Android Programmierung: Andre Ippisch, Raphael Bialon

Vibora Feed - Die App zur Webseite

Jochen Peters

Heinrich-Heine-Universität Düsseldorf
Informatik MA - SS 2016

August 2016

Inhalt

ViboraFeed ?!

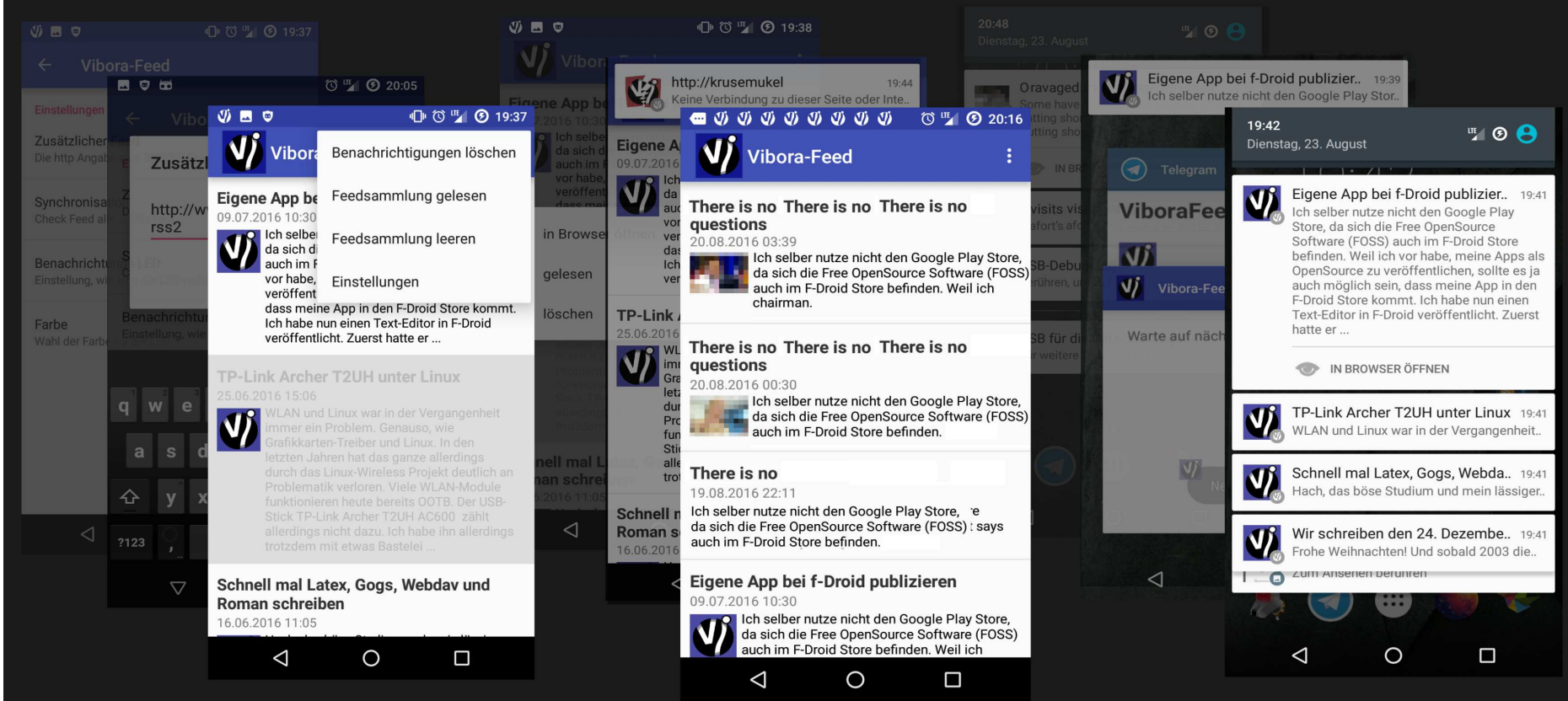
ViboraFeed ?!

ViboraFeed ?!

