custom language filters

20	Limit file sharing
21	Change direct messaging permissions
-	**Permission controls**
22	Separate manage roles into specific permissions
23	Separate manage channels into specific permissions
24	Supplement current default permissions
25	Create necessary UI elements for new permissions
26	Show only generated links
Medium	

27	~~Check to see if Mandatory two factor authentication is enabled/working~~
28	Verify implementation of end to end encryption
29	Examine to see if permission controls are operating as intended
30	Synchronising communication between users while using end to end encryption.
31	Inspect to see if spam filtering is enabled/working
Low	
32	Monitor performance impacts.
33	Optimise Response time of application.
34	Making sure the Certificate of the website is up to date and valid.
35	Improve employee security training.
36	Ensure consistency across new UI elements

Scrum Sprint #2 - Sprint Goal

Implement End-to-End Encryption

Item #	Description
-	**End to End Encryption**
10	Create Server
11	Use Chat Protocol
12	Key generation
13	Verify Keys
14	Process connection
15	Terminate connection
16	Ios, Verify Protocol Compatibility Devices(Android, Linux, Windows)
32	Monitor performance impacts.
30	Synchronising communication between users while using end to end encryption.
28	Verify implementation of end to end encryption
33	Optimise Response time of application.

Product Backlog (After sprint 2)

Item #	Description

Very High	
-	**Enforce Mandatory Multifactor Authentication**
1	~~Setup In browser~~
2	~~Setup In application~~
3	~~Create UI for 2FA input(Browser/Application)~~
4	~~New User Signs Up/Login~~
5	~~Current User Logins~~
6	~~Select 2FA Option(New User/Current)~~
7	~~Description what 2FA Method you want(Sms, Authenticator)~~
8	~~Receive 2FA Authentication~~
9	~~Generate Backup Key/Code~~
-	**End to End Encryption**
10	~~Create Server~~
11	~~Use Chat Protocol~~

12	~~Key generation~~
13	~~Verify Keys~~
14	~~Process connection~~
15	~~Terminate connection~~
16	~~Ios,~~ ~~Verify Protocol Compatibility Devices(Android, Linux, Windows)~~
High	
-	**Spam filtering**

17	Time Between Messages
18	Algorithm Characterise Message
19	Custom language filters
20	Limit file sharing
21	Change direct messaging permissions
-	**Permission controls**
22	Separate manage roles into specific permissions

23	Separate manage channels into specific permissions
24	Supplement current default permissions
25	Create necessary UI elements for new permissions
26	Show only generated links
Medium	
27	~~Check to see if Mandatory two factor authentication is enabled/working~~
28	~~Verify implementation of end to end encryption~~
29	Examine to see if permission controls are operating as intended
30	~~Synchronising communication between users while using end to end encryption.~~
31	Inspect to see if spam filtering is enabled/working
Low	
32	~~Monitor performance impacts.~~
33	~~Optimise Response time of application.~~

34	Making sure the Certificate of the website is up to date and valid.
35	Improve employee security training.
36	Ensure consistency across new UI elements

Scrum Sprint #3 - Sprint Goal

Improving application security measures through permission control and spam filtering

Item #	**Description**
-	**Spam filtering**
17	Time Between Messages
18	Algorithm Characterise Message
19	Custom language filters
20	Limit file sharing
21	Change direct messaging permissions
-	**Permission controls**
22	Separate manage roles into specific permissions
23	Separate manage channels into specific permissions
24	Supplement current default permissions

25	Create necessary UI elements for new permissions
26	Show only generated links
29	Examine to see if permission controls are operating as intended
31	Inspect to see if spam filtering is enabled/working
34	Making sure the Certificate of the website is up to date and valid.
35	Improve employee security training.
36	Ensure consistency across new UI elements

Product Backlog (After sprint 3)

Item #	**Description**
Very High	
-	**Enforce Mandatory Multifactor Authentication**
1	~~Setup In browser~~
2	~~Setup In application~~
3	~~Create UI for 2FA input(Browser/Application)~~

4	New User Signs Up/Login
5	Current User Logins
6	Select 2FA Option(New User/Current)
7	Description what 2FA Method you want(Sms, Authenticator)
8	Receive 2FA Authentication
9	Generate Backup Key/Code
-	**End to End Encryption**
10	Create Server
11	Use Chat Protocol
12	Key generation
13	Verify Keys
14	Process connection
15	Terminate connection
16	Verify Compatibility Protocol Devices(Android, Ios, Linux, Windows)

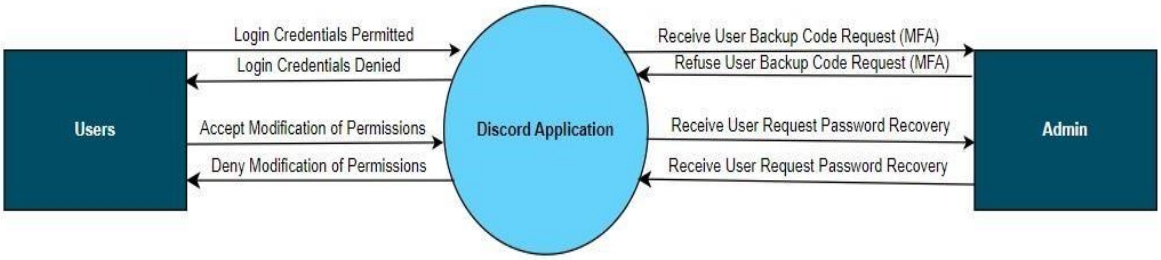
High	
-	**Spam filtering**

17	~~Time Between Messages~~
18	~~Algorithm Characterise Message~~
19	~~Custom language filters~~
20	~~Limit file sharing~~
21	~~Change direct messaging permissions~~
-	**Permission controls**
22	~~Separate manage roles into specific permissions~~
23	~~Separate manage channels into specific permissions~~
24	~~Supplement current default permissions~~
25	~~Create necessary UI elements for new permissions~~
26	~~Show only generated links~~

Medium	
27	~~Check to see if Mandatory two factor authentication is enabled/working~~
28	~~Verify implementation of end to end encryption~~
29	~~Examine to see if permission controls are operating as intended~~
30	~~Synchronising communication between users while using end to end encryption.~~
31	~~Inspect to see if spam filtering is enabled/working~~
Low	
~~32~~	~~Monitor performance impacts.~~
33	~~Optimise Response time of application.~~
34	~~Making sure the Certificate of the website is up to date and valid.~~
35	~~Improve employee security training.~~

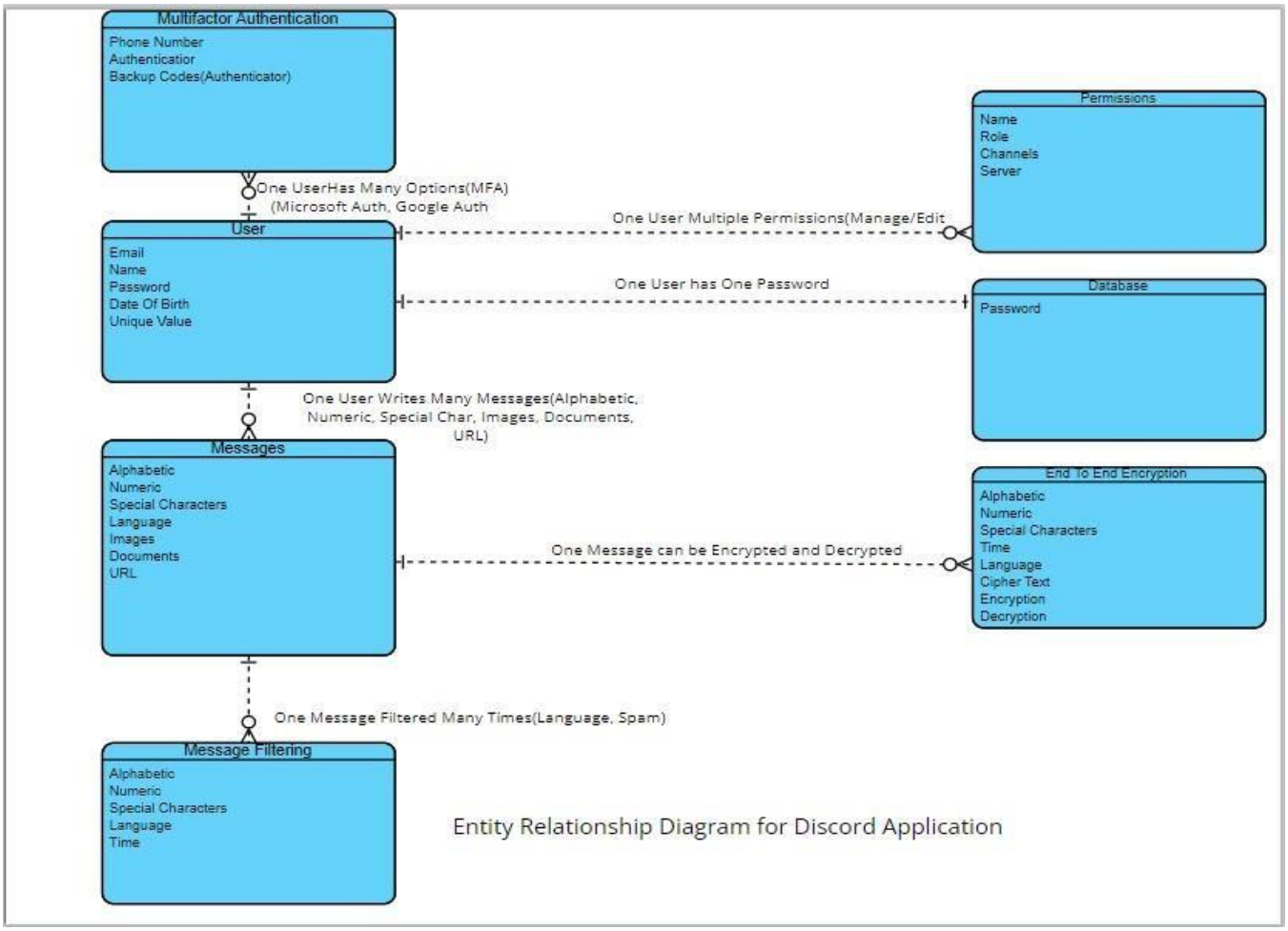
36	~~Ensure consistency across new UI elements~~

