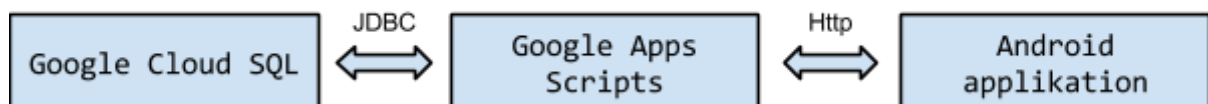


## Android SDK targets

- Minimum SDK: 19
- Target SDK: 19

## Mjukvarans uppbyggnad

Mjukvaran består av tre delar. En databas som tillhandahålls av Google Cloud, en frontend applikation som användaren kan interagera med, och ett antal Google Apps Scripts som sköter kommunikationen mellan applikationen och databasen. En illustration av mjukvaruarkitekturen kan ses i figur 1 nedan.



Figur 1 En illustration som beskriver mjukvaruarkitekturen

För kommunikation mellan databasen och Google Apps script används JDBC. Alla scripts är publicerade som webbapplikationer och androidapplikationen använder dessa genom Apaches HTTP bibliotek. Ett HTTP-get request används för att applikationen, informationen finns sparad i textsträngar. nropa ett specifikt script via dess unika URL. Ett Http-response erhålls då av mobilap

## Applikation

Android applikationen består av ett antal olika typer av javaklasser:

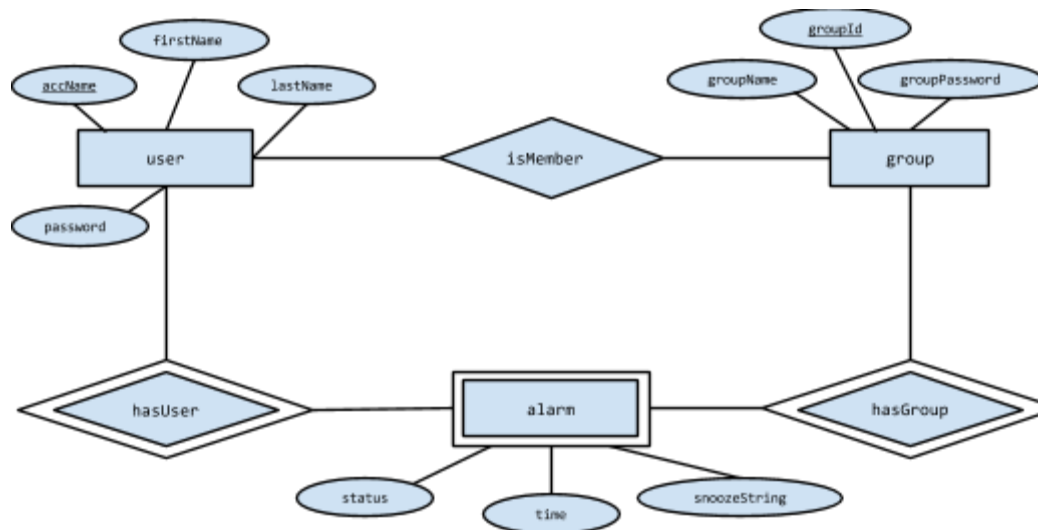
- ListAdapters
  - Adapters som används för att översätta listor av information till grafiska komponenter.
- ListElements
  - Objekt som representerar data för att visas grafiskt, i detta fall klasserna GroupClass och MemberClass.
- Activities
  - De skärmar, fönster, som användaren kan interagera med.
- BroadcastReceiver
  - Den klass som körs då ett alarm startas, i detta fall AlarmReceiver.
- AsyncTasks (privata och publika)
  - Klasser som används för att utföra arbete på en annan tråd än huvudtråden, används i huvudsak för anrop till databasen.

När information sparas lokalt på den mobila enheten används ett Android SharedPreferences objekt och all information sparas i key-value par.

## Databas

Databasen är en mySQL-databas som tillhandahålls av Google Cloud Services.

Databasen kan anslutas till med en extern databasredigerare för adminuppgifter, så som att skapa tabeller och attribut. I Figur 2 visas ett ER-Diagram för databasen.



