

## Bakgrund

Detta är slutprojektet i kursen mobilutveckling.  
Jag valde att göra en chat app som är kopplad till Firebase.

## Planering

Projektet delade jag upp i 5 milstolpar och för närvarande 18 kanbankort.  
Projektverktyget jag använde är github projects.  
Där kopplade jag varje kort till en milstolpe så man kunde tydligt se när den var klar.

**Milstolpar** - <https://github.com/tallner/GritMobileSprintV2/milestones>

MS1 Planning - <https://github.com/tallner/GritMobileSprintV2/milestone/1> - 22/10

MS2 GUI Design - <https://github.com/tallner/GritMobileSprintV2/milestone/2> - 24/10

MS3 Implementation activities - <https://github.com/tallner/GritMobileSprintV2/milestone/3> - 28/10

MS4 Implementation fragments - <https://github.com/tallner/GritMobileSprintV2/milestone/4> - 01/11

MS5 Report and deployment - <https://github.com/tallner/GritMobileSprintV2/milestone/5> - 03/11

## Layout

Appen är uppbyggd med 4 activities och 3 fragments

StartActivity – startpunkt för appen

LogInActivity – inloggning

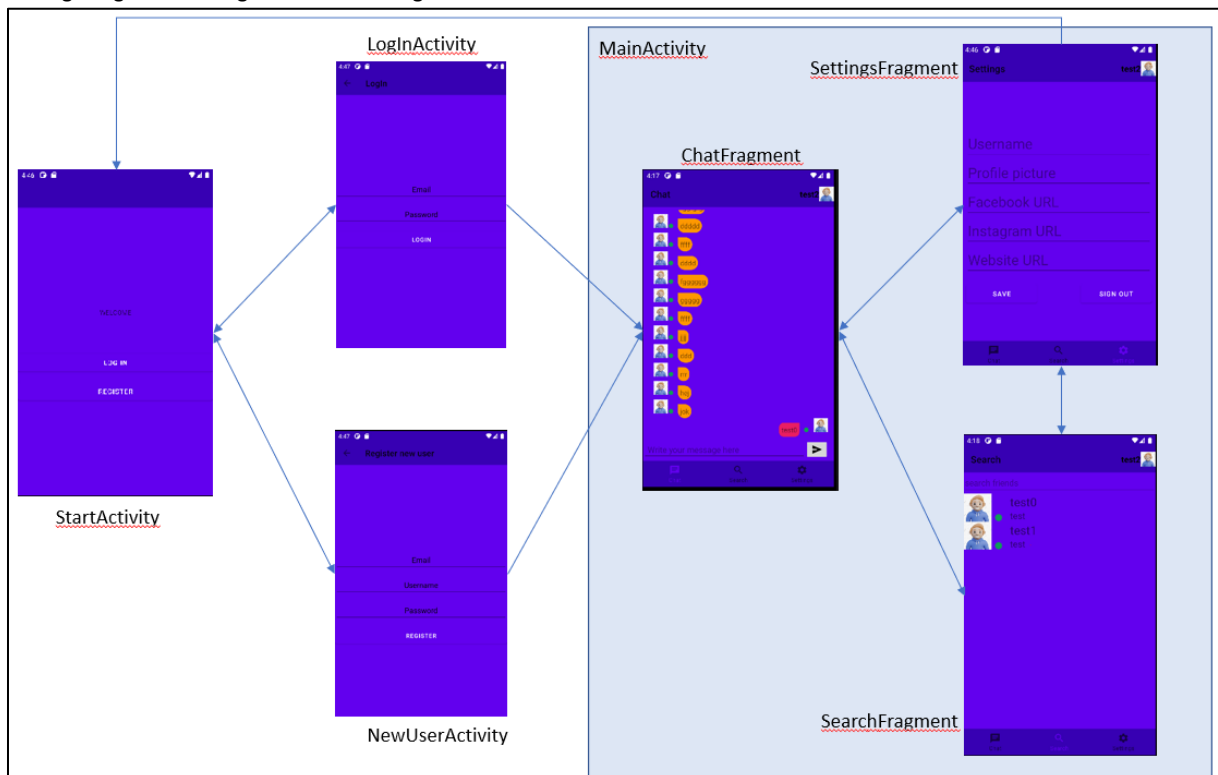
NewUserActivity – skapa ny användare

MainActivity – hanterar fragmenten när man är inloggad

ChatFragment – här ser man sin nuvarande chat

SearchFragment – här letar man upp användare och väljer vem man ska skriva till

SettingsFragment – här gör man inställningar för sin användare



## Tekniska val

### Fragments och Activities

Jag valde att använda både fragments och activities för att lära mig att kombinera dom i ett projekt.

Man kunde använt enbart fragments och då hade navigering varit lite enklare för att man kunde använt enbart bottomnavigationbar.

### Firebase services

Som databas för att spara användarinformation och chattar använder jag Realtime Database.

För inloggning och användarhantering används Authentication.

För att spara bilder använder jag Storage.

### Databas

Jag valde att använda Realtime Database därför att vi hade redan använt Firestore i andra projekt.  
Om jag skulle vidareutveckla appen hade jag gått över till Firestore.

### **Programbeskrivning**

Navigering mellan activities sker med Intent.

Navigering mellan fragment sker med NavController och bottomNavigationBar

### StartActivity

Startpunkt för appen

Väljer enbart om man går vidare till inloggning

### LoginActivity

Tar in email och password och verifierar mot Firebase med FirebaseAuth.

Om inloggningen är lyckad skapas en ny task för att detta inte ska vara med på backstacken i MainActivity:

```
i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK or Intent.FLAG_ACTIVITY_NEW_TASK)
```

### NewUserActivity

Registrerar en ny användare på Firebase med FirebaseAuth.

Den nya användarens information sparas i databasen.

### MainActivity

Här sätts bottomBar navigeringen upp.

### ChatFragment

En recyclerView används för att visa chattmeddelanden.

En addValueEventListener används för att se om det finns förändringar i datan i den angivna referensen.

Om det finns så loopas datan igenom och sparas i en ArrayList.

Denna ArrayList skickas sen in i en ChatAdapter som i sin tur skickas in till RecyclerViewn.

Det ChatAdapter gör är att den väljer hur och vilken data som ska visas i RecyclerView.

SharePreferences används för att spara vilken användare man chattar med så att dessa meddelande läses in automatiskt när man kommer till chatFragment igen.

### SearchFragment

Fungerar ungefär som ChatFragment med RecyclerView och en UserAdapter.

En skillnad är att här finns två funktioner för att hitta alla användare eller söka efter en enskild användare.

### SettingsFragment

Funktionalitet ej ännu implementerad.

Här ska användaren kunna göra inställningar för sin profil.