System Context Diagram



System Context Diagram for Discord Application

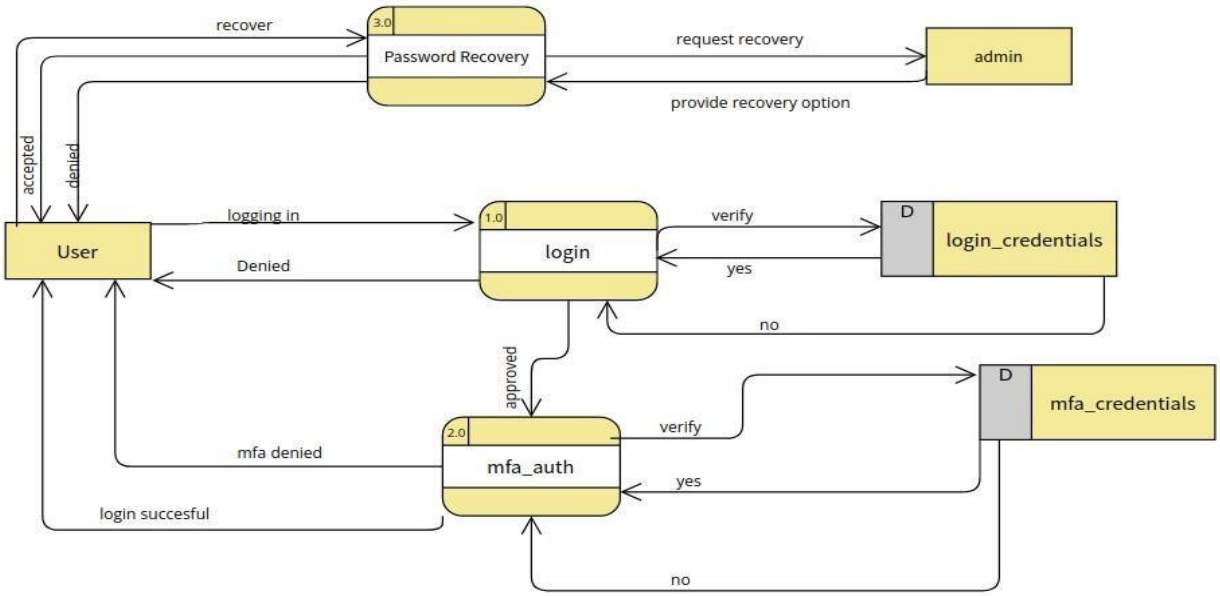
Entity Relationship Diagram



Entity Relationship Diagram for Discord Application

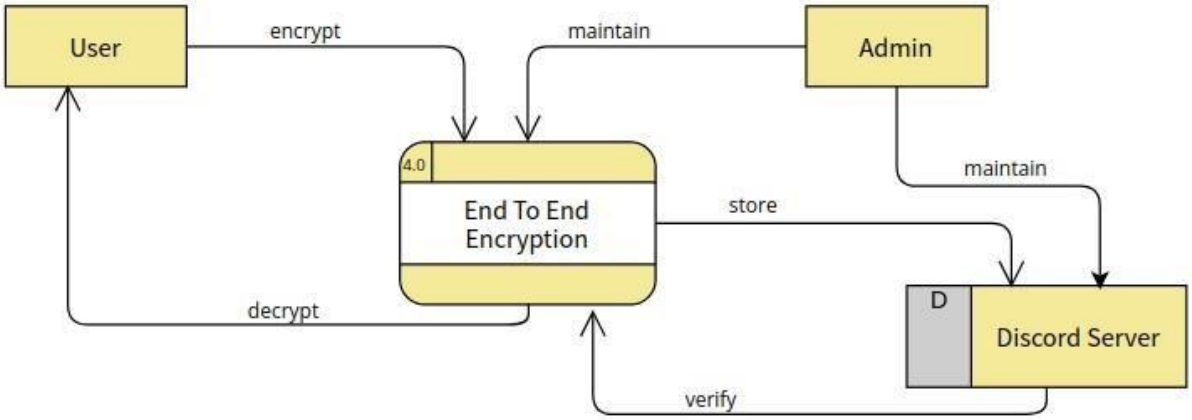
Data Flow Diagrams

Data Flow Diagram for Sprint #1



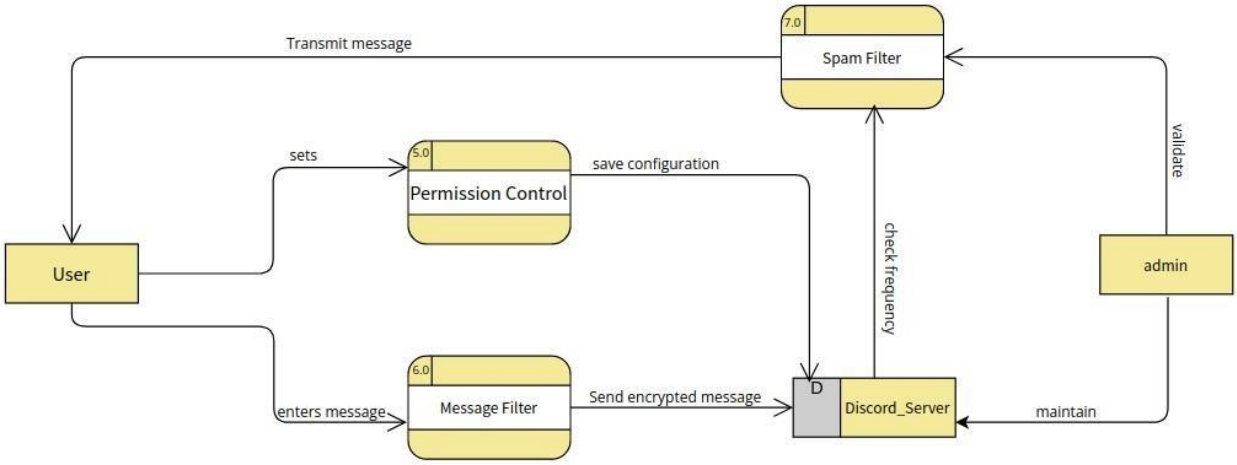
First Level DFD for Sprint #1

Data Flow Diagram for Sprint #2



First Level DFD for Sprint #2

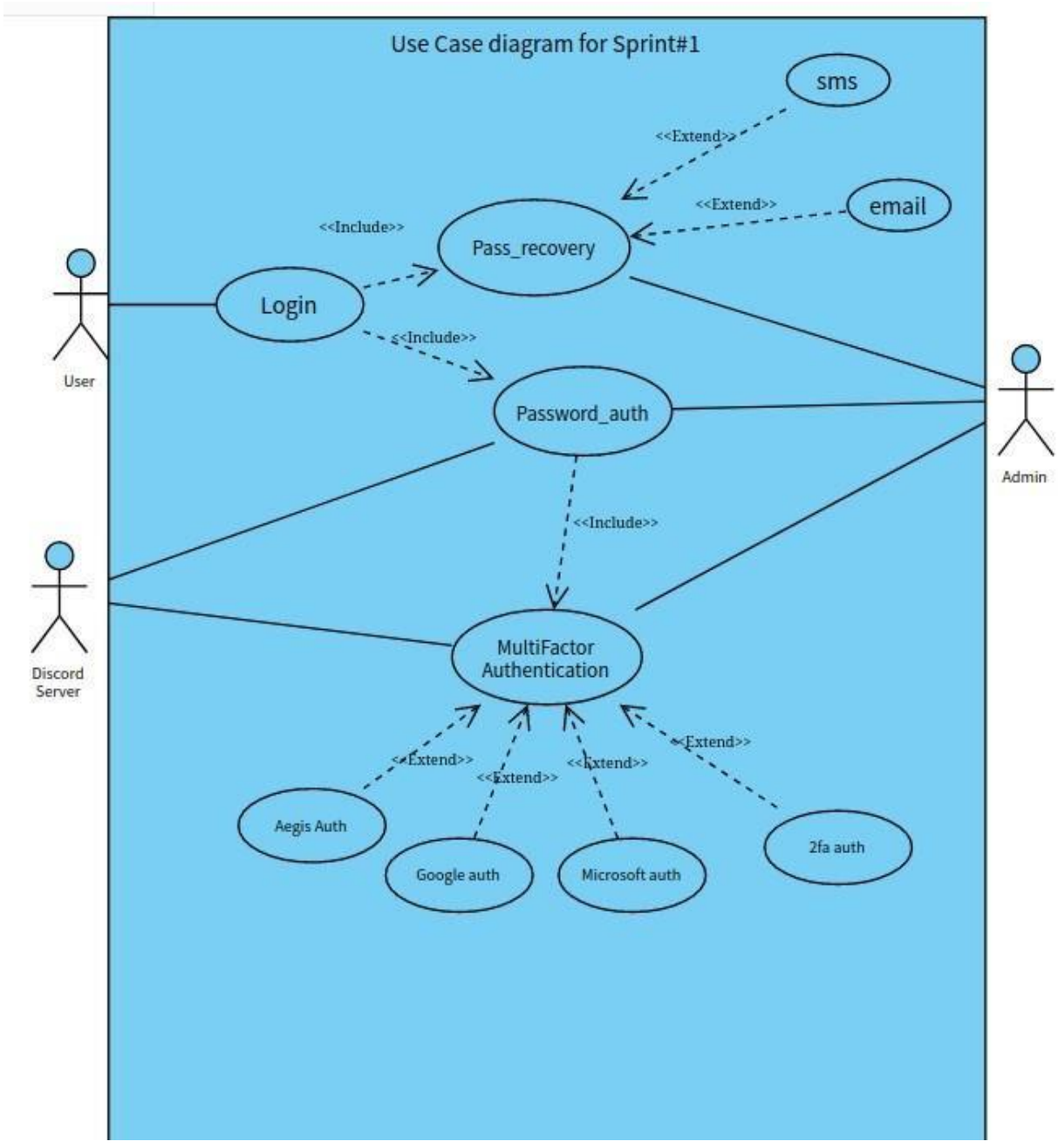
Data Flow Diagram for Sprint #3



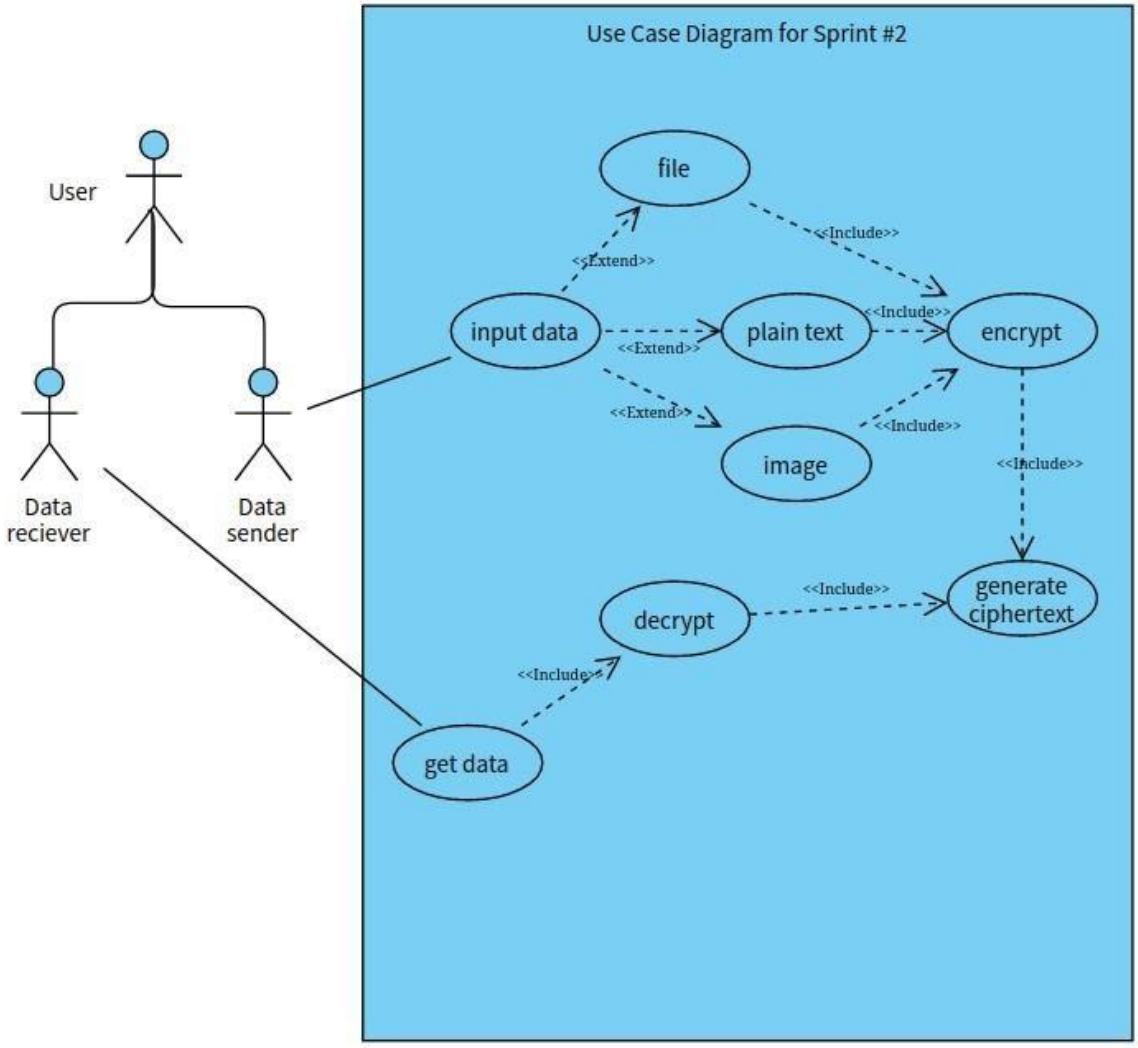
First Level DFD for Sprint #3

