

Softwaretechnik I Projekt



Aiman Al-Alie
Hamdan Al-Arashi
Max Jacobsen-Mann
Naresh Kumar Metha
Bibek Upreti
Sashi Bhusan Yadav

Finanz App

Probleme, Schwierigkeiten und Lösungen	2
Einnahmen Hinzufügen	2
UI Entwurf	2
Einträge anzeigen und die Detailansicht	3
Statistik	3
Datenbank	4
Reflektion	4
Gruppen Protokoll	5
KW 49	5
KW 50	6
KW 51	7
KW 52	8
KW 1	9
KW 2	10
Quellen	10

Probleme, Schwierigkeiten und Lösungen

Einnahmen Hinzufügen

Am anfang haben wir uns überlegt, das Datum als Text Input Feld zu implementieren, da wir am Anfang noch keinen Kalender hatten, jedoch war es schwierig, den Input auf die Limitierungen vom Datum zu begrenzen, z. B. konnte man "120.02.12345" schreiben, was ja kein Datum ist. Deswegen haben wir das Text Input Feld zu einem Kalender umgewandelt, welcher den aktuellen Monat angezeigt und man dann dort den bestimmten Tag raussuchen kann. Außerdem konnte man bei dem Namen Input Feld unendlich viele Zeichen eingeben, weswegen wir die Eingabe auf 64 Zeichen begrenzt haben und wir haben noch eine Überprüfung gemacht, um zu prüfen, ob der Name weniger als 2 Zeichen hat, da diese nicht erlaubt sind. Bei der Währung haben wir die Währung mit dem Text Input Feld eingeben, jedoch hatte dies das Problem, dass das Zeichen alles sein konnte, auch keine Währung. Aus diesem Grund haben wir das Text Input Feld durch ein Dropdownmenü getauscht, wo man die voreingestellten Währungen auswählen kann.

Im Laufe der Bearbeitung ist mir aufgefallen, dass man ein paar Abfragen weglassen kann, da diese in der XML Datei einfach hinzugefügt werden können, deswegen habe ich die überflüssigen Abfragen aus dem Code genommen.

UI Entwurf

Am Anfang habe ich alle einzelnen Anwendungsfälle mit separaten Aktivitäten erstellt, dies hat ganz gut funktioniert, jedoch hatte ich schon beim Testen im Emulator gewisse Performance Probleme, zum Beispiel längere Wartezeiten beim Öffnen von mehreren Aktivitäten hintereinander. Während meiner Arbeit mit den Aktivitäten bin ich über die Fragmente gestolpert, zu diesem Zeitpunkt hatte ich jedoch keine Ahnung, wie ich diese dann in die ganzen Aktivitäten einbringe, aus diesem Grund habe ich ein neues Projekt erstellt und von Anfang an alles mit Fragmenten implementiert. Hierbei ist mir ein Problem aufgefallen, und zwar dass ich gewisse Sachen nicht richtig benutzen kann, weil alles auf der Hauptaktivität läuft und von da aus gestartet und verarbeitet wird. Währenddessen ist mir aufgefallen, wie einfach zu implementieren die Aktivitäten waren und dass die Fragmente schon komplexer aufgebaut sind. Bei den Einträgen anzeigen Menü hatte ich Schwierigkeiten, ein Fragment über ein anderes Fragment zu legen, ohne den normalen Ablauf zu beeinflussen.

Einträge anzeigen und die Detailansicht

Bei Einträge anzeigen habe ich mich für die Listview entschieden, um die Einträge aus der Datenbank anzeigen zu lassen, dafür musste ich eine xml-Datei (Eintrag_Cell.xml) für einen Eintrag definieren, wie dieser dann in der Liste angezeigt wird. Als Nächstes musste ich eine Adapter Klasse schreiben, welche die Einträge und deren Daten auf die in der xml-Datei definierten Textviews setzte. Nach dem ersten mal Testen ist mir aufgefallen, dass das Datum nicht das gewünschte Format hat und musste dieses im Anschluss anpassen, indem ich ein eigenes Format erstellt habe und das Datum vom Eintrag mit dem neuen Format formatiert habe. Außerdem habe ich die getItem Funktion überschrieben, da diese dann den Eintrag aus der Liste wiedergibt an gegebener Position. Nach mehreren Testläufen ist mir dann aufgefallen, dass die Listeneinträge sich der Einträge anzeigen Fragmentes immer verdoppeln, da ich nicht so sehr mit Listview und Android Studio vertraut bin, habe ich leider keinen Lösungsansatz für dieses Problem. Nach der Präsentation im Praktikum haben wir den Kritikpunkt, dass die Einträge mehrfach auftauchen aufgenommen und haben diesen anschließend gelöst bekommen, in dem wir die Struktur des SQL-Statements etwas angepasst haben und in dem Fragment immer eine neue Liste erstellen, welche dann mit den Einträgen gefüllt wird.

