Kyle Connect

Functional Specification

Course: Bachelor of Science (Honours) in Software Development

Name: Kyle Kinsella

Student Number: C00273146

Date: 25/10/24

Supervisor: Joseph Kehoe

Table of Contents

What is my project?

Introduction

The aim of my project is to develop a communication platform inspired by an app called Discord, a popular chat and collaboration tool. My project will be created to be an accessible and user-friendly environment where individuals or groups can engage in seamless communication.

Learning outcome

I chose to develop a project similar to Discord for several reasons. Primarily, I aim to acquire more knowledge in various types of programming languages and frameworks, such as JavaScript, Node.js and Go. These types of technologies are required for developing an application such as Discord. In addition to that I will improve my technical documents, such as a functional specification, research poster, presentations and much more. This project will enhance my technical skills and my ability to write effective technical documents, equipping me with valuable expertise for future software development projects after I graduate college.

Technical Issues

Target platform

The target platform for my users to use my product is going to be a website. I chose to pick a website because it is very accessible for anyone to use.

Security

As this project is primarily to be able to communicate with people or friends, it is very important that we encrypt any form of communication between one user and the other. In order for my website to be as secure as possible I am going to include some of the below encryption algorithms.

What is encryption?

Encryption is the protection of data by using very complex algorithms to jumble up the data in such a way that nobody can get access to the data. The only way to get access to the data is a key. This key gives you access to the data.

What is decryption?

Decryption is the process of taking the encrypted data and transforming it back to its original form.

Am I going to be storing messages?

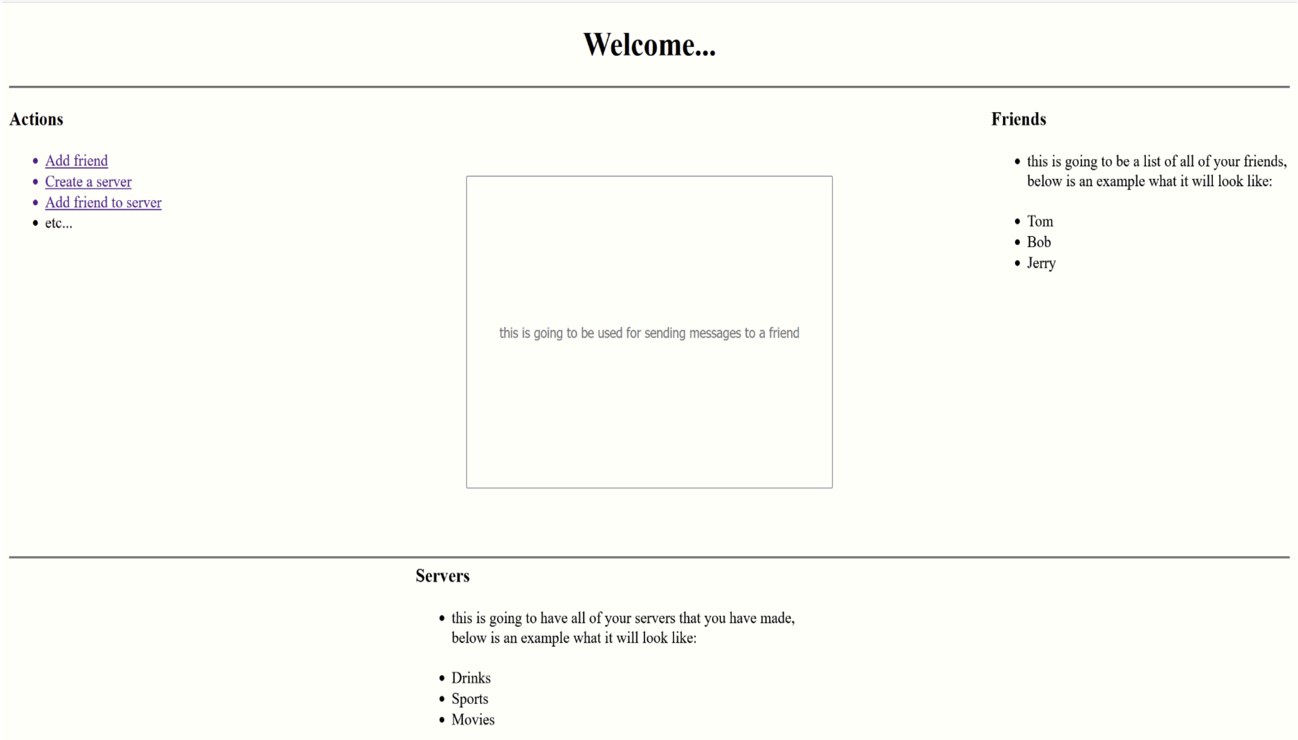
Yes, I will be storing messages because, if you sign out of your account and then sign back in, I will need to retrieve the necessary messages from your friends and other information if needed. I will be storing the messages in a database in the cloud, I will not be storing the messages locally on your machine, this is due to a potential security issue.

