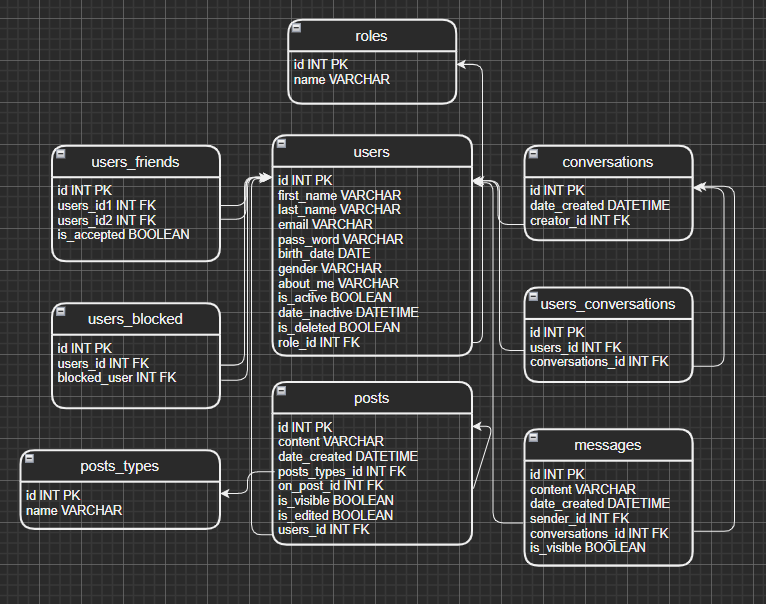
# Rapport Databas-projekt



## Databasdesign Facebook behöver ha användare. Användare ska kunna bli vän med varandra. En användare ska kunna blocka en annan användare. Användare ska kunna skriva inlägg och kommentarer på inlägg, kommentar på kommentar osv. Användare ska kunna ha konversationer med så många andra användare dom vill. I varje konversation ska man kunna skriva meddelanden.

# Djupdyk

Problem  
Hur jag ska lösa en liknande Messenger-struktur.   
1. Gruppkonversationer. Hur möjliggör jag att samma grupp av användare kan ha flera olika konversationer med varandra?  
2. En konversation ska kunna finnas mellan användare utan att de ännu skrivit meddelanden.  
3. Mellan två personer ska det finnas en enda konversation som alltid är mellan dessa två, eftersom Messenger fungerar så.

Åtgärd  
1. En kopplingstabell med två främmande nycklar som pekar på användare-tabellens primärnyckel och konversation-tabellens primärnyckel. Det möjliggör för en användare att delta i flera konversationer och för en konversation att ha flera användare. Meddelandet man skriver hör till konversationen (samt vem som skrivit det). Då kan vi hämta ut alla meddelanden per konversation. Alltså kan tex. Olle, Anna och Stefan ha två olika konversationer med meddelanden som handlar om jobb i ena och bara roliga bilder i den andra.

2. Eftersom deltagare är kopplade till konversationer i en kopplingstabell och inte till meddelandet så kan man skapa konversationer utan att behöva skriva ett meddelande.  
  
3. Eftersom kopplingstabellen mellan användare och konversation möjliggör att två specifika personer kan ha många konversationer så måste det antingen till en dialog-tabell där två användares id står eller så får det lösas i C#-kod. Som det är nu väljer jag att lösa det i C#-kod. Det är onödigt att lägga till fler tabeller om det inte behövs, det är samma information som ska lagras så därför är detta bäst löst via funktioner anser jag.   
Tex. om en konversation finns mellan två specifika användare, så skapas ingen ny.

Minidyk 2  
Problem  
Radera all data genom att använda cascading delete eller flagga data som raderad?  
  
Åtgärd  
Jag valde att flagga data som raderad pga. att en användare kan radera sitt konto men ångra sig en stund senare och vilja ha viss information tillbaka. När jag har läst om raderade data så har jag även förstått att viss information som en primär nyckel ska helst inte raderas då det kan bli problematiskt om något annat sätts på samma nyckel senare. I mitt program kan en användare inaktivera sitt konto och om de inte loggat in inom 30 dagar så flaggas de som raderade och kan inte komma åt sin information.

# Utveckling Raderade data De användare som varit inaktiva i mer än 30 dagar flaggas som raderade. Detta ska jag ändra så information som kan knytas till en viss personen ska raderas också pga. GDPR.

Messenger  
Utöka kopplingstabellen där användare och konversation kopplas. Tex. lägga till ett attribut som håller information om när användaren senast läste i konversationen.   
Utöka meddelande-tabellen med attribut som bild som är av datatypen BLOB.

Notiser  
Användare ska få notiser vid ny händelse (tex vän förfrågan, kommentar).   
Jag tänker lösa det med en notis-tabell. Även en trigger-funktion tex. insättning till notis-tabellen efter insättning till vän-tabellen.  
  
Användare-profil  
Användare ska kunna lägga upp bilder, både via inlägg och direkt till ett album. Då behövs en tabell *fotoalbum* och en tabell *foton*. Foton får ha en främmande nyckel som pekar på fotoalbum då ett foto tillhör ett album och ett album kan ha flera foton.

Angelina Holmqvist SUVNET22