Dadurch wurde der Fehler behoben, dass sich die Einträge immer vervielfältigt haben. Anschließend habe ich die Suche mit dem "Von Datum" und "Bis Datum" implementiert. Am Anfang hatte ich noch ein paar Probleme mit dem Kalender, da dieser nur 1.1.2023 wiedergegeben hat und nicht 01.01.2023, wie es auch in der Datenbank steht, dies habe ich durch eine Hilfsmethode gelöst bekommen. Im weiteren Verlauf ist mir aufgefallen, dass das Datumsformat mit dd. MM.yyyy ein paar Probleme bereitet und zwar vergleicht die Datenbank scheinbar nur den Tag und mehr nicht.

```
select * from eintrag where datum >= '01.01.2023' and datum <= '31.01.2023'
```

Wenn ich diesen Query in der Datenbank ausführe, liefert er mir alles richtig mit den Tagen, jedoch gibt er mir auch alle anderen Einträge, welche nicht im Januar und auch nicht in 2023 sind. Wir haben das Datums Format geändert bekommen, wodurch die Date SQL Abfragen alle funktioniert haben. Bei der Detailansicht hatte ich am Anfang das Problem, dass ich keine Parameterwerte dem Fragment mitgeben konnte, um diese dann in der Detailansicht zu verwenden, dies habe ich mit einer Methode newInstance gelöst, welche die Werte aus dem Listenelement übergibt. Im nächsten Schritt habe ich die übergebenen Parameter den dazugehörigen Textviews zugeordnet.

Statistik

Am Anfang haben wir für die Piechart Einträge vorgefertigte Einträge benutzt, dies haben wir jetzt mit der Datenbank verbunden und eine Ausgaben-Tabelle hinzugefügt. Danach haben wir in dem Statistik Fragment, die von und bis Daten integriert, damit diese die Werte für das Piechart verändern.

Datenbank

Hier haben wir uns erst mal an JDBC gehalten und uns überlegt, ob wir eventuell später auf ein ORM basiertes Datenbank System wechseln. Die Datenbank hat zwei Tabelle (Eintrag und Ausgabe), welche beide die ID, den Namen, den Betrag, das Datum, die Kategorie, die Währung enthält. Nachdem man die Semantik von den Querys öfter benutzt hat, konnten wir die SQL Statements mit einem RawQuery lösen.

```
return DB.rawQuery( sql: " Select * from Eintrag", selectionArgs: null)
```

Reflektion

Das Aktivitätsdiagramm war sehr hilfreich, da man dort gut nachvollziehen konnte, wie das Programm dann aufgebaut werden sollte. Das UML-Diagramm war sehr hilfreich, da man schon mal die Grundstruktur von den gegebenen Klassen hatte und man sich anhand dessen vortasten konnte. Z. B. kann man dann alles in einem Eintrag Objekt speichern und kann dieses übergeben und nicht die einzelnen Attribute. Der UI-Entwurf am hilfreichsten, da dieser die UI-Implementation vereinfacht hat, da man genau wusste, wo man welches Objekt/Element platzieren muss und nicht erst selber überlegen muss, wo man jetzt das bestimmte Objekt platzieren würde. Jedoch war es ab und zu nicht eindeutig, wie das dargestellt, dann funktionieren soll (siehe Hamburger Menü aufgeklappt), bei dem Entwurf war es leider nicht eindeutig, ob alle Untermenüs aufgeklappt werden sollen oder nur eines, dies kann man aber dann auch selbst scheiden, welche variant für einen besser liegt dem kommt hinzu, das manche vorgaben schwer ein zu halten waren, da diese Entscheidungen problematisch zum Umsetzen waren.

