Dokumentasjon DiscordClone

# 1. 🛠 Opprett

- Prosjektet er oppretta som eit Visual Studio-prosjekt under namnet `DiscordClone`.  
- Prosjektet er versjonsstyrt med Git og lasta opp til GitHub.

# 2. 📋 Planlegging

## Prosjektbeskrivelse

Eg har laga eit brukargrensesnitt som etterliknar designet og opplevinga av Discord. Dette prosjektet handlar ikkje om å lage eit funksjonelt kommunikasjonsverktøy, men om å gjenskape det visuelle uttrykket og brukaropplevinga.  
Målet er å lære om grensesnittutforming og design ved å etterlikne eit kjent og populært program.  
Målgruppe: Ungdommar som er interesserte i teknologi og brukaroppleving.

## Skisse (Mockup / Systemflyt)

Ein visuell skisse er utarbeidd og viser oppbygginga av brukargrensesnittet, inkludert:  
- Kanalliste  
- Meldingsområde  
- Brukarinformasjon

## Prosjektstyring

# 3. 👨‍💻 Utvikling

## Kodekvalitet

Koden er skriven i C# og er organisert med oversiktleg struktur og kommentarar. Sidan dette prosjektet berre fokuserer på grensesnittet, er det ikkje implementert back-end eller funksjonell logikk for meldingar.

## Algoritmar og datastrukturar

For å møte kravet til datastrukturar og enkel algoritmisk logikk:  
- Det er brukt lister (`List<string>`) for å representere ei oversikt over meldingar og kanalar.  
- Det er lagt inn ei enkel visningsrekkefølge (sortering av meldingar) som algoritmisk element.

# 4. ✅ Testing

## Testplan

Ettersom dette berre er eit brukargrensesnitt utan aktiv backend, har testing fokusert på visuelle og strukturelle element:  
- Er elementa på riktig plass og i riktig rekkefølge?  
- Tilpassar grensesnittet seg ulike vindusstorleikar?  
- Fungerer navigasjonen mellom kanalar?

## Testresultat

Ingen funksjonelle feil, men det vart oppdaga at enkelte element ikkje justerte seg riktig ved mindre skjermar. Dette vart løyst med enklare dynamisk justering av størrelsar i grensesnittet.

# 5. 📚 Dokumentasjon

## Brukerveiledning

1. Start programmet frå Visual Studio.  
2. Brukargrensesnittet vil vise ei liste med kanalar og eit område for meldingar.  
3. Klikk på ein kanal for å sjå meldingane (statisk innhald).  
4. Utforsk korleis design og struktur etterliknar Discord.

## Teknologiar brukt

- C#  
- Windows Forms / WPF  
- Git & GitHub – for versjonskontroll og prosjektstyring  
  
Grunngjeving for teknologival: C# med Visual Studio er godt egna for å lage desktop-applikasjonar og gir rask utvikling av brukargrensesnitt. GitHub gjer det enkelt å samarbeide og spore framgang.

# 6. 🎤 Framvisning

Ein kort presentasjon vil bli brukt til å:  
- Forklare målet med prosjektet og kva som er laga  
- Vise fram brukargrensesnittet i praksis  
- Fortelje om arbeidsprosessen og korleis skisser og GitHub vart brukt i utviklinga

# 📊 Vurderingskriterier – Eigenvurdering

|  |  |
| --- | --- |
| Kategori | Status |
| Planlegging (20%) | Klår beskrivelse og visuell skisse oppretta |
| Gjennomføring (60%) | Koden er ryddig og strukturert, viser god innsikt i UI-design |
| Testing (10%) | Testplan gjennomført for design og layout |
| Dokumentasjon (10%) | All dokumentasjon er levert i denne rapporten |