Use Case Diagrams

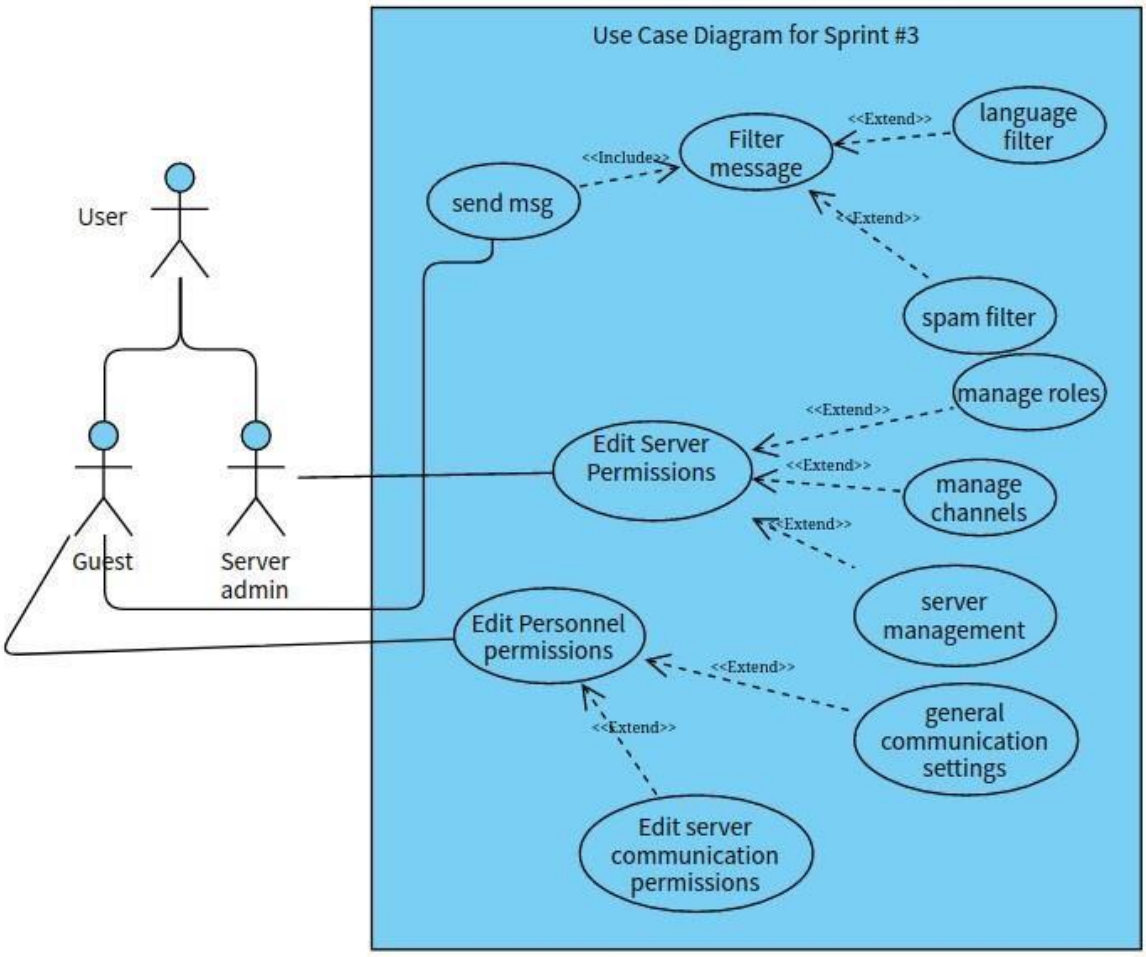
Use case diagram for Sprint #1



Use case diagram for Sprint #2



Use case diagram for Sprint #3

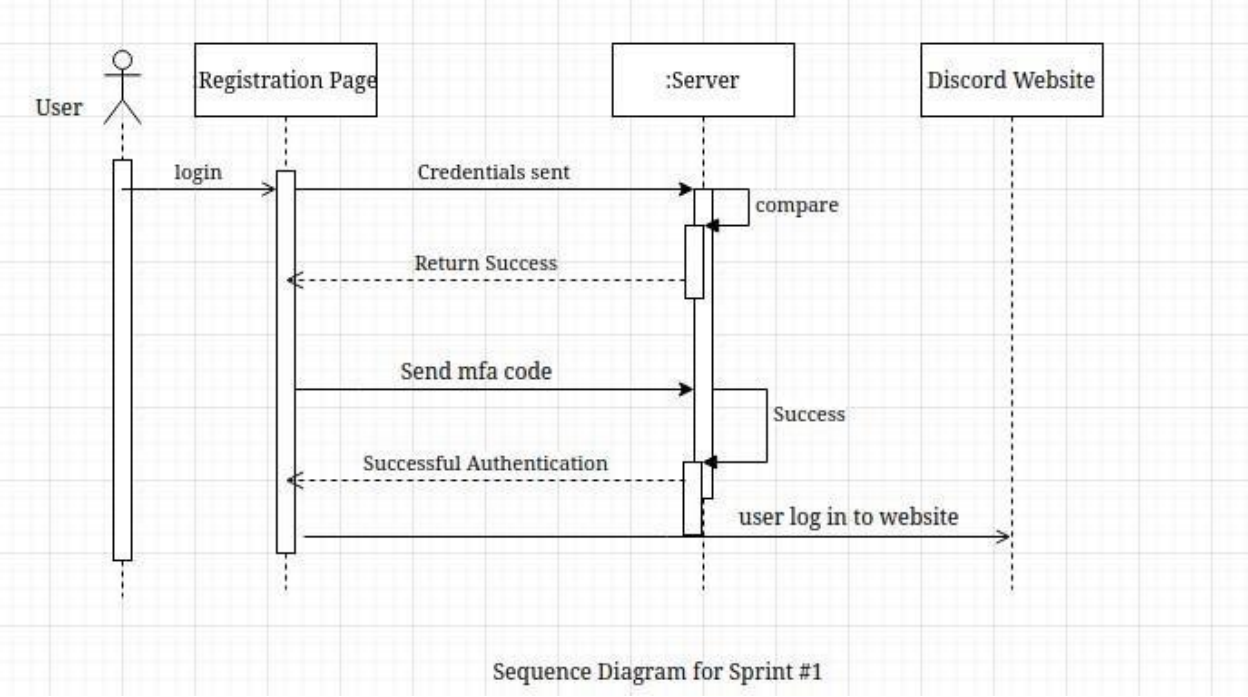


Sequence Diagrams

Use Case Number: 01	**Use Case Name:** Logging in with Multi Factor Authentication
Brief Description: The user logins in with their credentials which will prompt them for a MFA(Multi Factor Authentication) code which will also be authenticated.	
Actors: Users, Admin, Discord Server	
Frequency of Execution: On every login attempt.	