User interface

When a user is going to be using my product, I want my user interface and experience to be of very good quality. In order for this to succeed I will design my website in such a way that it is easy to navigate and use.

Below is a brief explanation on how my system is overall going to work.

At the start of the website the user will firstly have to create an account to be able to communicate with other people, once their account is created they will be able to login to their account and then start to use the website. Once the user has logged in to their account they will see something like the below image.



Welcome...

Actions

- Add friend
- Create a server
- Add friend to server
- etc...

this is going to be used for sending messages to a friend

Friends

- this is going to be a list of all of your friends, below is an example what it will look like:

- Tom
- Bob
- Jerry

Servers

- this is going to have all of your servers that you have made, below is an example what it will look like:

- Drinks
- Sports
- Movies

The above image is the home page of the website. When the user wants to add a friend they will navigate to the Actions tab highlighted in the colour orange and then click on add friend, then the user will enter a friends name to add, then the database will retrieve the names depending on the user input and then the user will be able to add whatever friend they wish. Once the user has added their friend, that friend will be added to the Friends tab highlighted in the colour green. Under the friends tab will contain all of your friends that you have added. When the user wants to start a conversation with their friend they will select a friend from their friends list and then the middle part of the website “central part of the application / web-app” will populate the selected friend and it will initiate a conversation between one user and the selected friend.

When a user wants to make a server, they will navigate to the Actions tab highlighted in the colour orange and click on “make a server”.

What is a server?

A server is a dedicated space for your friends to connect, co-create, engage and much more. On my website you will be able to make your own server and add your friends to the server that you have just made.

Then a page will come up asking you to enter the name of the server, then you will click a button to make your server. Then you can add sections within your server. For example, we named our server “Week days”, within our server called Week days, we could have a section to talk about stuff we do on a monday, tuesday, wednesday, thursday, friday, saturday and sunday. Once you have made your server it will be put into the bottom section of the image above, under the section called servers.

If a user already has a server made but they don't have anyone else in the server, they might want to add one or many friends to their server. In order to do this they will navigate to the Actions section in the user-interface and click on “add friend to a server”.

Then a page will come up with all of your friends that you have, and all of the servers you have and you will be able to select a friend to add to a certain server.

For example, if I have a friend called Jerry and I want to add him to my server called Week days, you might see something like the following: Add Jerry to Week days -> add button.

Target users

My project is to develop a communication platform for any type of user to be able to communicate with people around the world and their friends. My website is made for anyone that wants to be able to communicate with a friend or anyone in the world.

Format

The format that you will be using when communicating with anyone on my website will be the use of text, emojis and numbers. You won’t be able to send an image or gif, you can only send texts, emojis and numbers to your friend or friends.

Similar applications

During my research before starting to develop Kyle Connect I have found some similar applications such as:

Discord

Discord is an application that is used for communicating with friends either one to one or via a server with one or many people. Discord is different to Slack and Flock because it is an informal application, this means that people use it for games, talking with friends and much more. Below is an image of the Discord user-interface.

