

Dokumentation ProductAR

Maximilian Rehberger

July 26, 2019

1 Inhaltsverzeichnis

1	Inhaltsverzeichnis	2
2	Einleitung	5
2.1	Zweck	5
3	Allgemeine Übersicht	6
3.1	Beschreibung Ausgangssituation	6
3.2	Produkteinsatz	6
3.3	Produktumfeld	6
3.4	Produktfunktionalität	6
3.5	Personas	6
3.5.1	Nutzer	6
3.5.2	Verkäufer	6
3.5.3	Admin	6
4	Architekturkonzept und Entwurf	7
4.1	Ursprüngliches Architekturkonzept	7
4.2	Aktualisiertes Architekturkonzept	7
4.3	Anfängliche Skizze Datenbankentwurf	8
4.3.1	MySQL Datenbank (Remote)	8
4.4	Anfängliche Skizze Java Klassen	8
4.5	Endgültige Skizze Datenbankentwurf	8
4.5.1	SQLite Datenbank (Lokal)	8
4.5.2	MySQL Datenbank (Remote)	8
4.6	Endgültige Skizze Java Klassen	8
4.7	Übersicht Backend Server	8
4.8	Übersicht REST API	8
4.9	Technische Entscheidungen	8
4.9.1	Warum Android?	8
4.9.2	Welche Androidversion?	8
4.9.3	Welche Entwicklungsumgebung?	8
4.9.4	Warum eine MySQL Datenbank?	8
4.9.5	Warum eine REST API?	8
4.9.6	Vergleich mit Alternativlösungen	8
4.9.6.1	Firebase von Google	8
4.9.6.2	Alternative Datenbankmodelle	8
5	Technische Dokumentation	9
5.1	Android Manifest	9

5.2	Java Interfaces	9
5.2.1	ObjectInterface	9
5.2.2	ScanResultReceiver	9
5.2.3	IRetrofitCRUD	9
5.2.4	JsonPlaceholderApi	9
5.3	Java Klassen	9
5.3.1	Objekt Klassen	9
5.3.1.1	Object Class (Abstract)	9
5.3.1.2	Product	9
5.3.1.3	User	9
5.3.1.4	Model	9
5.3.1.5	Photo	9
5.3.1.6	Price	9
5.3.1.7	Shop	9
5.3.1.8	Category (Enum)	9
5.3.1.9	Currency (Enum)	9
5.3.1.10	Interval (Enum)	9
5.3.2	Aktivität Klassen	9
5.3.2.1	MainActivity	9
5.3.2.2	SplashScreen	9
5.3.2.3	ProductArActivity	9
5.3.2.4	ProductScanActivity	9
5.3.2.5	CaptureActivityPortrait	10
5.3.2.6	LastScannedProductsActivity	10
5.3.2.7	CreateProductActivity	10
5.3.2.8	ProductDetailActivity	10
5.3.2.9	ProductPhotoGalleryActivity	10
5.3.2.10	ProductPhotoDetailActivity	10
5.3.2.11	CreatePriceActivity	10
5.3.2.12	PriceHistoryActivity	10
5.3.2.13	RegisterActivity	10
5.3.2.14	LoginActivity	10
5.3.2.15	ProfileActivity	10
5.3.2.16	SettingsActivity	10
5.3.2.17	InfoActivity	10
5.3.3	Adapter Klassen	10
5.3.3.1	ProductListAdapter	10
5.3.3.2	PhotoAdapter	10
5.3.4	Hilfs Klassen	10
5.3.4.1	GeneralHelper	10
5.3.4.2	BarcodeHelper	10
5.3.4.3	QRCodeHelper	10
5.3.4.4	LoginHelper	10
5.3.4.5	SettingsHelper	11

5.3.4.6	ImageHelper	11
5.3.4.7	PhotoHelper	11
5.3.4.8	UploadHelper	11
5.3.4.9	PriceHelper	11
5.3.5	Fragment Klassen	11
5.3.5.1	ScanFragment	11
5.3.5.2	CustomArFragment	11
5.3.6	Retrofit Schnittstelle	11
5.3.7	Network Monitor	11
5.3.8	Background Service	11
5.3.9	Notifications	11
5.4	Ressourcen	11
5.4.1	Layout	11
5.4.2	Drawable Icons	11
5.4.3	App Icon	11
5.4.4	Animation	11
5.4.5	Menu	11
5.4.6	Assets	11
5.4.7	Values	11
5.5	Rest Api	11
6	Veröffentlichung im Google Play Store	12
6.1	Store Eintrag	12
6.2	Alpha Test	12
6.3	Beta Test	12
7	Zukünftige Entwicklungen	13
8	Fazit	14
9	Verwendete Technologie, Frameworks und Software	15
10	Verlinkung Repositories	16
11	Verlinkung Tutorials	17
12	Quellenangabe	18