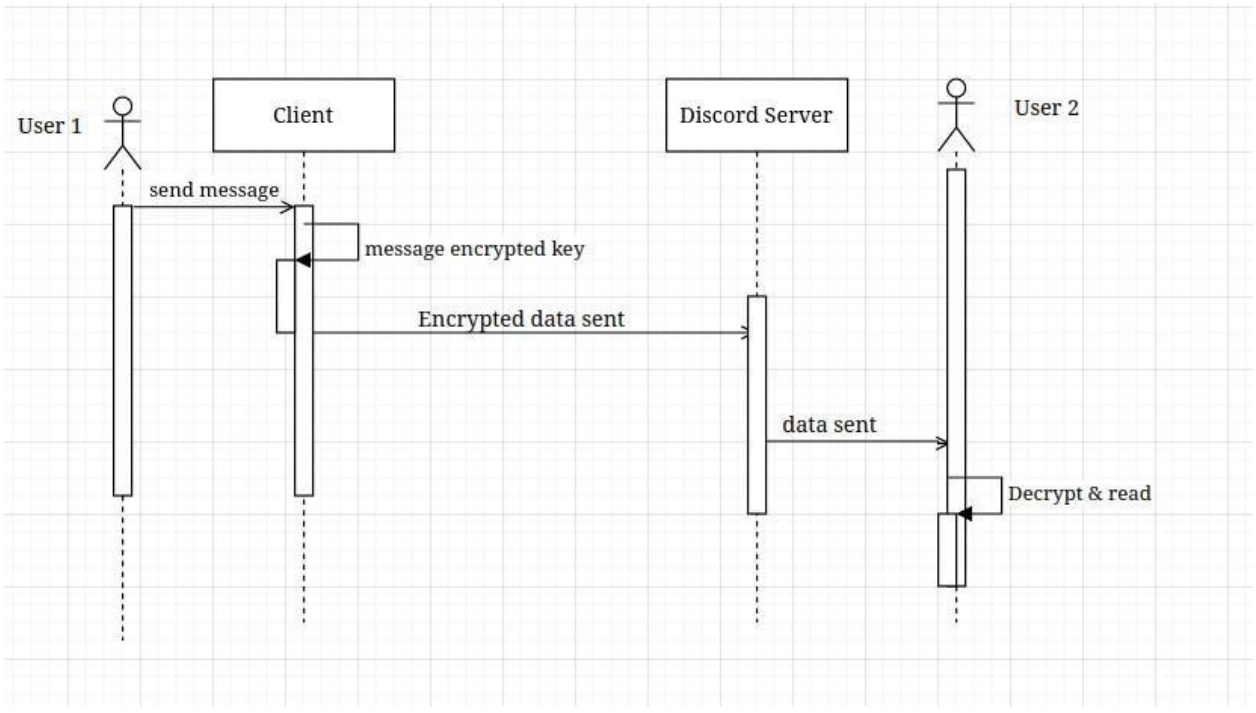
Scalability: Run on every user login attempt.
Criticality: Essential. Not running this will compromise security. Will negate benefits of Multi Factor Authentication implementation.
Primary Path: The following sequence is carried out for every user logging into their account. 1. Customer enters username and password. 2. Login credentials sent to the server for verification. 3. Enter login credentials and compare to stored values. 4. Return successful. 5. Prompt generated for multi factor authentication. 6. User provides code. 7. Code sent to authentication. 8. Server sends back successful authentication message. 9. User logged in.

Sequence Diagram for Sprint #1



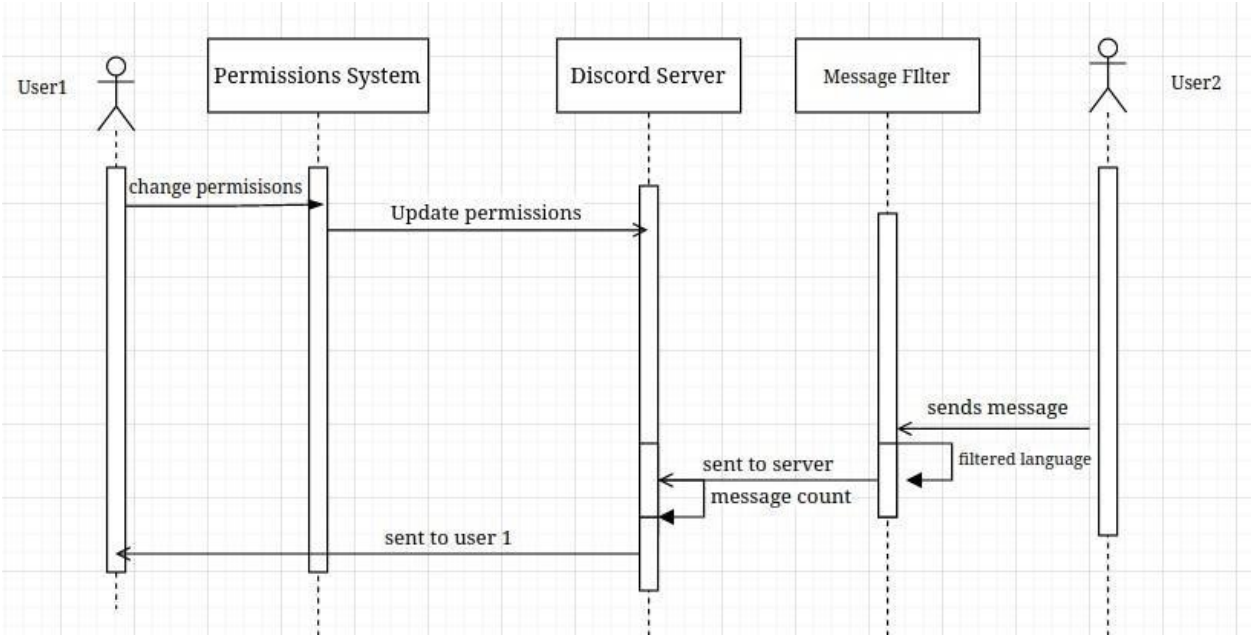
Use Case Number: 02	**Use Case Name:** End to End Encryption.
Brief Description: Whenever a user sends data, that data is encrypted on sender side and decrypted on the receiver's end.	
Actors: Sender, and Receiver	
Frequency of Execution: Everytime message is sent.	
Scalability: One instance running with each unique user to user communication.	
Criticality: Essential. Required if the user sends a message that is using end-to-end encryption so that privacy is sustainable.	
Primary Path: The following sequence is carried out for every time a user sends a message. 1. User sends data to another user. 2. Data is encrypted at the client. 3. User data is sent to the server. 4. Data remains encrypted. 5. Data then sent to the intended receiver. 6. Receiving user decryptes message. 7. Receiving user can read the message.	

Sequence Diagram for Sprint #2



Use Case Number: 03	**Use Case Name:** Spam Filtering
Brief Description: Increasing security permission controls by implementing better filtering and permission controls for users to administer their servers.	
Actors: Users: Guest, Server Admin, Permission System	
Frequency of Execution: When changes to permissions are made or a message is sent by the user.	
Scalability: Permissions can be scaled by the user at any given moment.	
Criticality: Pivotal. Filtering and permissions controls allow better security for users.	
Primary Path: 1. User change permissions and filter options for the server. 2. Changes are saved to server. 3. Sender enters a message to be sent to another user. 4. Message prior to encryption is filtered for language. 5. Message is sent to server. 6. Server decides if the user is spamming by message count. 7. Message ends at the receiver.	

Sequence Diagram for Sprint #3



Software Team Structure[13]

Control Centralized

There is a strict hierarchy with a defined team leader for the extent of the project who communicates requirements to the rest of the team. The project is well defined and simple with little to no problem solving required which is handled exclusively by the team leader.

Controlled Decentralized

This structure also has a defined leader but also has appointed secondary leaders for different sub tasks. Communication travels both horizontally within sub groups and vertically along the hierarchy. Problem solving is handled collaboratively and the project is usually simple with some moderate amounts of brainstorming needed.

Democratic Decentralized

Provides a very flexible approach to the structure of the team with emphasises on creativity and problem solving. There is no fixed leader with each team member taking leadership during the project. Communication occurs horizontally across the team which aids in solving complex issues. Low modularity and high team life are pivotal for the democratic decentralized structure.

Closed Paradigm

This paradigm is closely related to the central centralized team structure. It works better when the team has previous experience working on a past problem or project that is similar to the one they are currently trying to implement. The closed paradigm structure has a rigid hierarchy and cannot be innovative which means no new ideas or processes are attempted.

Random Paradigm

The strengths of this paradigm lie in its ability to innovate or when a technological advance must be made. The structure is very loosely structured which makes the project dependent on the

[13] B, S R. “Software Engineering Week #9

individual contributions of its members. This model is not efficient and may cause issues for projects that are routine in nature and need a more controlled structure. **Open Paradigm**[14]

Combines some of the benefits of both closed and random paradigms by retaining a strict team hierarchy but leaving room for innovation. Problems found using this model are solved collaboratively, communication is abundant between team members meaning that decisions are

usually resolved by consensus. The flexibility of this paradigm comes at a cost however leading to efficiency loss when compared to other paradigms.

Synchronous Paradigm

The synchronous paradigm divides a complex problem into smaller and easier sub tasks that are assigned to different team members who usually complete the task with little or no communication. The problem itself must be able to be broken into smaller tasks for this paradigm to be usable.

Team Structure

We chose to structure our team using the democratic decentralized structure because of the complicated nature of our project. The team has limited experience implementing the proposed features and working with tools such as Firebase and React. Our team is small and will remain together for the duration of the project, which lends itself naturally to the chosen structure. **Phase #1 - Scrum Master: Responsibilities and Scrum Team: Responsibilities**

Scrum Master Chris Miele	Defining Economic and Schedule Feasibility Found tools we can use for the project Found Discord upper management and execs(Linkedin) that we interviewed.

[14] B, S R. “Software Engineering Week #9

Scrum Team #1 Moiz Adil	Technical Feasibility for the feasibility study, researching possible tools to use in regards to software, hardware. Aided in the compilation of the feasibility report and in turn the scope of the project. Provided definitions for various requirement elicitation techniques while creating hybrid questionnaires.
Scrum Team #2 Steven Michalec	Responsible for the operational feasibility for the feasibility study. Provided the management summary and functionality sections for the feasibility report. Created and formatted the tables included in the report and scope. Completed the JAD description and conducted and documented the interview.

Phase #2 - Scrum Master: Responsibilities and Scrum Team: Responsibilities

Scrum Master Steven Michalec	Created the UP and Spiral process definitions. Defined the Scrum agile framework and the stages included. Aided in the creation of the product backlog and sprint log. Helped to create use case diagrams and sequence diagrams.
Scrum Team #1 Chris Miele	Defining the WaterFall and Prototyping Created System Context Diagram Created Entity Relationship Diagram
Scrum Team #2 Moiz Adil	Aided in the organisation and compilation of product backlog and in turn sprint backlog. Created Data Flow Diagrams for each sprint alongside use case diagrams.

Phase #3 - Scrum Master: Responsibilities and Scrum Team: Responsibilities

Scrum Master Moiz Adil	Researched and understood the software-team structure that would work best for our team. Ultimately resulting in the choice of democratic decentralized. Aided in answering W5HH principle questions. Researched and understood documentation for React framework. Researched and implemented firebase for authentication.
Scrum Team #1 Steven Michalec	Defined the varying team structures available and assisted in answering the W5HH principle questions. I also helped to implement multi factor authentication required for the first sprint goal.
Scrum Team #2 Chris Miele	Finding documentation on how to create UI in React Documentation on Firebase Multifactor Authentication API. Explained why we chose DD. Assisted with W5HH principles.

W5HH Principle[15]

Why is the system being developed?

The system is being developed because there are clear disadvantages to how Discord currently implements security into their application. These limitations can be found within a few key parts of the application, the problems we wish to address are:

- Non-mandatory two factor authentication of users
- Permission scope for users that are granted by admins
- Lacklustre spam filtering
- Encryption only for data in transit

[15] B, S R. “Software Engineering Week #9

The goal is to remedy these issues to improve the security and safety of users without impacting the performance of the software.

What will be done?