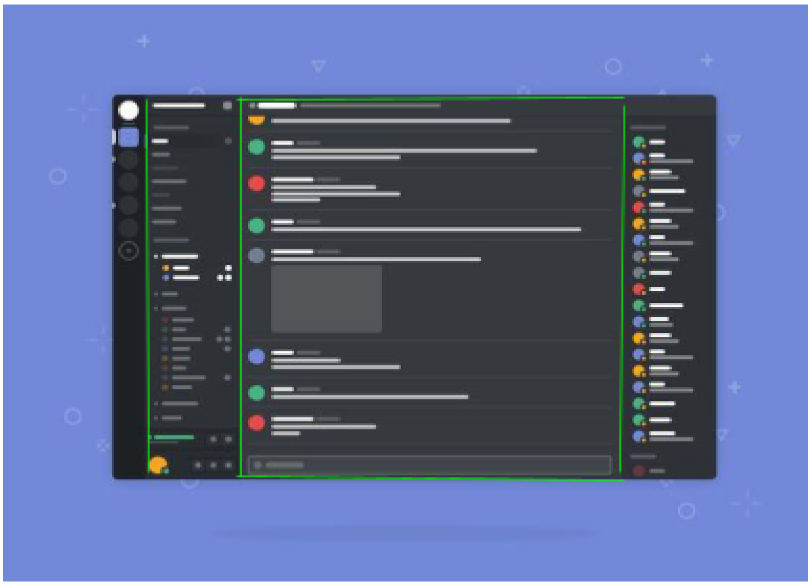


Figure 1, see reference [5].

Explanation of Discord

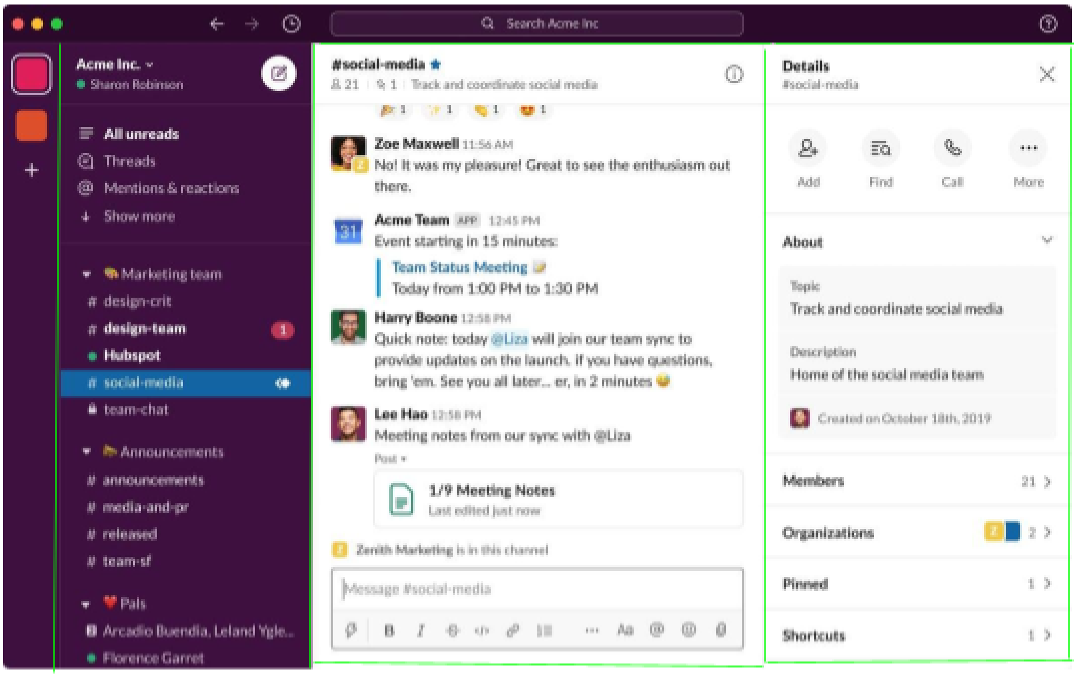
In the above image of Discord, on the left hand side of the user-interface there are your servers that you are in on the far left hand side. The middle part of the image is the main part of the application, this part of the application is where your messages get to so the users of the system can view the messages.

How to Discord

In order for the user of the system to use the above image you will need to find the white dot on the top left hand side of the user-interface under that dot will be all of your servers that you are in. To get into a server click on the server you want to go into and the system will bring the user into that server to view messages within that server, these messages will be in the middle part of the screen.