- RSS Reader für vibora.de
- Webseite populär machen (f-Droid Shop)
- Zusätzlichen Feed durch User anpassbar (NEU)
- keine Liste von Listen von Vorschau
- ganzer Feed-Text sofort sichtbar

Vibora-Feed

Die Schlange durch die OpenSource Welt *als App für Android*



Funktionen

Funktionen

Funktionen 1/4

- Da Feeds nur kurze Texte sind:
  - komplett in Notifikation
  - komplett in Listenzeile
  - **keine** Aggregation
- Vorschaubild Extraktion

Funktionen 2/4

- Gelesen-Markierung
- GUI: `de` und `en`
- Öffnen des Links über Notifikation
- Nachtmodus (18:00 - 6:59)

Funktionen 3/4

- Modified Auswertung
- Retry bei Verbindungsabbruch
- Datenbank als Cache
- Müll leeren
- Alarm Start beim Booten
- Kontext- und Optionsmenü

Funktionen 4/4

Einstellungen

- Notifikationsfarbe + Blinken
- Refresh Intervall
- eigenen Zusatzfeed (NEU)
- Umschalten auf Zusatzfeed (NEU)

Aufbau und Code

Application: ViboraApp

Aufbau und Code

Application: ViboraApp

- Config / App Konstanten
- Async Task für Preferences
- Strict Mode
- onCreate: new BroadcastReceiver **Alarm**

Aufbau und Code

MainActivity

Aufbau und Code

MainActivity

- Optionsmenu
- Nachtmodus
- Wertet Intents von Alarm aus (alarmReceiver)
- onPause/Resume: setzt *withGui*
- App wirklich verlassen?

Aufbau und Code - MainActivity

onCreateOptionsMenu(Menu menu)

```
MenuInflater inflater = getMenuInflater();
inflater.inflate(R.menu.mainmenu, menu);
super.onCreateOptionsMenu(menu);
return true;
```

Aufbau und Code - MainActivity

onOptionsItemSelected(MenuItem item)

```
DbClear dbClear = new DbClear();
switch (item.getItemId()) {
  case R.id.action_preferences:
    ...
```

Aufbau und Code - MainActivity

Nachtmodus

Aufbau und Code - MainActivity

```
UiModeManager umm =
  (UiModeManager)
     getSystemService(Context.UI_MODE_SERVICE);
int hourOfDay =
  Calendar.getInstance().get(Calendar.HOUR_OF_DAY);
```

Aufbau und Code - MainActivity

```
if (hourOfDay > 6 && hourOfDay < 18)
  umm.setNightMode(UiModeManager.MODE_NIGHT_NO);
else
  umm.setNightMode(UiModeManager.MODE_NIGHT_YES);
```

Aufbau und Code - MainActivity

App wirklich verlassen?

Aufbau und Code - MainActivity

```
@Override
public void onBackPressed() {
  QuitDialogFragment dialog =
    new QuitDialogFragment();
  dialog.show(
    getSupportFragmentManager(), "Dialog"
  );
}

public void onUserExit() {
  super.onBackPressed();
}
```

Aufbau und Code

QuitDialogFragment

Aufbau und Code - QuitDialogFragment

QuitDialogFragment *extends* DialogFragment

```
public Dialog onCreateDialog(Bundle s) {
  Builder b = new
     AlertDialog.Builder(getActivity());
  LayoutInflater i =
     getActivity().getLayoutInflater();
  b.setView(i.inflate(R.layout.quit_dialog, null))
    .setPositiveButton("Ja",    JACODE)
    .setNegativeButton("Nein", NOCODE);
  return builder.create();
}
```

Aufbau und Code - QuitDialogFragment

JACODE

```
new DialogInterface.OnClickListener() {
   public void onClick(...) {
      ((MainActivity) getActivity()).onUserExit();
      dialog.dismiss();
   }
}
```

Aufbau und Code - QuitDialogFragment

NOCODE

```
new DialogInterface.OnClickListener() {
   public void onClick(...) {
      QuitDialogFragment.this.getDialog().cancel();
   }
}
```

Aufbau und Code

FeedListFragment

Aufbau und Code

FeedListFragment

- Context Menu
- Liste der Feeds aus DB
- LoaderManager für Cursor aus DB
- Nutz URI des FeedContentProvider
- Macht CRUD via getContentResolver() der Activity

Aufbau und Code - FeedListFragment

- onCreateView()
  - adapter ist `FeedCursorAdapter`;
  - setListAdapter(adapter);

-> implements `LoaderManager.LoaderCallbacks<Cursor>`

Aufbau und Code - FeedListFragment

- `onCreateLoader()`
  - ist Verbindung zur URI / DB
  - erzeugt CursorLoader
- `onLoadFinished()`
- `onLoaderReset()`

Die letzten beiden geben Cursor an den `FeedCursorAdapter` weiter.

