

Android 5.0

peter.ekler@aut.bme.hu



Automatizálási és
Alkalmazott
Informatikai Tanszék



AutSoft
BME

Android 5.0

It is going to be ...

...wait for it...



Lollipop (Legendary 😊)

Android 5.0

- Új készülékek (telefon, tablet, óra, autó, ...)
- Új futtatókörnyezet
 - > Android Runtime (ART)
- Megújult UI design
 - > material design
- Fejlett értesítések
- Energiahatékonyság növelése
 - > Project Volta
- További új API-k 😊



Android, nem csak telefonon...

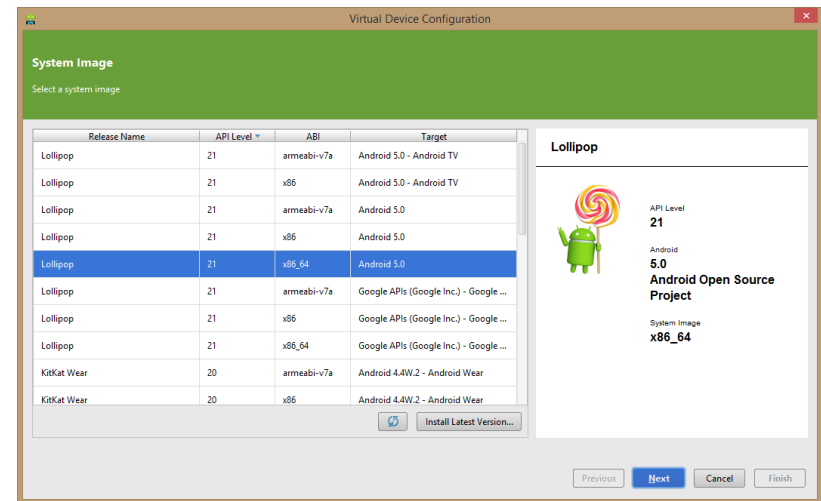
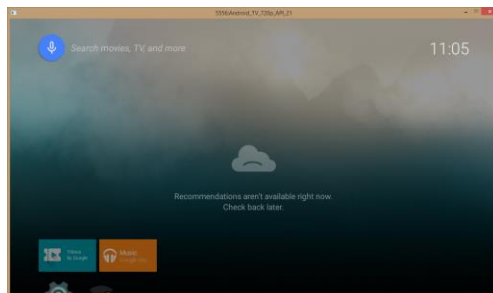


Android Runtime (ART)

- Default futtatókörnyezet Android 5-ben
 - > 4.4-ben még csak választható volt
- Legfőbb funkciók:
 - > Ahead-of-Time (AOT) fordítás
 - > Fejlettebb GC
 - > Fejlettebb debugging támogatás

Fejlesztői eszközök - aktualitások

- Megújult AVD Manager
- Hardveres gyorsítás Google Maps esetén is! 😊
- 64 bites hardveres image
- Android Studio folyamatos fejlődése
- Okosóra emulátor
- Android TV emulátor



Multimédia API és adattárolás

- Új API: *android.hardware.camera2*
 - > Testreszabható fénykép készítés
 - > *onCaptureCompleted(...)* ... 😊
- Audio kezelés
 - > Floating-point formátum
 - > Audio adat *ByteBuffer*-ben
- Kiterjesztett Storage Access Framework
 - > *DocumentsProvider*



Hálózati kommunikáció

- Multi-networking API
 - > Alkalmazás kiválaszthatja a számára megfelelő hálózatot (*requestNetwork()*, *NetworkCallbackListener*)
- Bluetooth Low Energy
 - > A készülék már Bluetooth LE periféria eszközként is tud működni
 - > Broadcasting támogatása
- Továbbfejlesztett NFC
 - > Android Beam a megosztás menüben
 - > *invokeBeam()*