The solution to the issues previously stated is to add and augment the security measures within the application to address those concerns. The first proposed change is to make multi-factor authentication mandatory. The addition of MFA improves the integrity of the authentication process tremendously. The next addition is to implement end-to-end encryption, which protects user data during transfer and rest periods. Finally, modifying the permission controls will give more power to administrators, enhancing security measures on servers.

When will it be done?

The timeline for the implementation of our solution is 13 weeks. The project will be delivered in multiple iterations until a cohesive environment is reached for users.

Who is responsible for each function?

Steven

Responsible for the registration and logging in process which includes successfully sending authentication credentials to firebase and receiving the user credentials. Included in this process is the routing and navigation of pages which guides the user through the application.

Chris

Html and css for voip-app, Explained why we choose DD, Assisted with W5HH Principles, Found documentation about UI and Assisted with implementing Multi Factor Authentication.

Moiz

Researched firebase for db and validation. Researched implementation of sms/phone provider validation. Ultimately decided against this form of auth.

Where are they organizationally located?

The stakeholders in the company are C Level Executives, Management, Employees, and Customers(End-Users). The C Level Executives are affected because they task our team with adding upgrades to discord they are paying 170k for our service. The features we are implementing

are Multi Factor Authentication, End to End Encryption, Spam Filtering and Permission Control. Management is also being affected by the implementation because they are going to be the ones that are administering the features and overseeing potential addons to the feature that we have provided. The Employees are going to be affected because they are the ones that are going to be fixing the code and adding features. The Customers(End-Users) are affected because they are the ones that are going to be affected by the new features that are going to be implemented Multi Factor Authentication when they are logging into their account, End to End Encryption where there messages are encrypted on the server, spam filtering so you can't send lots of messages Allowing End-users to control their permissions on servers.

How will the job be done technically and managerially?

The team has decided to apply the Scrum framework to the project. Scrum is a process that uses agile methodology to focus on providing an incremental product to the customer. Each sprint will incorporate a section of the required systems to be implemented in the Discord application. We have decided to structure the team using democratic decentralized to assist us in solving any complex issues we may encounter. The team is planning to use Firebase for the back end of our application and React for the front end.

How much of each resource is needed?

Firebase as a backend to store user information, authentication, and host the website. Visual Studio Code is used to code the website. React (Frontend) creating a ui for our voip app.

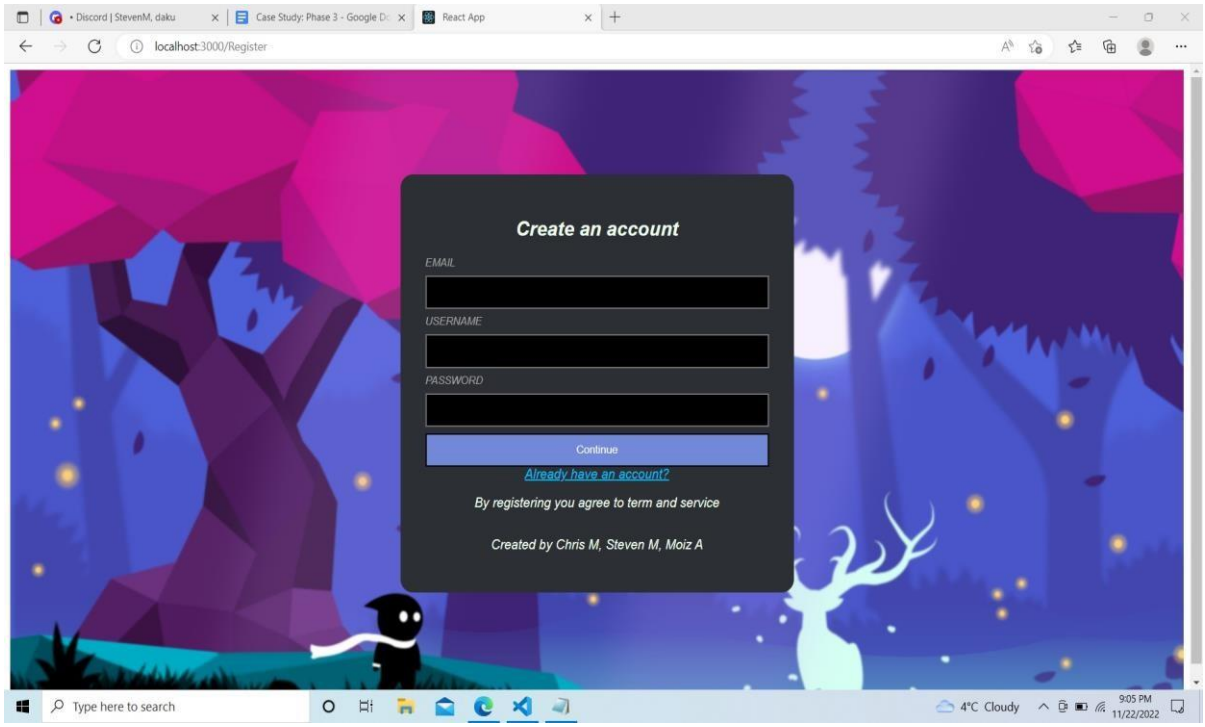
System Implementation

For sprint 1

Which tool was selected for the project? Explain.

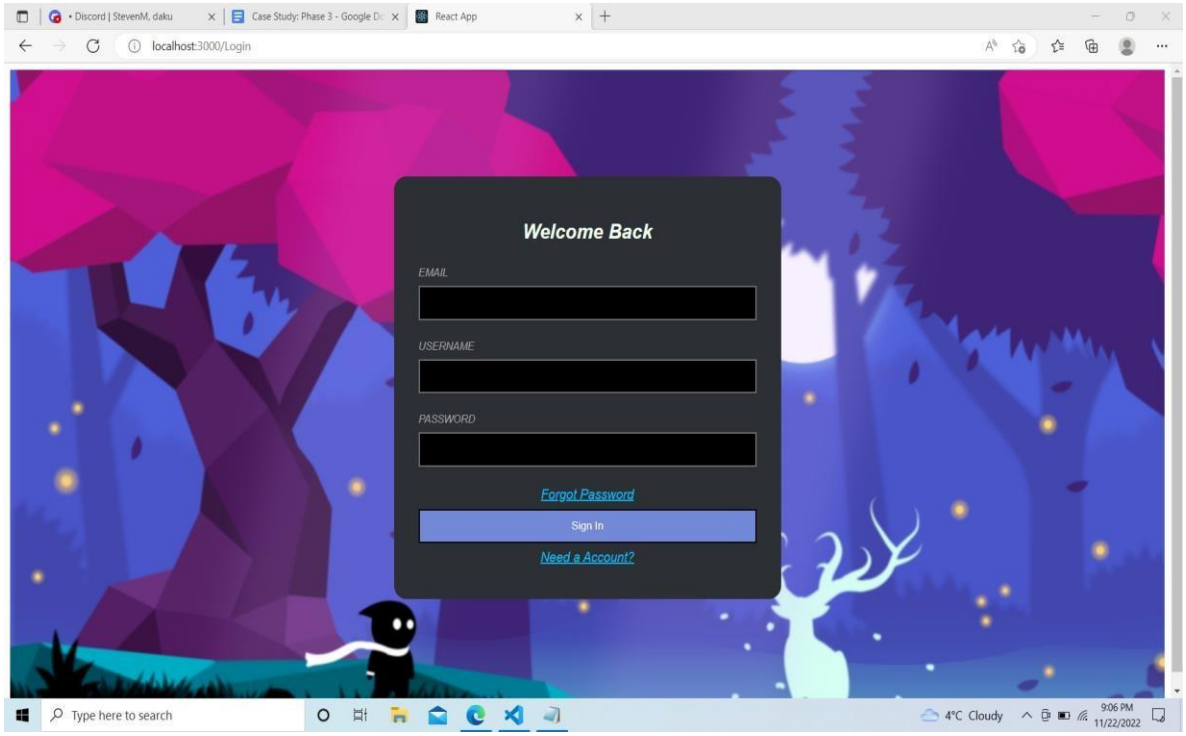
We used React, a JavaScript library to provide a UI; in tandem with firebase for backend validation and authentication. This allowed us to authenticate new users account creation and provide a way for current users to validate login. We chose the React library because it allows for the creation of simple dynamic components in turn producing a fully fledged UI. Firebase is a “Backend as a Service” that provides a real time database. It also provides an authentication scheme that is easily integrated with our web app.

User Interface Registration[16]

