YAPP Team Contract

No	Name	Roles	Responsibilities
1	Darlene Gabriele Kosasih (1820221057)	Leader	The team leader for the website chat project will be essential in guiding and managing the entire team to ensure the project's success. The team leader will provide clear direction and support to all team members, making sure everyone understands their specific roles and responsibilities. The leader will actively participate in coding, offering assistance and troubleshooting when needed to overcome technical challenges and maintain high-quality code. The team leader will also be responsible for creating and maintaining detailed project documentation, ensuring that all aspects of the project are clearly recorded and easily accessible. The leader will regularly remind team members of their tasks and deadlines, helping to keep everyone on track and ensure timely completion of each phase of the project. By closely monitoring progress and addressing any issues promptly, the leader will help maintain momentum and ensure that the team works effectively together. Their role is crucial in fostering a collaborative environment, providing motivation and support, and driving the project towards a successful completion of the chat application.
2	Franklin Lo (1820222029)	Front end (Coding)	The team member responsible for front-end coding in the website chat project will focus on designing and implementing the user interface and user experience elements of the chat application. This includes creating and maintaining the layout, visual components, and interactive features to ensure a user-friendly and responsive design. The front-end coder will collaborate with other team members to integrate the front-end with back-end functionalities, ensuring smooth communication between the user interface and server-side components. Additionally, this role involves testing and debugging the front-end code to address any issues and enhance performance, as well as ensuring cross-browser compatibility and adherence to design specifications.
3	Ryan Farrell Witaria (1820222064)	Back end (Coding)	The team member responsible for back-end coding in the website chat project will be tasked with developing and maintaining the server-side logic, database interactions, and application programming interfaces (APIs) necessary for the chat functionality. This includes designing and implementing robust, scalable server-side solutions to handle user messages, manage session states, and ensure secure data storage and retrieval. The back-end coder will work

			closely with the front-end team to ensure seamless integration of server-side operations with the user interface, troubleshoot and resolve any issues related to server performance or data handling, and ensure that the back-end infrastructure supports the overall functionality and performance requirements of the chat application.
4	Andrew Tanujaya (1820222027)	Document	The team member responsible for creating and enhancing project documentation will focus on making the document visually appealing and detailed, ensuring that it is clear and informative for all stakeholders. This includes formatting the document to improve readability, adding relevant details to provide comprehensive coverage of the project, and ensuring that all sections are well-organized and easy to navigate. In addition to their documentation duties, this team member will also contribute to coding tasks as needed, collaborating with others to develop and implement both front-end and back-end components of the website chat application. Their dual role in both documentation and coding will help maintain a high standard of project quality and ensure effective communication throughout the development process.

YAPP Contract

This project was given the name YAPP as by definition YAPP has a meaning of talking too much, hence given the name YAPP. The YAPP application aims to provide real-time communication, whether it be with two numbers of parties or more, in other words a private chat or a group chat. The main method of communication of YAPP is via text messages, as opposed to voice or video calls. As for now, YAPP aims to connect users over a network, meaning users have to be connected to the same internet ip address in a close proximity to be able to send the text messages.

The YAPP application requires a few aspects to be present in order to work. First and foremost, an application needs a Client that will open a network connection with the IM server at a specified IP address and port number. This client requires a method to specify the server's IP address, port, and user's username to connect with the network in hand. Once the client connection is open, the user will be able to access the application and its many features.

As for the means of communication itself, a communication application also requires a server to run on, especially when it's a network based communication application. As the server supports online communication, it most definitely will allow conversations between 2 or more individuals. As the means of connection is via network, the conversations made between clients should be instantaneous and separated from chats with different clients within the same network via a different chat panel.

The client and server interaction would have multiple important points. Client and server interactions first and foremost provide the functionality to view which users are currently logged into the client. It would also provide a functionality to create new conversations, joining existing conversations, and leaving active conversations. On top of that, the application would also have the function to have multiple active conversations simultaneously at the same time meaning the same client could have multiple incoming messages from other clients at the same time. And above all that, each chat would have the function to store the chat history between clients either it being between two or more clients.