2 Einleitung

2.1 Zweck

Produkte können zum Beispiel beim Einkaufen mit dem Smartphone gescannt werden und erkannt werden. Informationen werden angezeigt wie zum Beispiel Bilder oder ein Preisvergleich. Mithilfe der App soll man einen Barcode einscannen können und Informationen zu den Produkten erhalten. Weiterhin kann der Nutzer ein Produkt in Augmented Reality (AR) testen und sieht somit wie es in Wirklichkeit aussehen wird, wenn er es kaufen würden.

3 Allgemeine Übersicht

3.1 Beschreibung Ausgangssituation

Es gibt bereits viele Shopping-Apps wie zum Beispiel Ikea, H&M oder S'Oliver. Das Problem ist, dass jeder am Ende für jedes Geschäft eine eigene App auf dem Smartphone hat. Diese App soll die Möglichkeiten geben mehrere unterschiedliche Produkte in einer App zu speichern und zu verwalten. Also eine App für alle Produkte.

3.2 Produkteinsatz

Die App kann zum Beispiel als Einkaufsliste oder Wunschliste für Produkte eingesetzt werden. Darüber hinaus bieten sich noch viele weitere

3.3 Produktumfeld

Die App wird hauptsächlich im privaten Umfeld umgesetzt, beim Einkaufen in Geschäften oder Online-Einkauf.

3.4 Produktfunktionalität

Scannen von Produkten, Informationen zu Produkten, Preisvergleich, Bilder hochladen für Produkte, Produkte in AR testen.

