

Name	Angelique Freier
Titel der App	Cash Cow & Saver Fox
GitHub Repository URL	

Zielsetzung

Ziel der Abschlussaufgabe ist es, das Gelernte innerhalb des Swift-Moduls in einer eigenen App umzusetzen und diese App abschließend in einer Präsentation vorstellen und erklären zu können. Diese App soll anschließend als Teil eures Portfolios auf GitHub veröffentlicht werden.

Anforderungen an die App

Mindestens 5 Screens

Deine App muss mindestens 5 verschiedene Screens anzeigen. Jeder Screen soll dabei sinnvoll genutzt werden.

Navigation

Zur grundlegenden Navigation kann ein `TabView` genutzt werden (muss aber nicht). Wichtig ist, dass mindestens **1 Sheet** und **1 NavigationLink** genutzt werden.

Anzeigen von Daten innerhalb einer Liste

In einem deiner Views muss eine Liste angezeigt werden, welche sinnvoll integriert ist und nützliche Inhalte liefert. Optional können optional auch Swipe Actions integriert werden.

Abspeichern von Daten (Core Data / Firebase)

Daten, die von deinen Usern generiert werden, sollen entweder zur Offline-Nutzung mit **Core Data** zur Verfügung stehen oder du integrierst **Firebase**, um die Daten zu synchronisieren. Wenn du dich für Firebase entscheidest, nutze den Firestore zum Speichern und Authentication, um User zu authentifizieren.

API Call

Such dir eine API, welche thematisch relevanten Inhalt für deine App bereitstellt und nutze die API, indem du einen API Call integrierst. Die API kann mit oder ohne API-Key sein.

MVVM

Zur Integration der bisher gelernten Softwarearchitektur muss auch deine App MVVM nutzen. Optional kannst du hier direkt mit Repositories arbeiten.

Pitch

Schreibe in diesem Feld 3-5 Sätze, die beschreiben, welche Funktionalitäten deine App bieten wird und welchen Nutzen sie erfüllen soll. Stell dir vor, du beschreibst die App einem/einer Freund:in.

Die eine App ist für Erwachsene und soll einen Überblick über ihre Finanzen schaffen. Ein Erwachsener kann in seinem Profil Unterkonten für Kinder anlegen.

Die App für Kinder soll ihnen etwas Helfen im Umgang mit ihrem Geld, in dem sie ihre Einnahmen und Ausgaben festhalten und ihnen aufgezeigt wird wieviel sie noch zur Verfügung haben. Die Kinder können Sparziele anlegen und werden daran erinnert nicht zu viel auszugeben um ihr Ziel erreichen zu können. Für den gewissenhaften Umgang mit ihrem Geld bekommen sie Punkte und können sich damit kleine Spiele freischalten.

Aufschlüsselung der Anforderungen

Schreibe hier in jedes Feld (in Stichpunkten), wie deine App die einzelnen Anforderungen erfüllen soll und welche Technologien du dafür verwenden möchtest (z.B. NavigationStack für Navigation).

Mindestens 5 Screens:

- Auswahlscreen ob Erwachsene Version oder Kinderversion
- Auswahlcreens für Sicherung der App und Auswahl einzelner Punkte
- Anmeldescreen / Registrierungsscreen für beide Versionen
- Screens für Profil (Erwachsenenversion) und Übersicht (beide Versionen)
- für die Kinderversion Screens für Sparen, Finanzen, Belohnung und evtl. Spiele

Navigation:

Es werden mehrere NavigationStack's für die verschiedenen Navigationen zwischen den Screens verwendet. Sheets und Alerts werden angezeigt.

Anzeigen von Daten innerhalb einer Liste:

Einnahmen und Ausgaben werden in einer Liste angezeigt, sowie die Sparziele und Belohnungen.

Abspeichern von Daten (Core Data / Firebase):

Die Daten des Erwachsenen werden in Firebase gespeichert und die Daten der Kinder unter dem Account derer Eltern als Collection des Users.

API Call:

`http://api.deviser.zone/v1/quotes/THB/EUR/json?qty=1&k={YOUR_KEY}`
API Währungsrechner

MVVM:

Ich werde mindestens UserViewModel, ChildViewModel, FinanceViewModel, GameViewModel, RewardViewModel, OverviewViewModel verwenden. Könnten auch mehr werden.

Meilensteine

Die Projektphase ist in vier große Meilensteine unterteilt (alle zwei Wochen). Überlege dir, was du für deine App in dieser Zeit erreichen möchtest, z.B. UI oder bestimmte Features.

Es ist Pflicht, dass jeden Freitag der aktuelle Stand der App auf Github gepusht wird!

Woche 1 Freitag 17.11.23	<ul style="list-style-type: none">• Konzept erstellt• Figma-Prototyp fertiggestellt• Pflichtenheft ausgefüllt und abgegeben
Woche 3 Freitag 01.12.23	<ul style="list-style-type: none">• Ordnerstruktur erstellen• Firebase integrieren• die ersten Views erstellen
Woche 5 Freitag 15.12.23	<ul style="list-style-type: none">• MVVM implementieren• API einbinden• Repository erstellen• die View fertigstellen• Logik implementieren
Woche 7 Freitag 05.01.23	<ul style="list-style-type: none">• Spiele implementieren• Belohnungssystem erstellen

Bonus Features

Die oben beschriebenen Anforderungen stellen lediglich Mindestanforderungen dar. Wenn ihr zusätzliche Features integrieren möchtet, ist das natürlich super, sowohl für die Bewertung als auch für euer Portfolio!

Es sollten unbedingt zuerst die Mindestanforderungen erfüllt sein, bevor an Bonus-Features gearbeitet wird.

Feature	Erklärung
kleine Spiele	Als Belohnung fürs Sparen bekommt das Kind Punkte mit denen es kleine Spiele freischalten kann. Ich möchte versuchen mit SpriteKit zu arbeiten habe aber noch keine Vorstellung und muss mich erst einarbeiten.
Kommunikation zwischen Eltern und Kind	Die Eltern können den Kindern Spartipps geben, ihnen Emojis zuweisen oder extra Punkte geben. Das Kind kann sich mit einem Emoji bedanken.

Dieses Dokument muss unterschrieben werden, um einen "Vertrag" über die zu erbringende Leistung mit uns zu simulieren.

Mir ist bewusst, dass meine App anhand der Angaben in diesem Dokument bewertet wird.

Ueckermünde, 16. November 2023

Ort, Datum



Unterschrift