

# Time4You



02.07.2023  
ZAPP

Yakup Cilesiz und Felix Kölling

## Inhaltsverzeichnis

1. Externe Abhängigkeiten
2. Verwendete Android Komponenten
3. Struktur
4. Soll / Ist
5. Aufteilung der Arbeiten
6. Fazit

## Externe Abhängigkeiten und Verwendete Android Komponenten

Unsere App verwendet verschiedene externe Abhängigkeiten, Bibliotheken und Android-Komponenten.

Zunächst verwenden wir die Klasse `TimePickerDialog` aus dem Android-Framework, um einen Zeitpicker für die Auswahl von Uhrzeiten bereitzustellen. Dies ermöglicht es bestimmte Zeitpunkte auszuwählen und in der App zu verwenden.

Die Klasse `ContentValues` wird verwendet, um Werte in einer Datenbank zu speichern und zu verwalten. In Verbindung mit den Klassen `SQLiteDatabase` und `SQLiteOpenHelper` ermöglicht sie den Zugriff auf eine SQLite-Datenbank für die dauerhafte Datenspeicherung in der App. Die Datenbank ist notwendig um Nutzernamen, Profildaten sowie den aktuellen Punktestand des Nutzers bereitzustellen.

Die Klasse `Intent` wird verwendet, um zwischen verschiedenen Aktivitäten und Komponenten innerhalb der App zu navigieren. Durch das Starten von Aktivitäten mit `Intents` können Benutzeraktionen und der App-Fluss gesteuert werden. Wir benutzen sie hauptsächlich, damit der Nutzer seinen Profilnamen und sein Nutzerbild in der `LoginActivity` auswählen

kann. Sie wird auch im Shop verwendet, wo der Nutzer seine Punkte ausgeben kann.

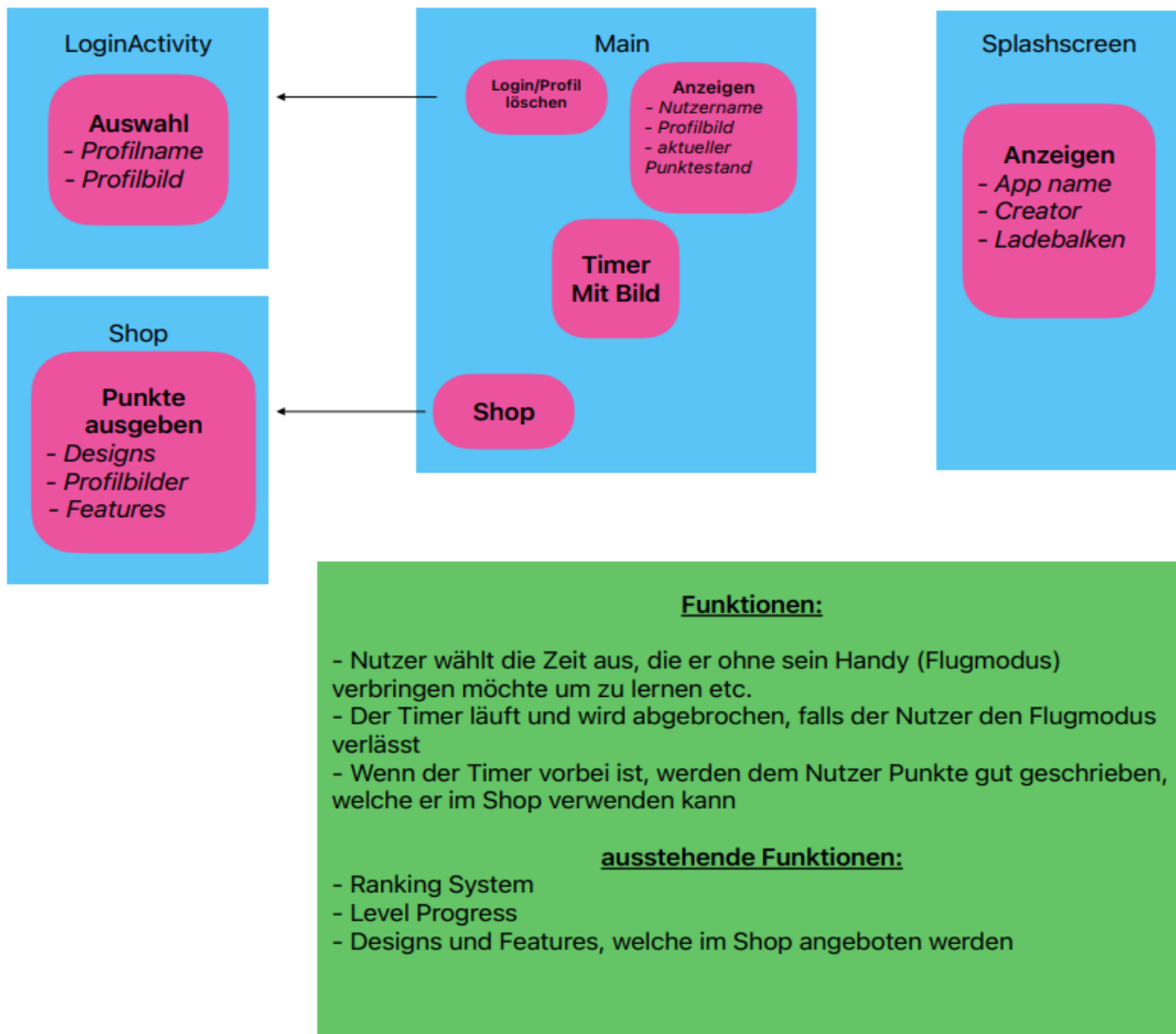
Die Klasse RingtoneManager ermöglicht den Zugriff auf Systemklingeltöne, sodass die App Benutzern die Auswahl eines bestimmten Klingeltons für Alarmer oder Benachrichtigungen ermöglicht. In unserem Projekt findet sie Anwendung, damit der Nutzer eine Benachrichtigung erhält, wenn der Timer beendet ist.