Aufbau und Code

FeedCursorAdapter

Aufbau und Code

FeedCursorAdapter

- `newView()`: ein Recycling Alter Views
- `bindView()`: Daten des Cursors in View füllen

Aufbau und Code

FeedContentProvider

Aufbau und Code

FeedContentProvider

- CRUD zur Datenbank via URI
- `_database` liefert der `FeedHelper`
  - name
  - version
  - löschen
  - anlegen

Aufbau und Code - FeedContentProvider

Wichtig:

- URI Matcher (FEED oder FEEDS)
- `SQLiteDatabase sqlDB = _database.getWritableDatabase();`
- `getContext().getContentResolver().notifyChange(uri, null);`

Aufbau und Code

FeedContract

Aufbau und Code

FeedContract

- liefert Konstanten und Queries
- Enthält viele Reformatierungs Methoden

Aufbau und Code

DeviceBootReceiver

Aufbau und Code

DeviceBootReceiver - reagiert auf “fertiges Booten”

```
public void onReceive(Context c, Intent i) {
  if (ViboraApp.alarm == null)
    ViboraApp.alarm = new Alarm();
  ViboraApp.alarm.start(c);
}
```

Aufbau und Code - DeviceBootReceiver

AndroidManifest.xml

```
<manifest android:installLocation="internalOnly"
    ...
```

<receiver> mit <intent-filter>:

```
<action
 name="android.intent.action.BOOT_COMPLETED"
/>
```

Aufbau und Code

Refresher und Alarm

Aufbau und Code

Refresher und Alarm

- War vorher im Prinzip eine **dicke** Funktion/Methode
- Splittung in 2 Klassen nur Kosmetik
- Refresher
  - viele XML Methoden sind nun in `FeedContract`
  - Splittung sollte Unit-Tests erleichtern

Aufbau und Code

Alarm:

- Koordiniert Methoden von Refresher
- prüft Ergebnisse

Refresher:

- URL Zugriffe
- Notifikationen

Aufbau und Code

Notification Minimal

Aufbau und Code - Notification Minimal

```
NotificationCompat.Builder mBuilder =
    new NotificationCompat.Builder(context);
mBuilder.setContentTitle(title)
    .setContentText(body)
    .setTicker(body)
    .setContentIntent(pi)
    .setSmallIcon(R.drawable.ic_launcher);
```

Aufbau und Code - Notification Minimal

pi = PendingIntent:

```
Intent in = new Intent(this, MainActivity.class);
in.setFlags(
  Intent.FLAG_ACTIVITY_CLEAR_TOP |
  Intent.FLAG_ACTIVITY_SINGLE_TOP
);
PendingIntent pi =
  PendingIntent.getActivity(this, 0, in, 0);
```

Aufbau und Code - Notification Minimal

Actionen:

```
Intent linkIntent = new Intent(
  Intent.ACTION_VIEW, Uri.parse("http://x3.de"));
PendingIntent linkpi = PendingIntent.getActivity(
  this, 0, linkIntent, 0);
mBuilder.addAction(
  drawableIcon, "open Link", linkpi);
```

Aufbau und Code - Notification Minimal

```
Notification noti = mBuilder.build();
noti.flags |= Notification.FLAG_AUTO_CANCEL;
NotificationManager nm =
  (NotificationManager)
     getSystemService(NOTIFICATION_SERVICE);
nm.notify(longId, noti);
```

Unit-Tests

Unit-Tests

Unit-Tests

- User kann (eingentlich keine) Eingaben machen !
- Ist Vibora.de Feed erreichbar?
- Verhalten sich diverse XML Extraktionen korrekt?

Unit-Tests Beispiel 1/2

Ist Vibora.de Feed erreichbar?

Unit-Tests Beispiel 1/2

```
@Test
public void testFeedUrl() throws Exception {
  String feedUrl =
     ViboraApp.Config.DEFAULT_rssurl;
  URL url = new URL(feedUrl);
  HttpURLConnection conn =
    (HttpURLConnection) url.openConnection();
  assertNotNull("open connection", conn);
```

Unit-Tests Beispiel 1/2

```
    int response = conn.getResponseCode();
    assertTrue(
      "response is not 200 or 304 ("+response+")",
      response == 200 || response == 304);
    conn.getInputStream().close();
}
```

Unit-Tests Beispiel 2/2

Datums Extraktion?

Unit-Tests Beispiel 2/2

```
String testDateStr =
  "Sat, 09 Jul 2016 08:30:04 +0000";

@Test
public void testRawToDate() throws Exception {
  Date date = FeedContract.rawToDate(testDateStr);
  assertNotNull(date);
  assertNotEquals(date.getTime(), 0);
  assertTrue(date.before(new Date()));
}
```