Energiahatékonyság támogatása

- *JobScheduler* funkció:
 - > Aszinkron feladatok futtatása megfelelő állapotban (ha töltőn van, ha WiFi hálózaton van, ütemezetten, stb.)
- Példa:

```
JobInfo uploadTask = new JobInfo.Builder(  
    mJobId, mServiceComponent) .  
    setRequiredNetworkCapabilities(  
        JobInfo.NetworkType.UNMETERED) .build();
```

```
JobScheduler jobScheduler = (JobScheduler)  
    context.getSystemService(  
        Context.JOB_SCHEDULER_SERVICE);  
jobScheduler.schedule(uploadTask);
```



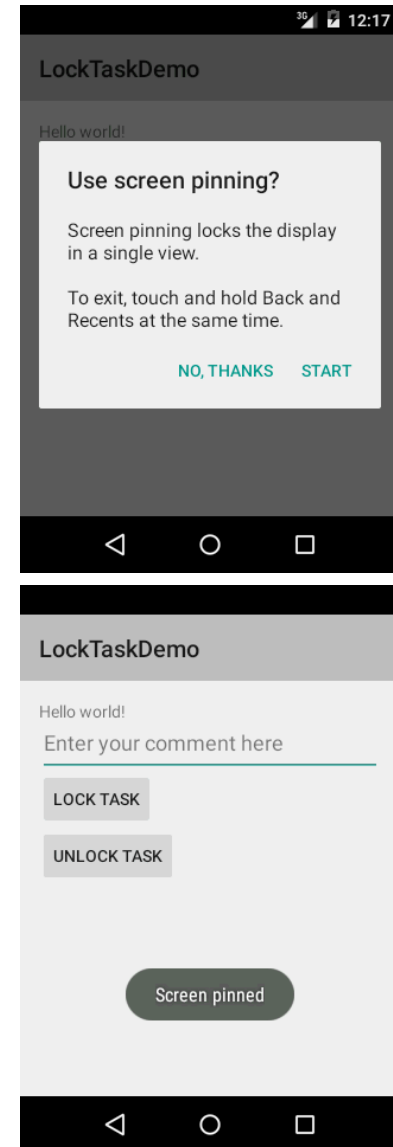
Fejlesztői eszközök energiahatékonyságra

- Statisztika készítés akkumlátor használatról:
 - > `dumpsys batterystats`
- Akkumlátorral kapcsolatos események
- Globális statisztikák
- Közelítő energia használat UID és rendszer komponensek alapján System UID alapú aggregált statisztikák
- App UID alapú aggregált statisztikák



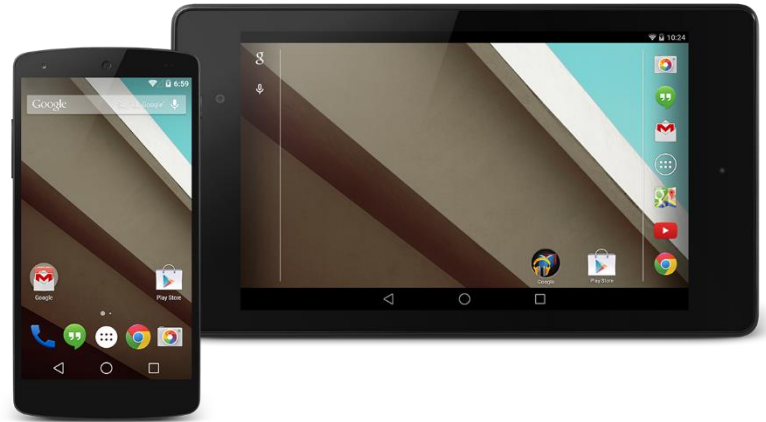
Task zárolás

- Task locking API
- Pl.: oktató alkalmazásokhoz
- Megfelelő engedély/beállítás szükséges (device owner app)
- Érintett viselkedés:
 - > Üres statusbar és értesítések tiltása
 - > Home és korábbi alkalmazások gomb elrejtése
 - > Más alkalmazás nem indíthat Activity-t
 - > Az aktuális alkalmazás indíthat új Activity-t, de csak ebben a Task-ban



