Documentation DiscordClone

# 1. 🛠 Setup

- The project is created as a Visual Studio project named `DiscordClone`.  
- The project is version-controlled using Git and uploaded to GitHub.

# 2. 📋 Planning

## Project Description

I have created a user interface inspired by Discord. This project is not a functional communication tool, but rather a recreation of the visual look and user experience.  
The goal is to learn about UI design by mimicking a popular and familiar application.  
Target group: Teenagers interested in technology and user experience.

## Sketch (Mockup / System Flow)

A visual mockup has been created showing the structure of the user interface, including:  
- Channel list  
- Message area  
- User information

## Project Management

# 3. 👨‍💻 Development

## Code Quality

The code is written in C# and organized with clear structure and comments. Since this project focuses only on the interface, no backend or messaging functionality is implemented.

## Algorithms and Data Structures

To meet the requirement for algorithms and data structures:  
- Lists (`List<string>`) are used to represent messages and channels.  
- A simple display order (message sorting) is included as an algorithmic element.

# 4. ✅ Testing

## Test Plan

Since this is only a user interface without active backend, testing focused on visual and structural elements:  
- Are elements placed correctly and in the right order?  
- Does the interface adapt to different window sizes?  
- Does navigation between channels work as expected?

## Test Results

No functional errors, but some elements did not adjust correctly to smaller screens. This was resolved by applying simple dynamic sizing rules in the interface.

# 5. 📚 Documentation

## User Guide

1. Start the program from Visual Studio.  
2. The interface will show a list of channels and a message display area.  
3. Click a channel to view its messages (static content).  
4. Explore how the design and layout mimics Discord.

## Technologies Used

- C#  
- Windows Forms / WPF  
- Git & GitHub – for version control and project management  
  
Reason for technology choice: C# with Visual Studio is well-suited for desktop applications and allows rapid UI development. GitHub supports collaboration and progress tracking.

# 6. 🎤 Presentation

A short presentation will demonstrate:  
- The goal and scope of the project  
- A walk-through of the interface  
- The development process using sketches and GitHub

# 📊 Evaluation Criteria – Self Assessment

|  |  |
| --- | --- |
| Category | Status |
| Planning (20%) | Clear description and visual sketch created |
| Implementation (60%) | Code is clean and structured, demonstrates UI design skills |
| Testing (10%) | Test plan executed for layout and design |
| Documentation (10%) | All documentation provided in this report |