3.5 Personas

3.5.1 Nutzer

3.5.2 Verkäufer

3.5.3 Admin

4 Architekturkonzept und Entwurf

4.1 Ursprüngliches Architekturkonzept

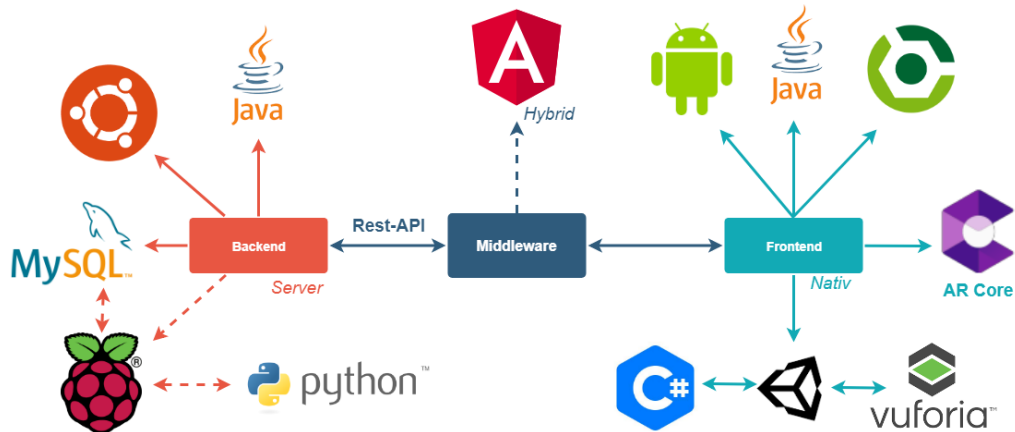


Figure 1: Ursprüngliches Architekturkonzept

4.2 Aktualisiertes Architekturkonzept

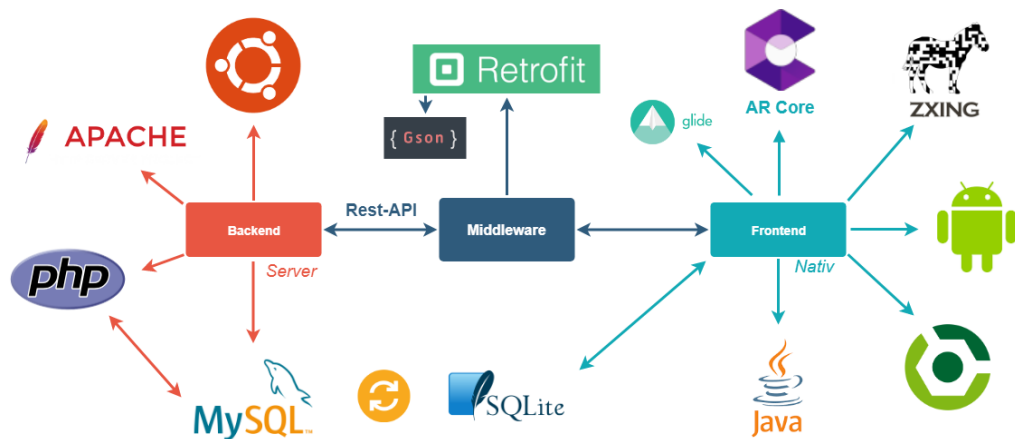


Figure 2: Aktualisiertes Architekturkonzept

4.3 Anfängliche Skizze Datenbankentwurf

4.3.1 MySQL Datenbank (Remote)

4.4 Anfängliche Skizze Java Klassen

4.5 Endgültige Skizze Datenbankentwurf

4.5.1 SQLite Datenbank (Lokal)

todo: erzeugen

4.5.2 MySQL Datenbank (Remote)

todo: erzeugen

4.6 Endgültige Skizze Java Klassen

todo: erzeugen

4.7 Übersicht Backend Server

4.8 Übersicht REST API

4.9 Technische Entscheidungen

4.9.1 Warum Android?

4.9.2 Welche Androidversion?

4.9.3 Welche Entwicklungsumgebung?

4.9.4 Warum eine MySQL Datenbank?

4.9.5 Warum eine REST API?

4.9.6 Vergleich mit Alternativlösungen

4.9.6.1 Firebase von Google

4.9.6.2 Alternative Datenbankmodelle

5 Technische Dokumentation

5.1 Android Manifest

5.2 Java Interfaces

5.2.1 ObjectInterface

5.2.2 ScanResultReceiver

5.2.3 IRetrofitCRUD

5.2.4 JsonPlaceHolderApi

5.3 Java Klassen

5.3.1 Objekt Klassen

5.3.1.1 Object Class (Abstract)

5.3.1.2 Product

5.3.1.3 User

5.3.1.4 Model

5.3.1.5 Photo

5.3.1.6 Price

5.3.1.7 Shop

5.3.1.8 Category (Enum)

5.3.1.9 Currency (Enum)

5.3.1.10 Interval (Enum)

5.3.2 Aktivität Klassen

5.3.2.1 MainActivity

5.3.2.2 SplashScreen

5.3.2.3 ProductArActivity

5.3.2.4 ProductScanActivity

5.3.2.5 CaptureActivityPortrait

5.3.2.6 LastScannedProductsActivity

5.3.2.7 CreateProductActivity

5.3.2.8 ProductDetailActivity

5.3.2.9 ProductPhotoGalleryActivity

5.3.2.10 ProductPhotoDetailActivity

5.3.2.11 CreatePriceActivity

5.3.2.12 PriceHistoryActivity

5.3.2.13 RegisterActivity

5.3.2.14 LoginActivity

5.3.2.15 ProfileActivity

5.3.2.16 SettingsActivity

5.3.2.17 InfoActivity

5.3.3 Adapter Klassen

5.3.3.1 ProductListAdapter

5.3.3.2 PhotoAdapter

5.3.4 Hilfs Klassen

5.3.4.1 GeneralHelper

5.3.4.2 BarcodeHelper

5.3.4.3 QRCodeHelper

5.3.4.4 LoginHelper

5.3.4.5 SettingsHelper

5.3.4.6 ImageHelper

5.3.4.7 PhotoHelper

5.3.4.8 UploadHelper

5.3.4.9 PriceHelper

5.3.5 Fragment Klassen

5.3.5.1 ScanFragment

5.3.5.2 CustomArFragment

5.3.6 Retrofit Schnittstelle

5.3.7 Network Monitor

5.3.8 Background Service

5.3.9 Notifications

5.4 Ressourcen

5.4.1 Layout

5.4.2 Drawable Icons

5.4.3 App Icon

5.4.4 Animation

5.4.5 Menu

5.4.6 Assets

5.4.7 Values

5.5 Rest Api

6 Veröffentlichung im Google Play Store

6.1 Store Eintrag

6.2 Alpha Test

6.3 Beta Test

7 Zukünftige Entwicklungen

8 Fazit

9 Verwendete Technologie, Frameworks und Software

10 Verlinkung Repositories

11 Verlinkung Tutorials

12 Quellenangabe