Slack

Slack is similar to Discord, in that you can create a server to be able to communicate with your business colleagues. Slack is a more formal communication application compared to Discord being a more informal application. Below is an image of the user-interface for Slack.



Zoe Maxwell 11:56 AM
No! It was my pleasure! Great to see the enthusiasm out there.

Acme Team APP 12:45 PM
Event starting in 15 minutes:
Team Status Meeting
Today from 1:00 PM to 1:30 PM

Harry Boone 12:58 PM
Quick note: today @Liza will join our team sync to provide updates on the launch. If you have questions, bring 'em. See you all later... er, in 2 minutes

Lee Hao 12:58 PM
Meeting notes from our sync with @Liza
Post
1/9 Meeting Notes
Last edited just now

Zenith Marketing is in this channel

Topic
Track and coordinate social media

Description
Home of the social media team

Created on October 18th, 2019

Figure 2, see reference [4].

Explanation of Slack

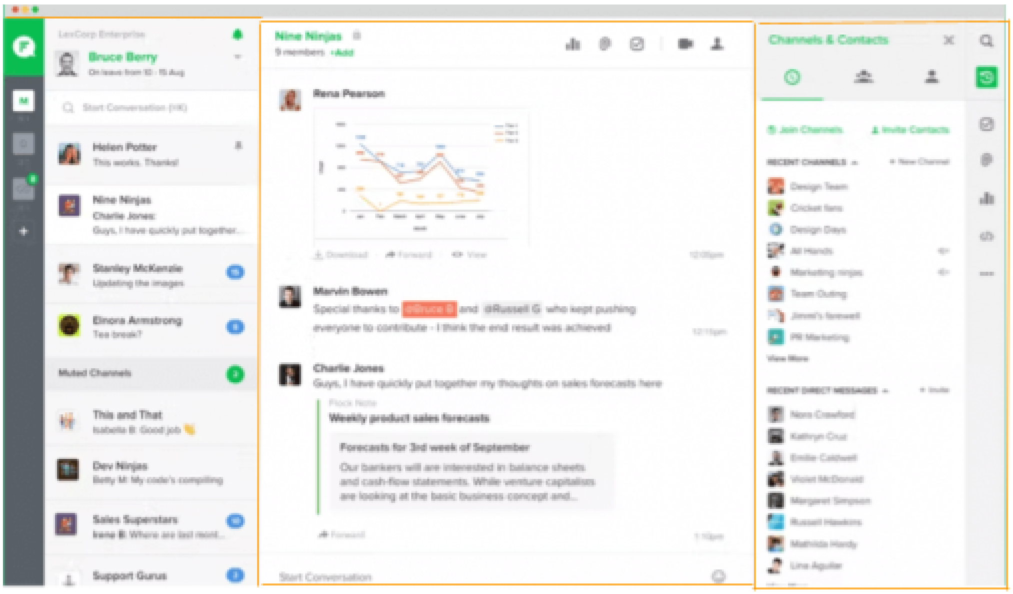
In the above image of Slack, on the left hand side of the user-interface there are your unread messages or servers, threads and some others. The middle part of the image is the main part of the application, this part of the application is where your messages get to so the users of the system can view the messages. On the right part of the user-interface you are able to add a person if you clicked on that person's profile and much more.

How to use Slack

In order for the user of the system to use the above image you will need to find a server that you want to enter on the top left hand side of the user-interface with the colour red or orange here you will have all of your servers that you are in. To get into a server click on the server you want to go into and the system will bring the user into that server to view messages within that server, these messages will be in the middle part of the screen.

Flock

Flock is a very similar application to Slack mainly due to the fact that the software is used for businesses meaning the software is a more formal usage of communication. Below is an image of the user-interface for Flock.



LexCorp Enterprise
Bruce Berry
On leave from 10 - 15 Aug

Start Conversation (⌘K)

Helen Potter
This works. Thanks!

Nine Ninjas
Charlie Jones:
Guys, I have quickly put together...