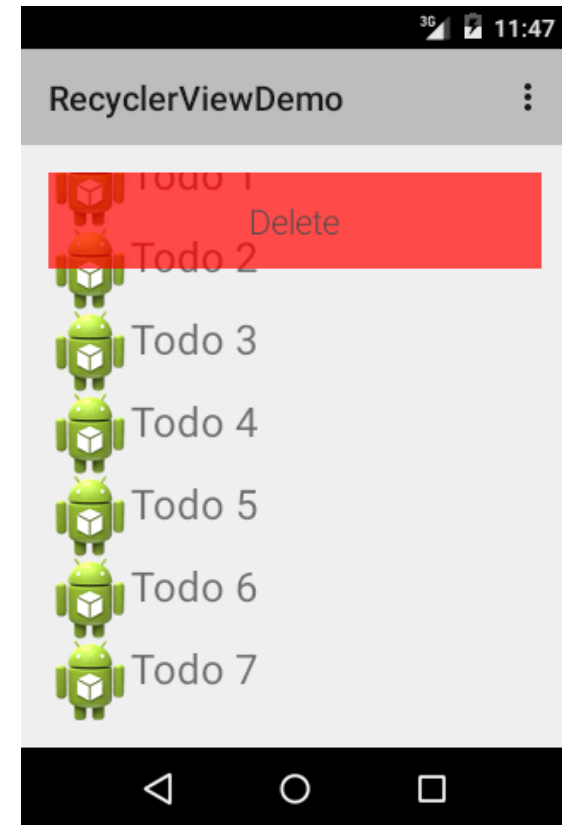
Material design

- Material téma
- View árnyékok
- *RecyclerView* és *CardView*
- *Drawable* animációk és stílusok
- Material design animációk és activity transition hatások
- Animatorok
- Továbbfejlesztett UI widgetek



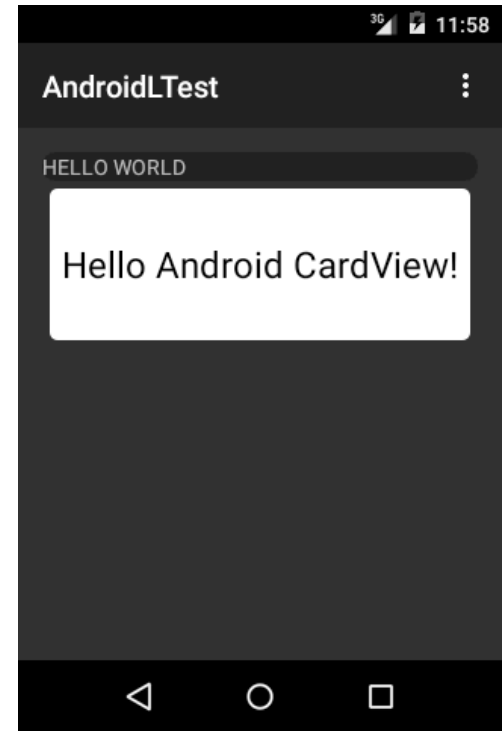
RecyclerView

- A *ListView* fejlett és flexibilis változata
- *ViewHolder* minta kikényszerítése
- Hatékony elem újrafelhasználás
- Fejlesztés:
 - > *ListView*, és *Adapter* mellett egy *LayoutManager*-t is kell készíteni
- A *LayoutManager* feladata a *findViewById(...)* felesleges sokszori hívása
- *Gradle* függőség:
 - > compile
'com.android.support:recyclerview-v7:+'