As for how the clients work, users are able to log in or register as a client and claim any username they choose, this is named as a non authentication system. The client would require password authentication but client can claim any username they choose.

About the client and server protocol, clients and servers require a set of commands or in other words a list of procedure to allow clients to perform communication between 2 or more clients. This specification ensures as grammar of the client and server protocol. This client and server protocol will also represent the state of the client, as the basis of how the protocol is put together can reflect how heavy the application is when it is being runned.

We also create YAPP to ensure it has a usability design, where it provides users comfort and flexibility. YAPP as its basis would provide 2 major compartments, which is the login or regsitration screen and the chatbox itself, quickly accesible, providing users ease of usage and minimalistic, taking away the unecessary parts for the important parts. This is important as these days, majority of device users have low attention spand, hence having a minimalistic design supports the users to chat and communicate in a better environment.

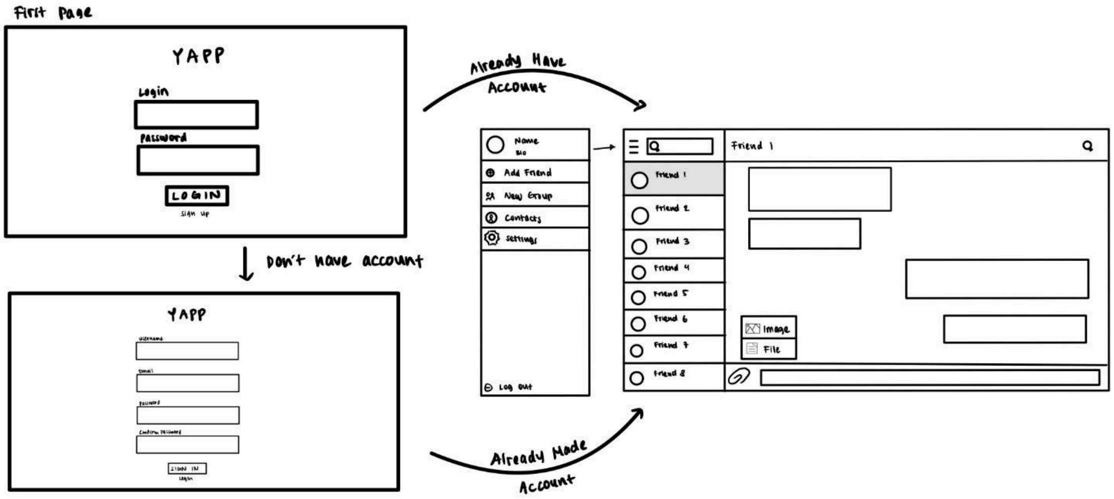

Image 1
Conversation Design

For this project, we'll be making a communication based application named YAPP where it will consist of 3 main pages which consist of the Login Page, Registration Page, and the Chat Page. The Login Page will consist of a Login Username and its password, while for the Registration Page, it will consist of the Username creation, the users Email, the users Password, and the Passwords confirmation. As for the Chat Page, it will consist of the functionality to send chats to the other person in hand together with the functionality to select who we want to send the messages to.

Other than that, the Chat Page also has the function to send images or files from the users device, regardless of the image or file type as well as the ability to open the Expand Options panel. In the Expand Options panel, the user is given flexibility and freedom to Add Friends based on their Username, check their personal contacts, open extra settings, and the ability to log out from the current account. Within the extra settings, users will be given the ability to invert the application color from light mode to dark mode and vice versa. This way, the user will be given more space and flexibility.

With the client and server protocol that YAPP provides, it creates a concurrency strategy that minimizes the chances that YAPP would encounter deadlocks. YAPP ensures minimum chances of deadlocks by creating the application work over a single network, this way it is clear the procedure it will follow between clients. This way, YAPP as a communication application should not encounter any deadlocks.