Stanley McKenzie
Updating the images

Elnora Armstrong
Tea break?

Muted Channels

This and That
Isabella B: Good job 👏

Dev Ninjas
Betty M: My code's compiling

Sales Superstars
Irene B: Where are last mont...

Support Gurus

Nine Ninjas
9 members +Add

Rena Pearson

Download · Forward · View 12:05pm

Marvin Bowen
Special thanks to @Bruce B and @Russell G who kept pushing everyone to contribute - I think the end result was achieved 12:11pm

Charlie Jones
Guys, I have quickly put together my thoughts on sales forecasts here
Flock Note
Weekly product sales forecasts
Forecasts for 3rd week of September
Our bankers will are interested in balance sheets and cash-flow statements. While venture capitalists are looking at the basic business concept and...

Forward 1:10pm

Start Conversation

Channels & Contacts

Join Channels · Invite Contacts

RECENT CHANNELS · + New Channel
Design Team
Cricket fans
Design Days
All Hands
Marketing ninjas
Team Outing
Jimmi's farewell
PR Marketing
View More

RECENT DIRECT MESSAGES · + Invite
Nora Crawford
Kathryn Cruz
Emilie Caldwell
Violet McDonald
Margaret Simpson
Russell Hawkins
Mathilda Hardy
Lina Aguilar

Figure 3, see reference [4].

Explanation of Flock

In the above image of Flock, on the left hand side of the user-interface there are your channels that you are in. The middle part of the image is the main part of the application, this part of the application is where your messages get to so the users of the system can view the messages. On the right part of the user-interface you are able to join other channels, invite contacts and much more.

How to use Flock

In order for the user of the system to use the above image you will need to find a channel that you want to enter on the left hand side of the user-interface that will show you all of the channels you are in. To get into a channel click on the channel you want to go into and the system will bring the user into that channel to view messages within that channel, these messages will be in the middle part of the screen.

Comparing Applications

When I am comparing Slack and Flock to Disord, I mention that Slack and Flock are a more formal way to communicate, and Discord is a more informal way to communicate. Discord is an application for friends to communicate in more than one way such as, talking about games, movies and much more, this is an example of an informal application. As where Slack and Flock are only used for one use and that is for business. To explain what I mean by formal and informal I will define these words.

Formal - To do something in a formal way means to follow a convention or etiquette. For example, When we are writing an email to a colleague we would have the email be structured and formal signatures to sign off the email.

Informal - To do something in an informal way means to not follow a convention. For example, if you were sending an email you wouldn't use structure and you might use abbreviations.

Operation of my product

Examples of a person or a group of people using my system

To display some examples on how someone will use my product. Below are a few examples.

1. A group of friends like to connect every Friday at 2 O'Clock in the evening to play a game of chess, they will be able to make a server with all of their friends in that server, once the time hits 2 O'Clock each friend can send a message into the server to let the other friends know that they are ready and waiting to play there game of chess.

2. A friend sends you a message asking you what you think the weather might be like tomorrow, then you send your message back with what you think the weather will be like tomorrow. Your friend mentions that if you guess the weather to be correct tomorrow he will give you €10.

3. In your local area there are a few people that like to keep the area nice and tidy. Someone will create a server and within this server they will be able to send a message to the server and specify what person is tidying on what day of the week. If someone is not able to be there day of the week they will be able to send a message to the server and ask someone else can they swap that day with another person in the server.

Explanation of my user-interface

The below image is what my user-interface might look like on my website. I am going to explain how the user-interface will work. Below are four sections named, Actions, Friends, Servers and a box in the middle of the screen.

Actions: The actions section is going to contain a collection of tasks for the communicator to do during their time using my product. For example, the person using the system will be able to make a server, add friends and much more.

