

## COSC 310 (Team Bamboozlers)

### Requirements Document

#### What the project is:

Discord Clone

#### Description:

Our project will be a Messaging Platform allowing users to create accounts, add friends and create groups with or send and receive messages to/from. It will also allow users to become moderators of groups and control members permissions within said groups.

### 1. Introduction

#### *1.1 Purpose of the Requirements Document*

The purpose of this document is to outline the user requirements and the functional purpose of the Discord Clone, a messaging platform. It will serve as a comprehensive guide for the development team to understand what features and functionalities are required, ensuring that the final product aligns with user expectations and project objectives.

#### *1.2 Scope of the Product*

The Discord Clone is a messaging platform that enables users to create accounts, add friends, form groups, and communicate through messages. It also includes features for users to act as moderators in groups, managing member permissions and overseeing group activities.

#### *1.3 Definitions, Acronyms, and Abbreviations*

- **User:** An individual who interacts with the Discord Clone platform.
- **Moderator:** A user with elevated permissions within a group to manage members and settings.
- **Group:** A chat environment where multiple users can communicate simultaneously.
- **DM (Direct Message):** A direct message sent from one user to another.

#### *1.4 References*

Discord: <https://discord.com/>

### *1.5 Overview of the Remainder of the Document*

The remainder of this document is divided into sections detailing the general description of the product, specific requirements including functional and non-functional aspects, followed by appendices and an index for easy navigation.

## **2. General Description**

### *2.1 Product Perspective*

The Discord Clone is an independent product, designed to provide an intuitive, user-friendly messaging experience. It aims to incorporate the best features of existing messaging platforms while introducing unique functionalities tailored to user needs.

### *2.2 Product Functions*

- Account creation and management
- Friend list management
- Group creation and management
- Messaging (both private and group)
- Moderator roles and permissions in groups

### *2.3 User Characteristics*

Users are individuals seeking a communication platform. They range from casual users who want to keep in touch with friends to businesses, teachers, and group leaders who need to manage large groups.

### *2.4 General Constraints*

- The platform must ensure data privacy and security.
- It should be scalable to accommodate a growing number of users.

### *2.5 Assumptions and Dependencies*

- Continuous internet connectivity for users.

### 3. Specific Requirements

#### 3.1 User Requirements

- Users must be able to register, login, and manage their profiles.
- Users must be able to add and remove friends.
- Users must be able to create and manage groups
- Users must be able to send, receive, edit, and delete messages.
- Moderators in groups must be able to manage member roles and permissions.

#### 3.2 Functional Requirements

- The system must allow Users to register using their email.
- The system must save the account information of users to the database.
- The system must send a verification email when a new User attempts to register.
- The system must send a confirmation email when a user successfully registers, changes their password or email.
- The system must allow users to update their profile and username.
- The system must update the database when Users edit their account information.
- The system must allow users to change their email or password.
- The system must allow users to access their friend list.
- The system must allow users to send and accept friend requests.
- The system must update the database and users friend lists when they accept friend requests.
- The system must support the creation and management of groups.
- The system must update the database and message history when users send, receive, edit, pin, and delete messages.
- The system must allow moderators to manage memeber roles and permission.

- The system must update the database if moderators change member roles and permission.

### *3.3 Non-Functional Requirements*

- **Performance & Reliability:** The system should be able to handle multiple users performing system functions such as messaging, registering, friend requesting.
- **Scalability:** Should accommodate an increasing number of users and groups while maintaining performance and reliability.
- **Security:** Robust security measures to protect user data and privacy. Policies such as email verification when resetting password and safe password and email storing.

## **4. Appendix**

### *4.1 Use Case UML Diagrams*

![requirements\usecasediagram.png](requirements/usecasediagram.png)