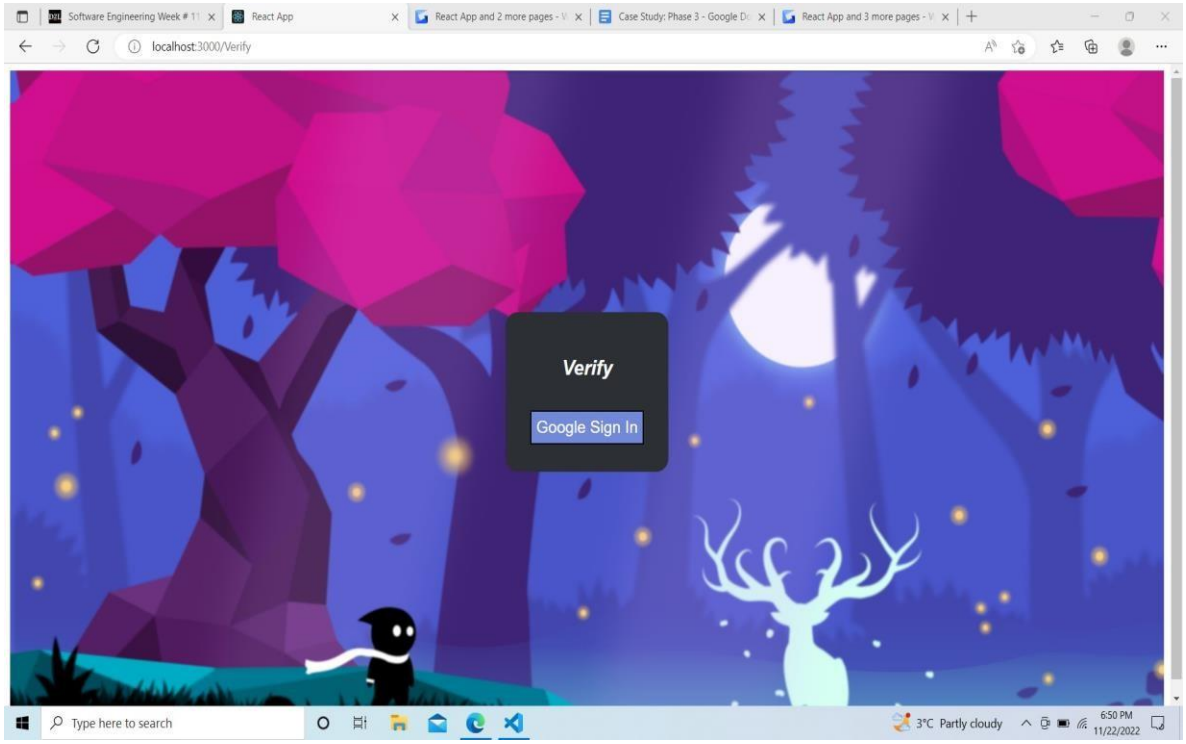


User Interface Login

[16] Saksham-kumar-14 Discord-clone colorswall
Discord colors palett

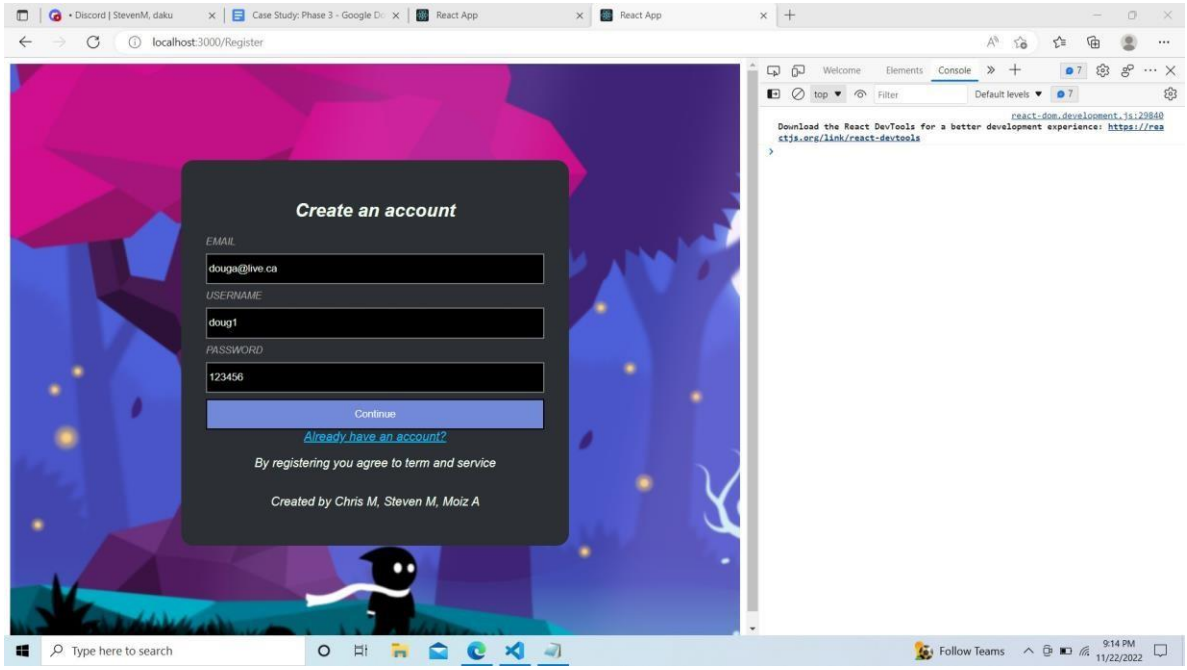


User Interface Verification

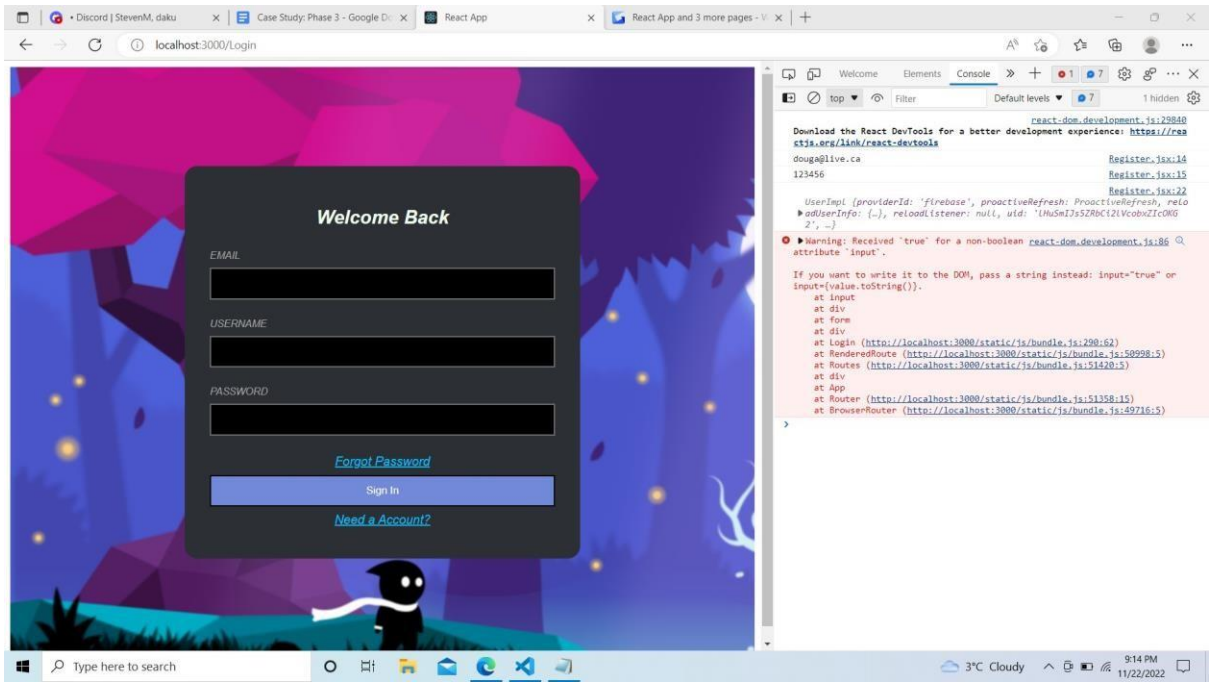


Functionality

User doug1 is registering an account.

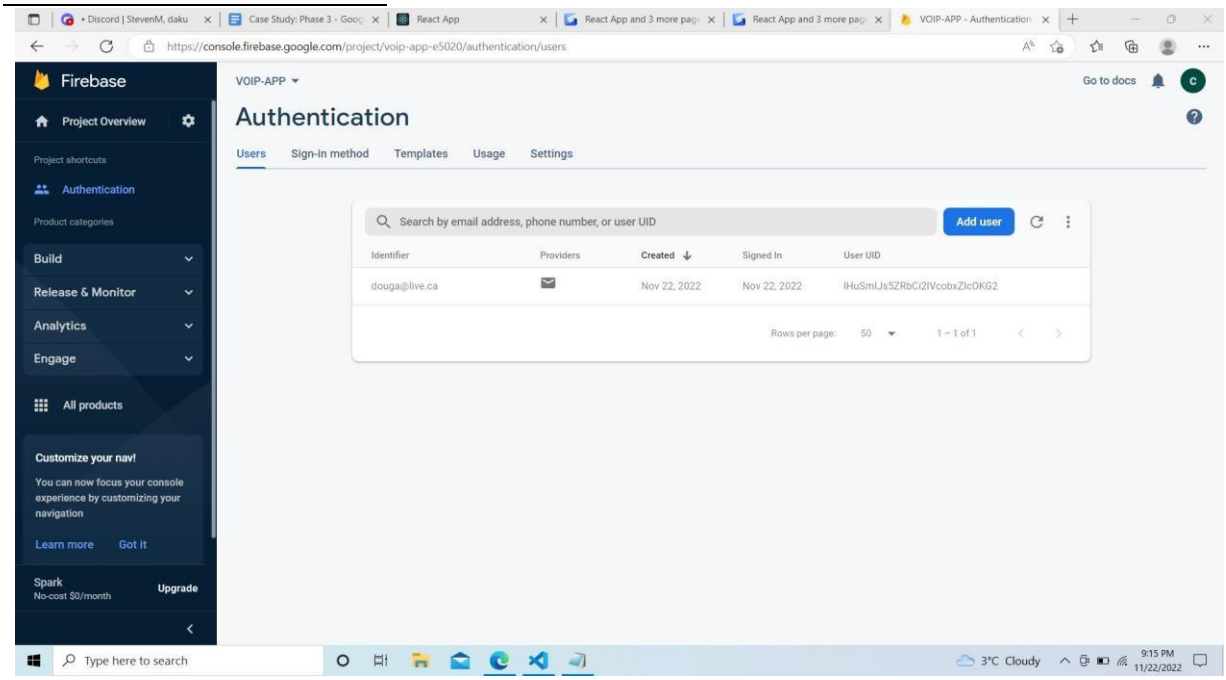


Once doug1 is registered on our site they are sent to the login page when they click continue. The developer tools show that the email and password authentication is working.

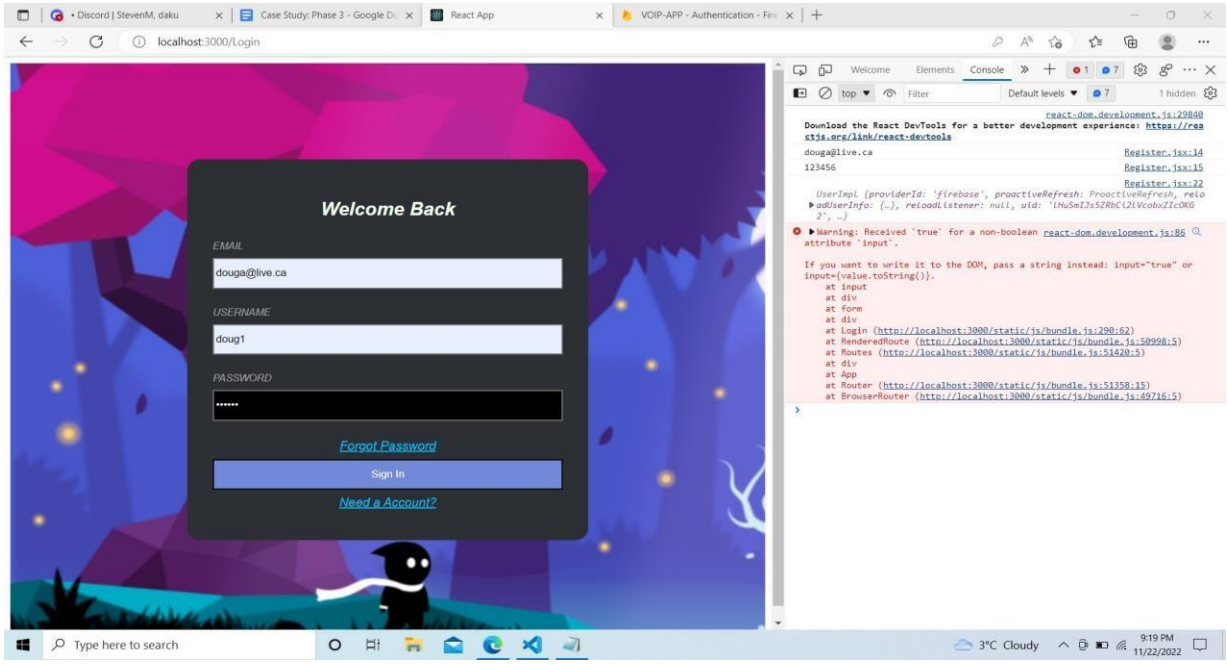


In our firebase which is linked to our voip-app website we can see that the user douga@live.ca account was registered.[17]