Gruppen Protokoll

KW 49

Name	Anwesend	Was wurde gemacht?	%-Anteil
Aiman Al-Alie	Ja	Basic UI entwurf diskutiert + Aufgabenverteilung	5
Hamdan Al-Arashi	Ja	Basic UI entwurf diskutiert + Aufgabenverteilung	5
Max Jacobsen-Mann	Ja	Basic UI entwurf implementiert	75
Naresh Kumar Mehta	Ja	Basic UI entwurf diskutiert + Aufgabenverteilung	5
Bibek Upreti	Ja	Basic UI entwurf diskutiert + Aufgabenverteilung	5
Sashi Bhusan Yadav	Ja	Basic UI entwurf diskutiert + Aufgabenverteilung	5

KW 50

Name	Anwesend	Was wurde gemacht?	%-Anteil
Aiman Al-Alie	Ja	Einträge hinzufügen + Fehlermeldungen+Eingabe begrenzungen, Besprechung	20
Hamdan Al-Arashi	Ja	Einträge hinzufügen + Fehlermeldungen + Eingabe begrenzungen, Besprechung	20
Max Jacobsen-Mann	Ja	Das Projekt mit der Datenbank verbunden und Einträge hinzufügen kompatibel für die Datenbank eingabe gemacht, Besprechung	40
Naresh Kumar Mehta	Ja	Besprechung	6
Bibek Upreti	Ja	Besprechung	6
Sashi Bhusan Yadav	Ja	Besprechung	6

KW 51

Name	Anwesend	Was wurde gemacht?	%-Anteil
Aiman Al-Alie	Ja	Kalender View verbinden + währung als Dropdown	25
Hamdan Al-Arashi	Ja	Kalender View verbinden + währung als Dropdown	25
Max Jacobsen-Mann	Ja	Die Activitys durch Fragmente ersetzt und die vorherigen funktion auf die Fragmente angepasst	50
Naresh Kumar Mehta	Nein		
Bibek Upreti	Nein		
Sashi Bhusan Yadav	Nein		

KW 52

Name	Anwesend	Was wurde gemacht?	%-Anteil
Aiman Al-Alie	Ja	Einträge anzeigen von bis mit Kalender verbunden, Besprechung	21
Hamdan Al-Arashi	Ja	Einträge anzeigen von bis mit Kalender View verbunden, Besprechung	21
Max Jacobsen-Mann	Ja	Die UML Modelle hinzugefügt, Besprechung	25
Naresh Kumar Mehta	Ja	Besprechung, Statistik hinzugefügt	11
Bibek Upret	Ja	Besprechung, Statistik hinzugefügt	11
Sashi Bhusan Yadav	Ja	Besprechung, Statistik hinzugefügt	11

KW 1

Name	Anwesen d	Was wurde gemacht?	%-Anteil
Aiman Al-Alie	Ja	versucht Einträge aus der DB anzuzeigen, die nur mit Bestimmten Datum "Von Bis"	15
Hamdan Al-Arashi	Ja	versucht Einträge aus der DB anzuzeigen, die nur mit Bestimmten Datum "Von Bis"	15
Max Jacobsen-Mann	Ja	Einträge anzeigen, den adapter für die Listview zum Einträge anzeigen und die Detail ansicht, wenn man auf ein Listenelement drauf klickt	40
Naresh Kumar Mehta	Ja	Statistik hinzugefügt mit Kalender verbunden	10
Bibek Upreti	Ja	Statistik hinzugefügt mit Kalender verbunden	10
Sashi Bhusan Yadav	Ja	Statistik hinzugefügt mit Kalender verbunden	10

KW 2

Name	Anwesend	Was wurde gemacht?	%-Anteil
Aiman Al-Alie	Ja	Powerpoint bearbeitet und Bilder hinzufügt, Einnahmen hinzufügen.	16,6
Hamdan Al-Arashi	Ja	Powerpoint bearbeitet und Bilder hinzufügt Einnahmen hinzufügen.	16,6
Max Jacobsen-Mann	Ja	Einträge anzeigen menü, Powerpoint angefangen, Einträge anzeigen duplikate gelöst und Datums suche, Detailansicht Hints bearbeitet, Statistiken bearbeitet mit Datenbank verbunden	16,6
Naresh Kumar Mehta	Ja	Powerpoint bearbeitet, Statistiken bearbeitet mit Datenbank verbunden, Ausgaben hinzufügen in Datenbank eingebunden	16,6
Bibek Upreti	Ja	Powerpoint bearbeitet, Statistiken bearbeitet mit Datenbank verbunden, Ausgaben hinzufügen in Datenbank eingebunden	16,6
Sashi Bhusan Yadav	Ja	Powerpoint bearbeitet, Statistiken bearbeitet mit Datenbank verbunden, Ausgaben hinzufügen in Datenbank eingebunden	16,6

Quellen

Titelbild = <https://hub.packtpub.com/how-create-breakout-game-godot-engine-part-1/>