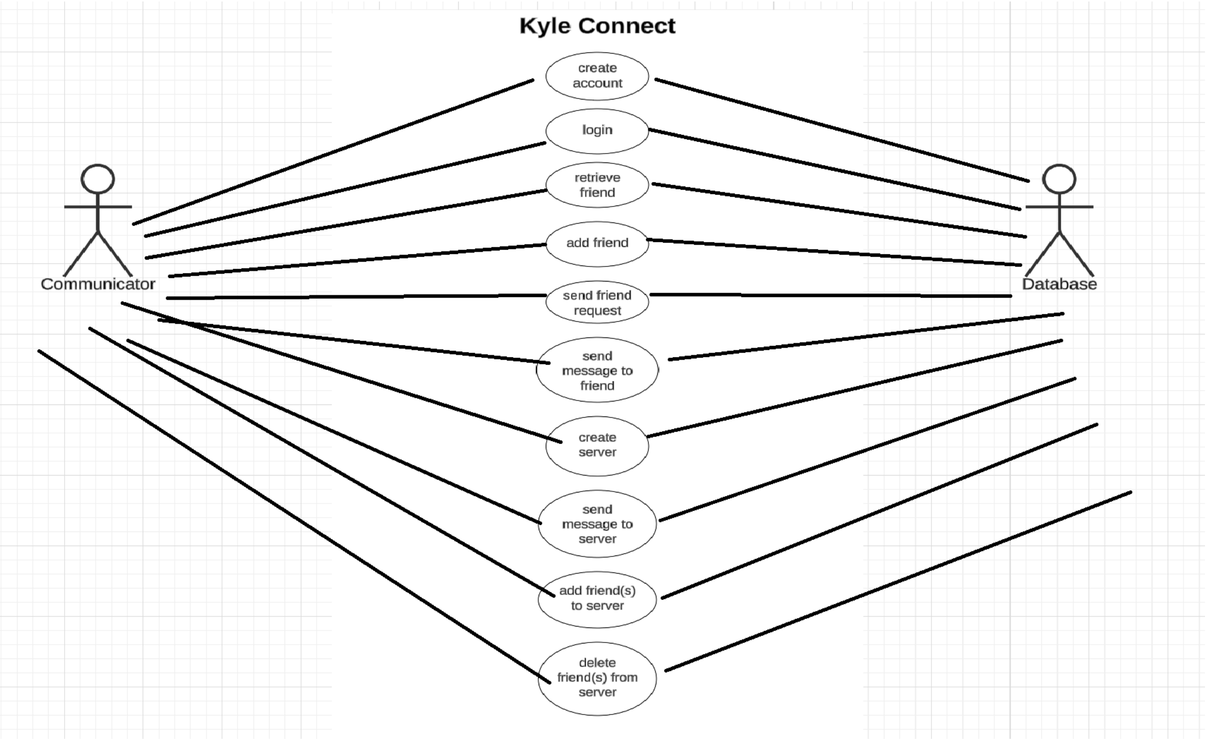
Friends: The friends section will contain all of your friends that you have added. Initially when you have no friends added the friends section will be empty, but once you add a friend that friend will be populated under the friends section.

The box: The box in the middle of the user-interface is going to be where you do the communication with your friend. To start a conversation with a friend you will need to click on the friend that you wish to communicate with under the friends section, then the box in the middle of the screen will initiate a conversation between the logged in person and the selected friend.

Servers: The servers section will contain all of the servers that you have created. Initially you will have no servers because you have not made any just yet, but once you have made a server under the actions section that server will be populated under the servers section.

Use Case Diagram

Communicator: A communicator is the person that is going to be using the system to do certain tasks, such as create an account, login and much more.



Detailed use cases

Name:	Create an account
Actor(s)	Communicator & database
Description:	When the communicator initially goes to my website they will have to create an account. The information they provide will be stored in a database for later use, such as logging in to their account. The communicator needs an account to be able to use my system.
Steps:	1. When the communicator initially enters the system they will be prompted to make an account. 2. Then a screen will show up on the communicator screen asking for some information such as your name, email and password. 3. After the communicator has entered this information that data will be put into the database for later use. 4. Then you will get a message saying “Account created”.

Name:	Login
Actor(s):	Communicator & database
Description:	When the communicator wants to use my system they will be required to login to their account that they made previously. They will be required to input their email and password, then the communicator will click on the login button, the inputted information will be checked and compared to what is in the database.
Steps:	1. After the communicator has made their account their data is stored in a database. 2. Then the communicator will click on a login button, then the communicator will be asked to enter either their name or email and their password. 3. Then the database will check to see if that data is in the database. If it is, they will be allowed access to their account, otherwise they will be denied any access.