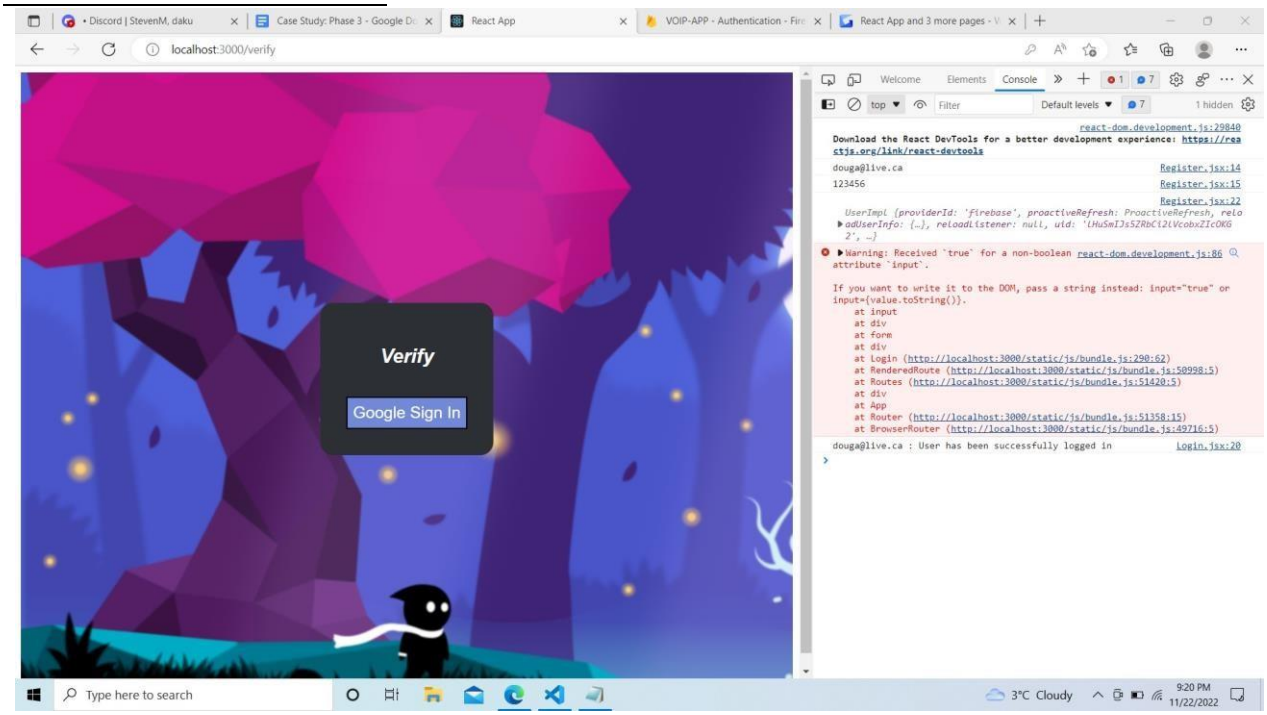
[17] Firebase
Firebase Password Authentication



User doug1 is entering his credentials.

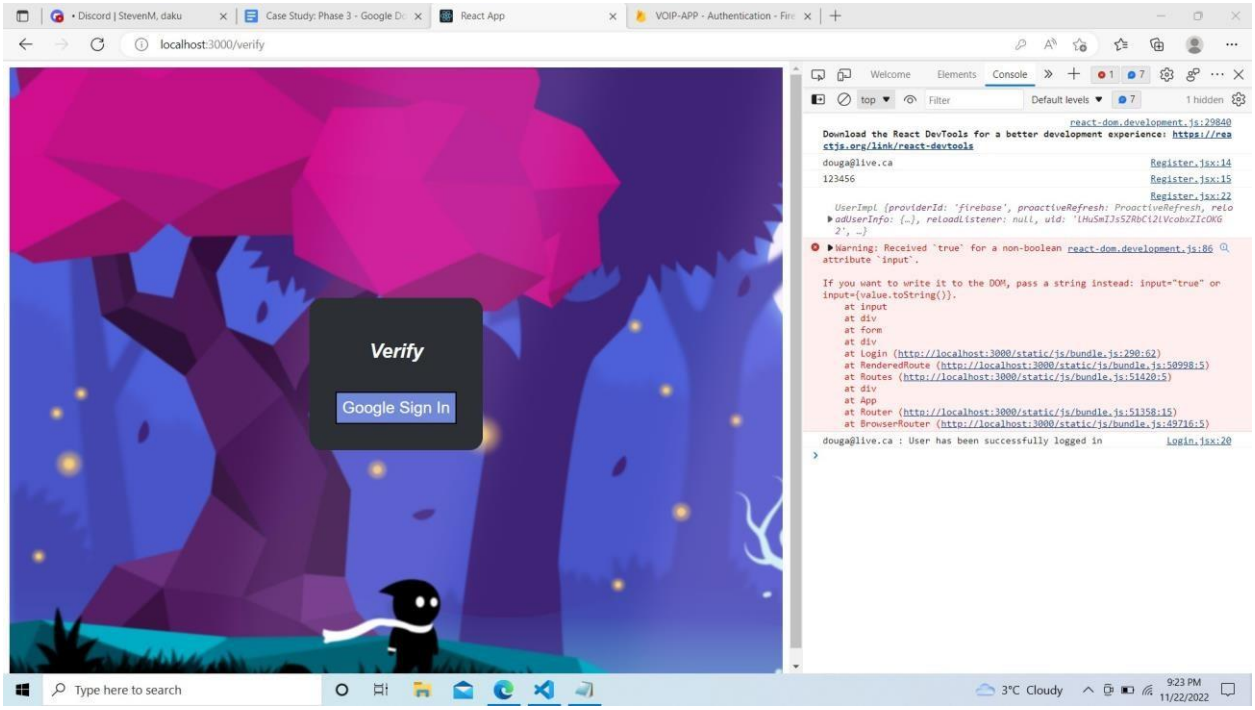


Confirming douga@live.ca is successfully logged in as seen in developer tools.[18]

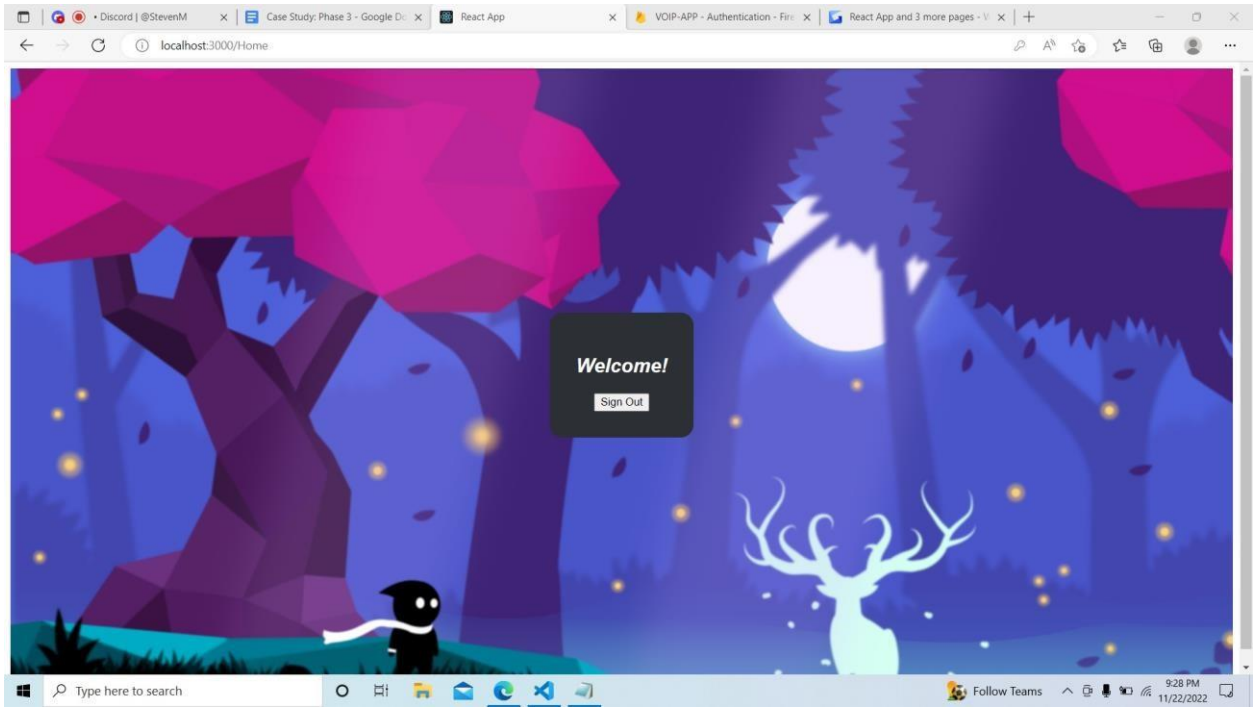


Based on our sprint 1(Multi Factor Authentication) douga must now enter his Google sign in credentials.