Figur 2 Ett ER-diagram för databasen

Attributen talar för sig själv för *group* och *user*. I tabellen *alarm* sparas larmets status som en INT, där 0 är ett avstängt larm, 1 ett aktivt och 2 en snooze. *Time* är tiden larmet går igång, och *snoozeString* är ett potentiellt meddelande från en annan användare som visas när larmet aktiveras.

## Google Apps Scripts

Google Apps Scripts är ett scriptspråk skapat av google som bygger på javascript. Scripten implementeras direkt i webbläsaren och publiceras som webbapplikationer. När ett scripts URL anropas via ett Http-get request körs doGet() metoden i scriptet. Applikationen skickar med relevant information till scriptet i de fall där scriptet behöver detta för att interagera med databasen. Exempelvis skickas användarens användarnamn och ett valt grupp-id med när en användare vill registrera sig i en ny grupp.

## Applikationens larmfunktion

När användaren sätter ett larm skapas ett PendingIntent med tillhörande grupp-id och unik hashkod. Detta PendingIntent skickas till androids AlarmManager klass, där det aktiveras vid en given tidpunkt. När denna aktivering sker notifieras de lyssnare som lyssnar på Intentet, i detta fall klassen AlarmReceiver.

AlarmReceiver gör ett nytt Intent, med samma grupp-id som det Intent den tog emot, och använder detta för att öppna en activityn ActiveAlarm. Det är här larmet går igång för användaren. När larmskärmen i ActiveAlarm skapas laddas också den tillhörande snooze-strängen ned från databasen och visas på skärmen.

På denna skärm finns en snooze-knapp och en stäng av-knapp. När användaren stänger av larmet anropas ett google-script, vilket ändrar statusen för larmet från 1 (aktiv) till 0 (inaktiv). Vid en snooze notifieras databasen på samma sätt, fast med siffran 2 (snooze). Vid en snooze sätts också ett nytt larm efter en viss tid. Denna tid kan ställas in i ActiveAlarm.

Vid de ovannämnda stegen notifieras databasen genom ett anrop till ett Google Apps Script med relevant information för att hålla databasen uppdaterad. Till exempel uppdateras databasen varje gång ett nytt larm ställs och när ett existerande larm sätts på eller stängs av.

### **Applikationens gruppfunktion**

När en användare loggar in i applikationen hamnar den i aktiviteten Groups. När denna aktivitet skapas anropas ett Google Script som returnerar en lista på samtliga grupper användaren är medlem i. Denna lista tas om hand i applikationen och visas upp grafiskt för användaren. Varje objekt i denna lista representerar ett GroupClass objekt och har tre variabler: larmstatus, larmtid och grupp-id, där larmstatus och larmtid är användarens larm för den gruppen.

När en användare vill gå in i en ny grupp anger den gruppens grupp-id och lösenord. Dessa skickas till ett Google Script som sedan kollar om denna information stämmer överens med det som finns i databasen. Om så är fallet läggs användaren in i gruppen och applikationen blir meddelad.

När en användare trycker på en grupp den är medlem i skapas ett intent som tar användaren till aktiviteten InsideGroup. När denna aktivitet skapas anropas ett Google Script som returnerar en lista på samtliga medlemmar i gruppen. Denna lista tas om hand i applikationen och visas upp grafiskt för användaren. Varje objekt i denna lista representerar ett MemberClass objekt och har tre variabler: larmstatus, larmtid och användarnamn.

När användaren trycker på en medlem i gruppen dyker det upp en textinmatningsruta och en OK-knapp. Denna ruta är aktiviteten sendSnoozeString och används för att skicka ett meddelande till ett specifikt larm i databasen. Detta meddelande kommer då sparas på det existerande larmet för den valda användaren. När denne användares larm sedan går igång kommer denna sträng att laddas ned och visas. På samma sätt kan ytterligare funktioner tänkas läggas till, exempelvis ljud, bild och videoöverföring.

När en användare går ur en grupp anropas ett Google Script som raderar de tillhörande raderna i databasen. När en användare loggar ut raderas larmen på samma sätt.