

TDP028 - Reflektionsrapport

Under projektets gång har jag följt kodstilen för Android och varit noga med att inte dra för mycket ut i egna svängar. Det har medfört att jag börjat med ett projekt som utgår från en automatiskt genererad mall. När mallen är genererad finns en filstruktur, utifrån filstrukturen är det svårt att inte följa MVC mönstret.

Jag har en MainActivity som är grundstommen i hela projektet. Från MainActivity åker det in och ut olika fragments. Varje fragment har sin egna fil och klass och använder några av de metoder som är grundläggande för ett fragment.

Efter en rotation i Android är det om att en Activity startar på nytt. Detta är lite jobbigt för mig eftersom jag då behöver använda kod som sparar min "state" för både activity och fragment. En kringgående lösning för detta var att inte tillåta rotation för applikationen, men detta är inte en snygg lösning och det kändes inte särskilt mycket Android. Så senare i projektet så har kod implementeras som tillåter rotation av appen. Jag tycker själv att det varit givande att implementera så att rotation fungerar eftersom det har fått mig att inse vad för viktigt variabler jag använder mig av. T.ex. i parkerings fragment för Google Maps har jag behövt spara undan vart personen tittar på kartan. Det var inget jag själv tänkte på tills jag fick buggen i knät.

Ibland vill jag kunna prata från Fragment till Activity eller från Activity till Fragment. Det är ingen "lätt" uppgift i Android. T.ex. så medför kommunikation Activity->Fragment att man behöver spara undan information i en bundle tillsammans med en nyckel och sedan "packa upp" denna bundle när fragmentet startas. För att prata med Fragment->Activity behöver Activity skapa en "lyssnare" för en funktion som Fragment kallar på. I min kod använde jag dessa två metoder för att skicka runt DatabasReferensen i FireBase, men senare i projektet la jag märke till att referensen för firebase var något jag kunde komma åt globalt, vilket gjorde att jag blev av med mycket "onödig" kod som jag skapat.

Prestandamässigt kan jag påpeka att RecyclerViewAdapter för listan är något som ökar prestandan för stora listor genom att bara ladda in "Views" för de items som visas just då, vilket är bra. Jag har dock ett litet problem och det är att jag tror ImageLoader laddar en bild på UIThread vilket för att appen kan ibland vara oresponsiv när den startas. Men, jag vet inte riktigt vad problemet är, det kan också vara så att inloggningen med google stannar upp UIThread. Parkeringshistorik laddar dock ALL historik, vilket ibland skapar lite laddtid.

För övrigt kan jag påpeka att jag försökt använda Android Studio på bästa sätt genom att använda mycket av den automatiska genererade koden. Android Studio hjälper också en att skriva @Override funktioner, vilket har varit väldigt skönt.

Allt som allt är jag nöjd med projektet, jag tycker att det har varit viktigt att försöka koda rätt och inte "improvisera". Jag är relativt nöjd med koden men appen i sig är inte särskilt användbar. Jag skulle också kunnat förbättra analysen i appen.