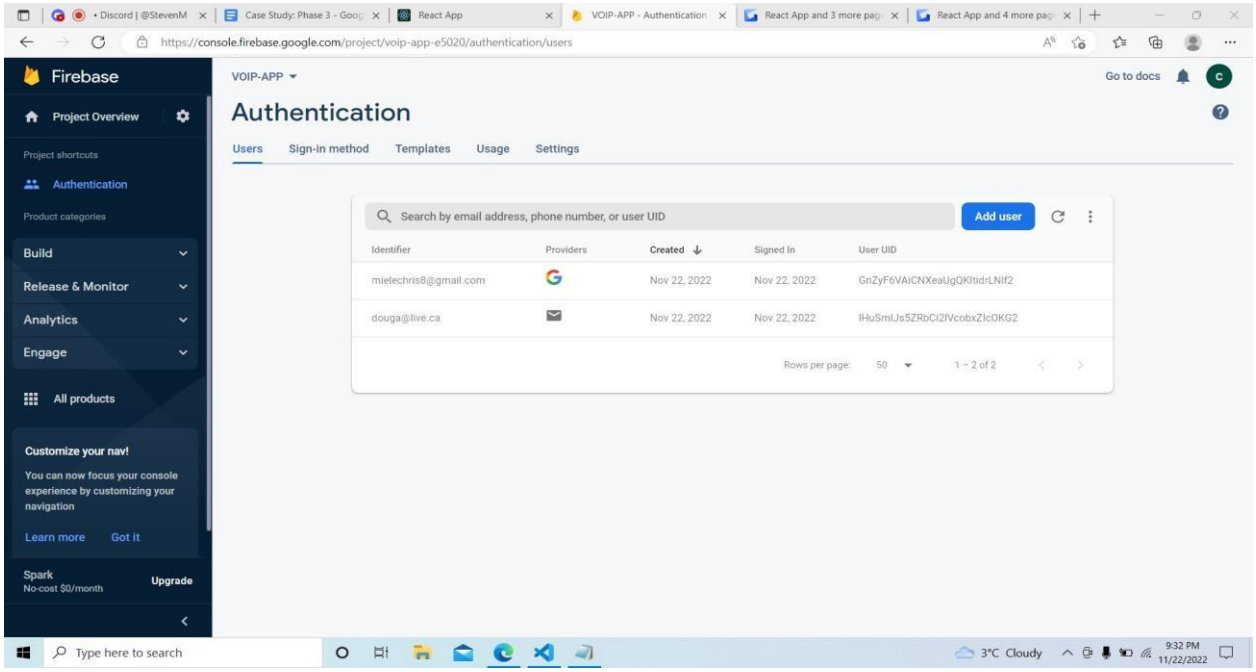
[18] Firebase Google Authentication



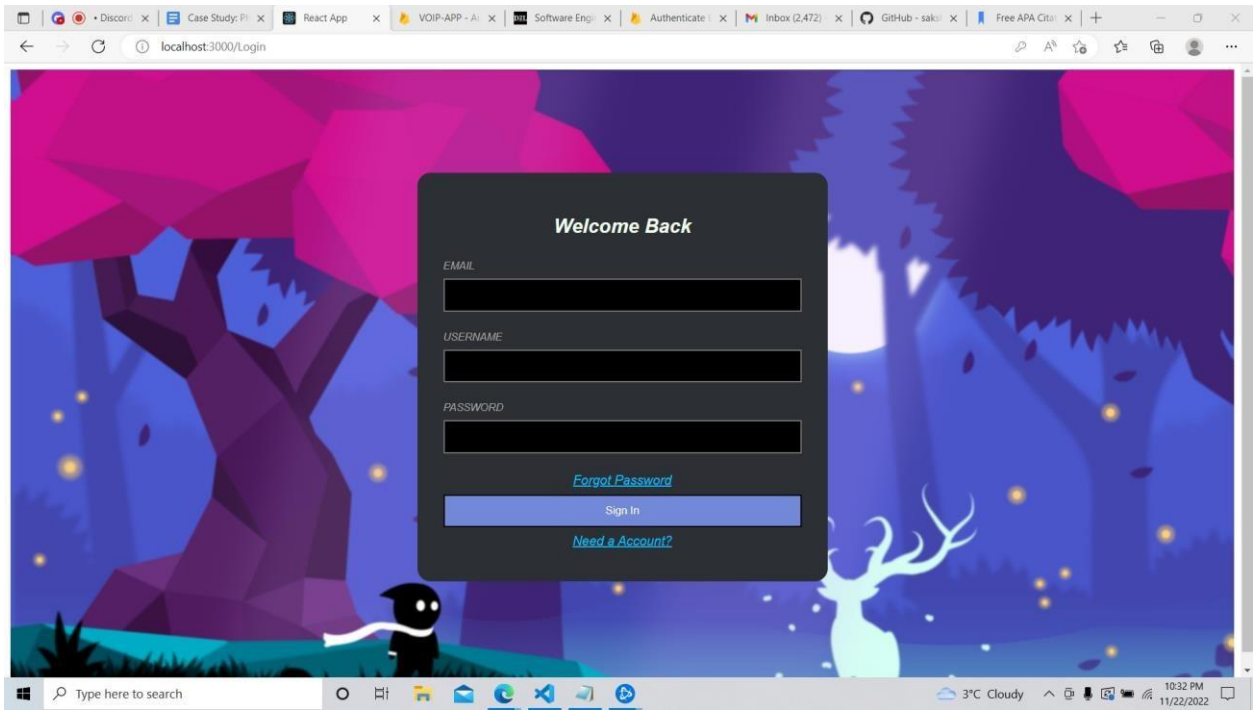
douga entered login credentials and now he has the option to sign out.



Firebase shows douga is signed into his Gmail.



Once douga clicks sign out he is sent to the login page.



Software Testing

Software Testing

Is the key activity in developing and maintaining a system.The objectives are not that of development rather the **objective** aims at breaking the system and revealing defects within it.[19]

Sr #	Type of Testing	Description
1.	Unit	Unit tests are typically used to test each module of a system in isolation. Tests are written by developers to test the functionality of the system as it is created.

[19] B, S R. "Software Engineering Week #11" Oct. 2022

2.	Integration	Integration Testing checks components are tested in isolation. Make sure the components work properly before being put together.
3.	System	System Testing ensures the whole System works properly.
4.	Regression	The selective re-tests of the system or component that verifies that the modifications have not caused unintended ripple effects.
5.	Coverage	Coverage Testing degree which a software program has been tested.
6.	Mutation	Mutation Test a copy of the original file with some changes and compare results with original.

7.	Validation	Determine if the software meets all the requirements defined in the SRS.
8.	Alpha and Beta	Alpha test you are testing with a small group of people. Beta testing rolling out to a larger group of people.
9.	Acceptance	Testing technique performed to analyze whether the system has met customers required specifications. This test typically tests the entire system or a large portion of it.

Test Planning

Scrum Sprints	No. of Test Cases	Functionality Features
1	14	3

2	10	3
3	10	2

Test Procedure

Test Case No.	**Test Description/Data**	**Expected Result**

2	Testing the ability to sign up when you are creating an account in our voip app. Test Data Email, Username, Password	The Email, Username, and password should show up in firebase. Also, you should be able to sign in with the new credentials on the voip-app.
16	Testing to see if users can message one another in our voip-app. Test Data is Strings of characters sent from one user to another.	The Users that are signed into the voip-app should be able to see the messages between one another.
26	Testing to see if when you type a swear word it is blocked on the user side. Data would be Strings of characters. Test to make sure you can create permissions in the voip-app Data would be Permissions.	The Expected result for the spam filter is blocking a swear word on your end and also blocking the swear word when the other user receives the message. The Expected result is the ability to join a server and allocate permissions for a user on said server.

Test Report

Date	**Test Case No.**	**Test Result**	**Pass/Fail**	**Tester Name**	**Test Sign**
2022/11/28	2	Expected	Pass	Chris	

2022/11/29	16	Expected	Pass	M	
2022/11/29	26	Swear words not blocked	Fail	S	

Software Deployment

Definition

The process in which new software or update is made available from the developers to the users that will be using the software. The importance of software deployment is to improve the efficiency and speed of delivering a quality product. There are four main strategies used in the deployment process.[20]

Direct Cutover Approach

This method requires stopping the old system and starting the new system at the same time, which comes with risks due to unforeseen issues and possible data availability problems. The benefit, however, is the speed at which the new system can be implemented.

Parallel Approach

The old system, and the newly developed system, are run concurrently with each other, which protects data availability. The issue however, is that running two sets of the same system can become very costly.

Phase-in Approach

This approach breaks down changes into components that are slowly phased into the current system. It is an iterative process that reduces implementation risk but significantly increases the time taken for all changes to be added.

Pilot Approach

For changes that need to occur to a large group of users, sometimes the best approach is to test the changes on a small subset of users to iron out possible issues during implementation. If the process is successful, then another approach can be used to roll out the new system. This process will increase the time it takes for deployment but decreases potential risks. **Project Deployment Plan**

The deployment plan we intend to use for this project is the phase-in approach. This process ensures that the data currently saved in the old system is safe and that each module can be successfully implemented into the discord application. The cost overhead for the parallel approach

[20] B, S R. “Software Engineering Week #12” Oct. 2022

would be too expensive, and the direct cutover approach could lead to data and program availability issues. The phase-in method is a happy medium, ensuring a safe transition to the new security components.

Software Maintenance

Definition

Maintenance is the process of either correcting bugs or updating the software to improve performance or other valuable traits. The procedure of maintaining the software happens after deployment and is needed because software is never perfect. Adapting to changes in the operating environment also requires changes to occur in the software. There are four types of maintenance that can be used to address faults in software.[21]

Corrective Maintenance

As the name implies, corrective maintenance helps to address and fix any defects in the software or product that is being maintained. These bugs can be found during an internal review process or by customers.

Adaptive Maintenance

Adaptive maintenance hopes to address any problems with the software in terms of the operating environment. There are no new functionalities being implemented into the program through this type of maintenance. Its focus is to ensure that all current functionality is working.

Perfective Maintenance

The intended effect of perfective maintenance is to make meaningful improvements to the current system without having those changes impact the functionality of your customers. Trying to make the program more extensible and future features more straightforward in implementation is a subset of perfective maintenance.

Preventive Maintenance

Preventive maintenance helps to alleviate future problems by preventing them from occurring and is a type of perfective maintenance. Regular preventive maintenance hopes to keep the software in working condition.

[21] B, S R. “Software Engineering Week #12” Oct. 2022

References

Duggan, E. W., & Thachenkary, C. S. (2004). Integrating nominal group technique and joint application development for improved systems requirements determination. *Information & Management*, *41*(4), 399-411. Retrieved 9 25, 2022, from https://doi.org/10.1016/S0378-7206(03)00080-6

Agarwal, B. B., Tayal, S. P., & Gupta, M. (2010). Software engineering and testing. Jones & Bartlett Learning pg 31. Software Engineering and Testing - B. B. Agarwal, S. P. Tayal, Mahesh Gupta - Google Books

B, S R. "Software Engineering Week 2-2." 22 Sept. 2022.

B, S R. "Software Engineering Week #3." 22 Sept. 2022.

B, S R. "Software Engineering Week #4-2 4 Oct. 2022.

B, S R. "Software Engineering Week #5 4 Oct. 2022.

B, S R. "Software Engineering Week #9" Oct. 2022

B, S R. "Software Engineering Week #11" Oct. 2022 B,

S R. "Software Engineering Week #12" Oct. 2022

Sakshama-kumar-14, Discord clone made using react js.
GitHub - saksham-kumar-14/discord-clone: Discord clone made using react js

Discord colors palette, ColorsWall. (2018, May 25). Colorswall.com https://colorswall.com/palette/181/

Firebase
Firebase (google.com),

Firebase Password Authentication
Authenticate with Firebase using Password-Based Accounts using Javascript (google.com)

Firebase Google Authentication
Authenticate Using Google with JavaScript | Firebase