Name:	Add a friend
Actor(s):	Communicator & database
Description:	Once the communicator has logged in they will view the user-interface. The communicator will be able to add a friend if they have an account, this means the friend you are trying to add must have an account to be able to add you.
Steps:	1. After the communicator has logged into their account they will be greeted with the user-interface of the system. 2. The communicator will need to get familiar with the user-interface, to find the feature to add a friend. 3. Once the communicator has found the action they want to do they will be shown a pop-up. 4. In the pop-up they will need to enter the name of the friend they want to add. 5. After this is done, depending on what name the communicator entered the database will search for every entry with the entered name and the database will return all of the names depending on what name you entered. 6. Then the communicator will be able to add that friend.

Name:	Send friend request
Actor(s):	Communicator & database
Description:	Once the communicator wishes to add a friend they will need to enter the name of the friend they wish to add then a friend request will be sent, then that friend will either accept the request or not.
Steps:	1. After the communicator has logged into their account they will be greeted with the user-interface. On the user-interface the communicator will start off by trying to add a friend. 2. This will show a pop-up with some information to be entered, you must enter in the name of the friend you wish to add. 3. Once the communicator has entered all of the information, a friend request will be sent to that friend that was entered. 4. The friend will then get a message on their account showing that someone has added them. Then they will have the option to accept the friend request or not. 5. After this is done, the database will update and store that one user is a friend of another user.

Name:	Retrieve a friend
Actor(s):	Database
Description:	When someone signs up to my service they are able to add and delete a friend, but to be able to do this I need to go to my database and retrieve the appropriate data for either adding or deleting a friend.
Steps:	This process is done in the “add a friend” detailed use case 1. To retrieve a friend the communicator will be shown a pop-up and they will need to enter the name of the friend they want to add(retrieve). 2. After this is done, depending on what name the communicator entered the database will search for every entry with the entered name and the database will return all of the names depending on what name you entered.

Name:	Send a message to a friend
Actor(s):	Communicator & database
Description:	Once you have added a friend your friends list under the user-interface will populate the friend that you have just added. To initiate a conversation with your friend, click on your friend's name, then in the middle of the user-interface there will be a box and within this box you will have your conversation with whatever friend you have clicked on.
Steps:	1. To start a conversation with your friend you need to click on the friend that you want to message. 2. Then the box in the middle of the user-interface will initiate the conversation between Kyle and Test for example. 3. Either Kyle or Test can send a message first, then from there the conversation will just continue between Kyle and Test. 4. All of the messages sent from Kyle and Test will be stored in a database.

Name:	Make a server
Actor(s):	Communicator & database
Description:	When the communicator wants to create a server they will click on a button called “create a server”. This will popup a new screen to make your server, you will need to enter the name of your server and possibly some more information. Once you go back to the user-interface under the servers section, the server that you just made and the necessary information you entered will be populated underneath the servers section.
Steps:	1. When a communicator wants to create a server they will need to navigate to the “make a server” section on the user-interface. 2. After the communicator has clicked on the “create a server” they will be shown a pop-up. 3. In this pop-up they will need to enter the name of their server and add some sections within the server, then they will click on “create server”. 4. After they have clicked on “create server” the database will store that new server.

Name:	Send message to server
Actor(s):	Communicator & database
Description:	When the communicator wants to send a message to a server they will be shown a popup, this popup will ask the communicator what server do you want to send your message to. The communicator will then input a message to send to a certain server.
Steps:	1. A communicator will be asked what server do you want to send your message to or they can enter a server then send a message. 2. The communicator can then enter their message to send to a server. 3. This message is then stored in a database.

Name:	Add friends to your server
Actor(s):	Communicator & database
Description:	If a communicator wishes to add one of their friends to their server, they will be shown a popup and within the popup it will look something like the following, add “Tom” to “Week days”, tom being the name of the friend and week days being the name of the server. The communicator will also have the option to add more than one friend to a server.
Steps:	1. In order for a communicator to add a friend to a server they will need to click on “add friend to server” on the user-interface. 2. Then they will be shown a pop-up, the communicator will need to select a friend and a server to add them to. 3. The communicator will have the option to add more than one friend to a server. 4. Then the communicator will click on a button “add friend to server”, this information will be put into a database.