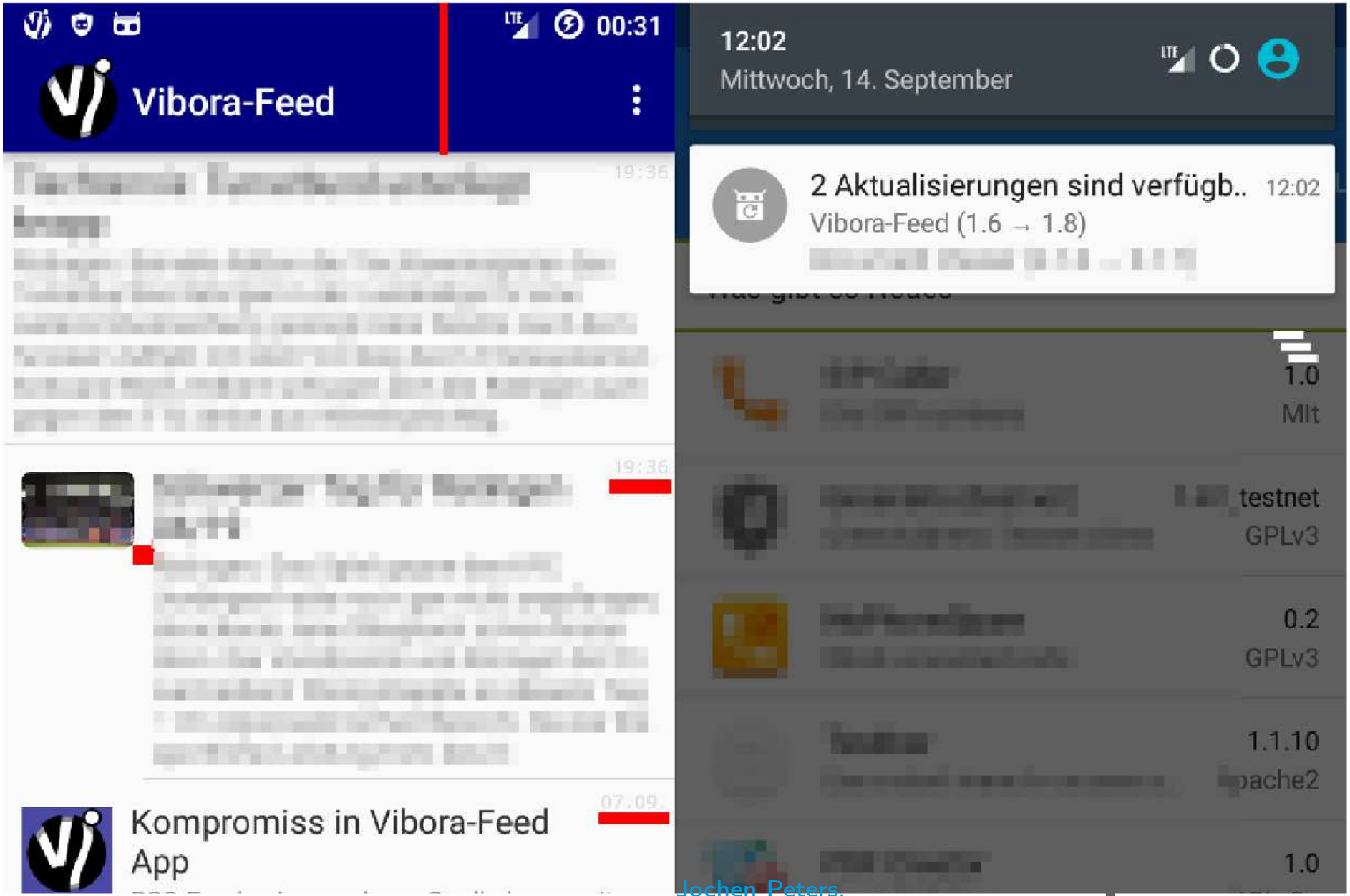
Ausblick

Wie ging oder geht es weiter?

Ausblick


ViboraFeed
Version 1.6
Benachrichtigungen löschen
Feedsammlung gelesen
Feedsammlung leeren
Einstellungen
Zusätzlicher Feed
ViboraFeed – unsere Android App ist da
Eigene App bei f-Droid publizieren

Ausblick


00:31
Vibora-Feed
19:36
19:36
Kompromiss in Vibora-Feed
App
07.09.
12:02
Mittwoch, 14. September
2 Aktualisierungen sind verfügb..
12:02
Vibora-Feed (1.6 → 1.8)
1.0
testnet
GPLv3
0.2
GPLv3
1.1.10
1.0

Ausblick

- dynamischer Nachtmodus via Helligkeitssensor oder Zeitangabe
- Querformat: Webseitenvorschau
- Vibora.de: Feedbilder
- Auswahl populärer Zusatzfeeds
- Hinweis, wenn URL fehlerhaft
- Anti-Sport Filter / Blacklist
- Suche in Actionbar

Danke

Fragen?