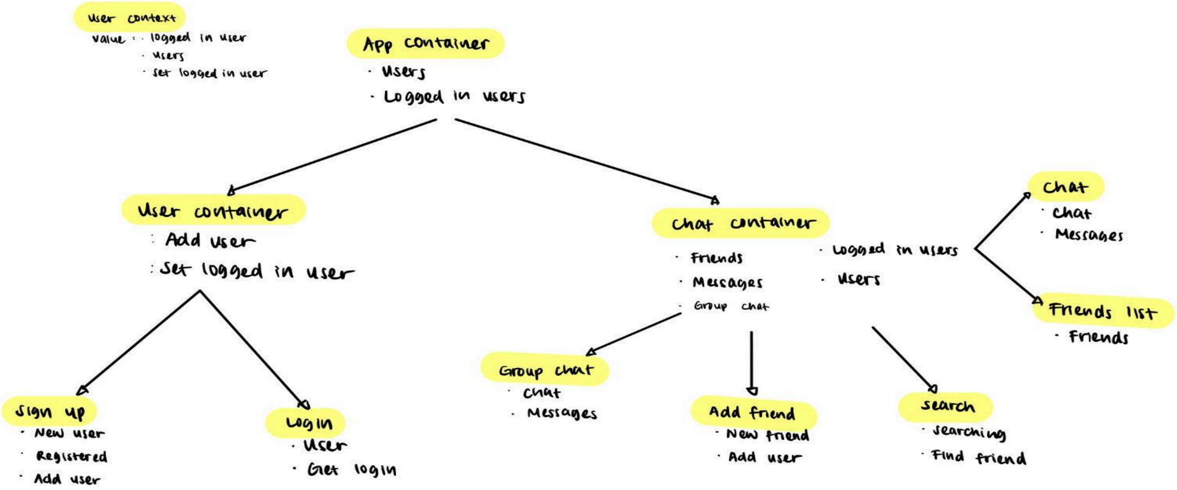

Image 2
Client Server Protocol

As for the algorithm that will be applied to this project, we'll be using a rather simple set of algorithm consisting of simple procedures on how the application is meant to be used. By the basics of this procedure, we'll have 3 basic user value which consists of "Logged in User", "Users", and "Set logged in user". "Logged in User" meaning the the user gets an option to pick which account they'd like to log in to, "Users" meaning the input of User account information in said device, and "Set logged in user" meaning the user that has been decided for said device.

Within the app container, which is the basic of the app itself, it provides a list of the accounts that has been registered to the device and an option to pick which account the user would like to log in to. As for the user container, it adds the option to add new users or set which account will be used for the device. On the other hand, the chat container consists of 5 functions, to select friends, messages, group chats, and the ability to select which account the device user would like to change to from an internal list of accounts.

Based on the user container page, it has 2 procedure based on it which are the sign up function and the log in function. The sign up function provides to create new user account, register it, and the ability to add that user to the list of users. As for the login function, it allows us to user list down the users information and log in to it based on the username and password that has been inputted into the applications system.

On the other hand the chat container page, it provides 5 function for the device user to utilize which are group chat, add friend, search, search, and chat function with each having its own algorithm and list of procedures. The group chat and chat is meant to send chat and read messages where it only differs in where the device user is willing to send messages to, either as private messages or group messages in group chats. The add friend feature allows the device user to input another users username and add them to the friends list that are connected to the current account. The search feature allows the account user to search past chats from the messages sent with the selected friend.

This is the current concept and list of procedures that the our team will be creating for this project application named as YAPP. As of now, YAPP idealizes to be a functional application that can connect people together from one account to another, this way it can connect people via online communication platforms. It also aims to provide user comfort and application modification to ease the users comfort while using the application, giving users flexibility to adapted to each users comfort.

YAPP can be ensured that it will run properly without any deadlocks by automatic testing. Once a client chats another client via the chosen user account, it will send a signal and message to the other client that a message is present and requires the other clients attention. This way, the user won’t need to revisit the code designed whenever a signal isn’t sent to the other clients device. Thus ensuring it’s safety, functionality, and flexibility for personal and multi-client based usage based on the design and protocol that has been implied to YAPP as a communication based application.

Signatures,

Darlene Gabriele Kosasih	Franklin Lo	Ryan Farrell Witaria	Andrew Tanujaya