Name:	Delete friends from server
Actor(s):	Communicator & database
Description:	If a communicator wishes to remove one of their friends from their server, it must be the communicator that added that friend to the server, they will be shown a pop-up that will show the communicator something like the following, remove “Tom” from “Week days”, tom is the name of the friend that is being removed from a server called “Week days”. The communicator will also be able to remove one or many friends from their server.
Steps:	1. The communicator will be able to delete a friend from a server if and only if they have added that friend to the server. 2. Otherwise they didn't add that friend to the server so they won't be able to delete that friend from the server. 3. If the friend has been deleted from the server the database will be updated as needed.

Non-Functional Requirements

As my project is a communication website I will need to have functional requirements such as adding a friend etc. But I also must have non-functional requirements. Below are the non-functional requirements that I must have for my project.

Functionality

The system will allow communication allowing users to connect from anywhere in the world. The system will support one to one messaging and one to many messages, this will be achieved with the use of servers.

Usability

The system will provide easy navigation, allowing users to find a certain feature within the system. To ensure high usability of the system, I will design my user-interface so that any feature can be accessed within three mouse movements.

Performance

The system will load in under three seconds.

Security

The system will utilise industry-standard encryption algorithms such as Advanced encryption standard (AES) to ensure the safety of user data.

Reliability

I am committed to ensuring that my product is reliable and trustworthy, ensuring that my system loads in under three seconds. Users can have confidence that the platform will load in under three seconds but also function effectively and efficiently.

Availability

I will have my system up 95% of the time and in operation.

Portability

As my product is a communication system you will be able to use my system on the following devices, laptops and desktops.

Risks

Throughout the development of this project I will encounter many potential risks. Some of the possible risks are described below.

Time

Time might be a potential issue during the development of the project, especially when balancing deadlines with ensuring the system functions properly and as intended.

Features

To manage my time effectively, I will prioritise the most critical features first defined in my use case diagram. By focusing on these features first, I can ensure the essential features are completed then I will focus on the other features.

Database load

During my project development, I will be using a database, and its load capacity potentially might pose a risk for storing user data, such as emails and passwords. If the database cannot handle the load, it could prevent the system from functioning properly by failing to store user information.

Performance

Performance is an issue that might cause me some problems. For example, if my website does not load under three seconds the communicator might click off my system because it is taking too long to load.

References

Wikipedia. (2022). *South East Technological University*. [online] Available at: https://en.wikipedia.org/wiki/South_East_Technological_University [1].

Slack (n.d.). *What is Slack?* [online] Slack Help Center. Available at: https://slack.com/intl/en-ie/help/articles/115004071768-What-is-Slack- [2].

Google.com. (2024). *What is flock software - Google Search*. [online] Available at: https://www.google.com/search?client=firefox-b-d&q=what+is+flock+software [Accessed 12 Oct. 2024] [3].

Erkic, A. (2022). *Slack vs Flock: Which app to choose?* [online] Pumble Blog. Available at: https://pumble.com/blog/slack-vs-flock/ [Accessed 13 Oct. 2024] [4].

Peterson, D. (2016). *Desktop App*. [online] Pinterest. Available at: https://www.pinterest.com/pin/257901516141390344/ [Accessed 18 Oct. 2024] [5].

https://www.facebook.com/flaticon (2020). *Speech bubble Icon - 2462719*. [online] Flaticon. Available at: https://www.flaticon.com/free-icon/speech-bubble_2462719?term=message&page=1&position=17&origin=search&related_id=2462719 [Accessed 18 Oct. 2024] [6].

https://www.facebook.com/flaticon (2021). *Yes Icon - 5129602*. [online] Flaticon. Available at: https://www.flaticon.com/free-icon/yes_5129602?term=yes+or+no&page=1&position=1&origin=search&related_id=5129602 [Accessed 18 Oct. 2024] [7].