

Utvecklarmanual

DAT255, Software engineering project
Chalmers, VT 2012

Patrik Ackerfors	861017-4990
Christoffer Kjernald	870820-5516
Erik Olesund	880917-2490
Markus Ulenius	880320-5635

Byggprocedur

1. Starta Eclipse
2. Installera EGit-plugin samt Android SDK API 7 till Eclipse vid behov
3. Klicka på File -> Import -> Git -> Projects from Git
4. Lägg till URI git@github.com:ackerfors/Listigt.git med SSH-protokoll
5. Välj master branch och import into existing projects
6. Välj ListigtAndroidProject och tryck finish.
7. Nu kan applikationen byggas och köras på emulator eller Android-telefon.

Applikationen använder sig av Android 2.1 API level 7 och har testats med emulator både med API level 7 (plattform 2.1) och med API level 15 (plattform 4.0.3), samt på en Samsung Galaxy S med operativsystem Android 2.3.3.

Färdiga APKer finns i olika versioner av applikationen i mappen dist.

Applikationens klasser

Appen består av nio klasser som är kommenterade med JavaDoc. Ytterligare beskrivning av klasserna, dess funktion och hur de skickar data till varandra presenteras dock nedan.

MainListActivity

En aktivitet av typen ListActivity. Den klass som instansieras först för att öppna och skapa en databaskoppling, sätta en layout samt populera vyer med data från databasen. Databasen kommuniceras med genom att en instans av ListsDbAdapter-klassen skapas i onCreate()-metoden. Sedan körs den interna metoden fillData() där vi använder oss av db-adaptern för att hämta en Cursor med alla titlar för de listor som skapats sedan tidigare. Sedan populeras en vy, list_row.xml, för varje titel som finns i databasen och visas som en lista på skärmen för användaren. Layouten för aktiviteten är main_list.xml.

När användaren klickar på en lista skapas ett Intent där en extras läggs till och skickas till den nya aktiviteten; MainItemActivity. Om användaren håller nere fingret över en lista kan denne välja Edit eller Delete. Om edit så startas en ny aktivitet, ListEditor, och denna får informationen

genom Intent extras.

MainItemActivity

Fungerar i princip på samma sätt som MainListActivity. Det som skiljer är att db-adaptorn som instansieras är ItemsDbAdapter, och denna används i sin tur på samma sätt. När denna aktivitet kör onCreate()-metoden så hämtas vilken lista som användaren klickade på i föregående aktivitet, MainListActivity genom `LIST_ID = extras.getLong("_id")`. Denna information används sedan i metoden `fillData()` för att db-adaptorn ska hämta bara de items som tillhör den listan som användaren valt.

Aktivitetens layout finns i `main_items.xml` och den individuella raderna som `fillData()` skapar med items finns i `items_row.xml`. Där har även en checkbox implementerats för att veta om item är bokad eller ej. Databaskopplingen är ännu ej fullt implementerad.

AbstractListigtDbAdapter

Består av en ytterklass, `AbstractListigtDbAdapter`, och en innerklass, `InnerSQLiteOpenHelper`, som i sin tur utökar klassen `SQLiteOpenHelper`.

Klassen innehåller mycket kommentering i JavaDoc så en kortfattad beskrivning kommer här att ske. Klassen skapar, öppnas eller uppdaterar användarens lokala SQLite-databas. Två stycken tabeller skapas:

- lists för alla listor att sparas i med kolumnerna: `_id` (PK) och `listTitle`
- items för alla artiklar att sparas i men kolumnerna: `_id` (PK), `itemTitle`, `description`, `booked` samt `parent`.

ListsDbAdapter

Utökar `AbstractListigtDbAdapter`. Pratar med tabellen `lists`. För mer information se JavaDoc.

ItemsDbAdapter

Utökar `AbstractListigtDbAdapter`. Pratar med tabellen `items`. För mer information se JavaDoc.

ListEditor

Om användaren vill editera en lista kommer titeln att skickas från MainListActivity via ett Intent. Här kan användaren editera titeln för den specifika listan. Innehåller även en `View.OnClickListener()` som lyssnar ifall användaren har mer än en bokstav i titeln och låter denne klicka på Confirm i sådana fall.

Layouten finns i `list_editcreate.xml`.

ItemEditor

Fungerar i stort sett som ListEditor, men denna aktivitet får även en artikels description via ett Intent. Användaren får således ytterligare ett fält som kan editeras.

Layouten finns i `item_editcreate.xml`.

ItemDetailsActivity

Om användaren klickar på en artikel aktiveras denna aktivitet. Använder onCreate() samt fillData() ungefär på samma sätt som MainListActivity samt ItemListActivity, men enbart en artikels information hämtas från databasen via en instans av ItemsDbAdapter. Layouten finns i item_details.xml

ShareListActivity

Denna aktivitet visar fält där användaren skriver in e-post adresser och skickar iväg informationen från listor de vill dela till dessa e-postadresser. Mycket av denna funktionalitet är inte egenutvecklad, utan en klass Mail.java används. Alla titlar från artiklarna i den listan användaren vill skicka iväg extraheras ur databasen m.h.a en Cursor i metoden getItems().