CardView

- Információk konzisztens megjelenítését támogatja kártyák formájában
- *FrameLayout* leszármazott
- „Kiemelkedés” támogatása
 - > *android:elevation*
- *További attribútumok*
 - > *card_view:cardCornerRadius*
 - > *card_view:cardBackgroundColor*
- *Gradle függőség:*
 - > *compile 'com.android.support:cardview-v7:+'*



Hivatalos változások

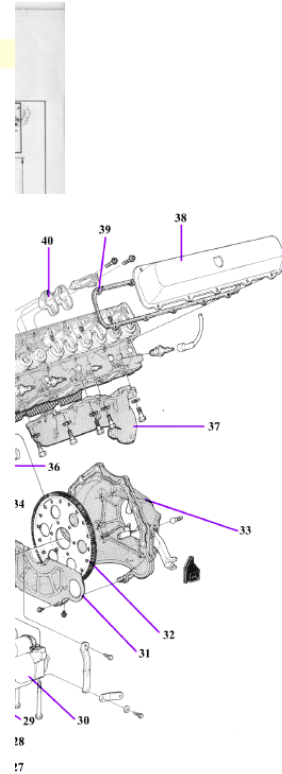
- https://developer.android.com/sdk/api_diff/21/changes.html
- Néhány új csomag/osztály:
 - > *TelecomManager*:
 - *Boolean isInCall()*
 - *showInCallScreen(boolean showDialpad)*
 - > *android.hardware.camera2*
 - > *UsageStatsManager*
 - > *android.media.tv*
 - > *android.system.Os*
 - > *android.media.projection*

Mi a mérnök feladata?

```
* upload, independent of whether the original activity is paused, stopped,  
or finished.
```

```
*/
```

```
public class Activity extends ContextThemeWrapper  
    implements LayoutInflater.Factory2,  
        Window.Callback, KeyEvent.Callback,  
        OnCreateContextMenuListener, ComponentCallbacks2,  
        Window.OnWindowDismissedCallback {  
    private static final String TAG = "Activity";  
    private static final boolean DEBUG_LIFECYCLE = false;  
  
    /** Standard activity result: operation canceled. */  
    public static final int RESULT_CANCELED = 0;  
    /** Standard activity result: operation succeeded. */  
    public static final int RESULT_OK = -1;  
    /** Start of user-defined activity results. */  
    public static final int RESULT_FIRST_USER = 1;  
  
    static final String FRAGMENTS_TAG = "android:fragments";  
  
    private static final String WINDOW_HIERARCHY_TAG = "android:viewHierarchyState";  
    private static final String SAVED_DIALOG_IDS_KEY = "android:savedDialogIds";  
    private static final String SAVED_DIALOGS_TAG = "android:savedDialogs";  
    private static final String SAVED_DIALOG_KEY_PREFIX = "android:dialog_";  
    private static final String SAVED_DIALOG_ARGS_KEY_PREFIX = "android:dialog_args_";  
  
    private static class ManagedDialog {  
        Dialog mDialog;  
        Bundle mArgs;  
    }  
    private SparseArray<ManagedDialog> mManagedDialogs;  
  
    // set by the thread after the constructor and before onCreate(Bundle savedInstanceState) is called.  
    private Instrumentation mInstrumentation;  
    private IBinder mToken;  
    private int mIdent;  
    /*package*/ String mEmbeddedID;  
    private Application mApplication;  
    /*package*/ Intent mIntent;
```



Android fejlesztés

- Hibamentes, hatékony működés
- Megfelelő library-k ismerete és használata
- Fejlesztőkörnyezet kialakítása
 - > Verziókezelés
 - > Continuous Integration
 - > Tesztelés
 - Unit tesztek
 - Integrációs tesztek
 - Teszt környezet
- Clean code
 - > Kódminőség
 - > Kód újrafelhasználhatóság
 - > SOLID tervezési elvek
 - > Refaktor
- Optimalizálás
 - > Memória, CPU használat és ... energiafogyasztás
- Test Driven Development



Köszönöm a figyelmet!



peter.ekler@aut.bme.hu