Die Klasse CountDownTimer wird verwendet, um einen Countdown-Timer in der App zu implementieren. Dies ermöglicht die Anzeige einer verbleibenden Zeit und das Auslösen von Aktionen nach Ablauf des Timers.

Die Klasse Settings ermöglicht den Zugriff auf die Android-Geräteeinstellungen. Damit kann die App beispielsweise auf die Lautstärke oder Benachrichtigungseinstellungen zugreifen und diese anpassen. In unserem Projekt, verwenden wir die Klasse, um dem Nutzer direkt zu ermöglichen, den Flugmodus zu aktivieren, falls dieser noch nicht angeschaltet sein sollte

Darüber hinaus verwenden wir verschiedene Android-Komponenten wie Buttons, ImageView, TextView und Toast, um eine Benutzeroberfläche zu erstellen und Interaktionen mit den Benutzern zu ermöglichen. Die Klasse AppCompatActivity wird als Basisaktivität für die App verwendet, um alle grundlegenden Funktionen und Funktionen einer Android-App zu unterstützen.

## Struktur



## Soll / Ist

Die Grundfunktion der App ist unverändert geblieben. Wir haben eine Anwendung entwickelt, mit der Benutzer den Flugmodus aktivieren können, um nicht gestört zu werden. Vor dem Aktivieren des Flugmodus kann ein Timer eingestellt werden, der nach Ablauf automatisch den Flugmodus deaktiviert.

Wir sind stolz darauf, dass wir diese Kernfunktion erfolgreich umgesetzt haben, da dies unser Hauptziel war. Allerdings mussten wir feststellen, dass wir nicht genügend Zeit hatten, um das geplante Punkte- und Shop-System zu implementieren. Außerdem hatten wir vor ein "Level Progress" zu implementieren, damit der Nutzer motiviert bleibt, bei seinen Lernsessions unsere App zu benutzen. Auch ein Ranking System war geplant, damit sich verschiedene Nutzer messen können. Dieser Teil des

Projekts wurde aufgrund von Ressourcenbeschränkungen und Zeitdruck nicht rechtzeitig fertiggestellt. Außerdem hatten wir an manchen Stellen erhebliche Schwierigkeiten, welche nicht vorherzusehen waren.

Trotz dieser Herausforderungen sind wir mit dem aktuellen Stand der App zufrieden. Wir haben eine solide Grundlage geschaffen und planen, das Punkte- und Shop-System in zukünftigen Updates zu implementieren. Der Fokus lag auf der Funktionalität und Benutzerfreundlichkeit der App. Außerdem sollte unsere App im Grunde simpel sein. Der Nutzer sollte nicht durch die App abgelenkt werden und sofort in der Lage sein, den Timer zu starten um in seine Aktivitäten eintauchen zu können.

Insgesamt haben wir wertvolle Erfahrungen gesammelt und wertvolle Einblicke in die App-Entwicklung gewonnen. Android Studio ist eine tolle IDE und Kotlin eine verständliche Sprache, in welche man sich gut einlesen kann. Man hat eine stetig steigende Lernkuve feststellen können. Nach dem Bewältigen jeder Hürde, haben wir Motivation geschöpft um uns den nächsten Problemen zu Stellen.

## **Aufteilung der Arbeiten**

Im Rahmen dieses Projektberichts möchten wir einen Einblick in die Aufteilung der Arbeiten bei unserem Projekt geben. Die Entwicklung des Codes erfolgte größtenteils in Zusammenarbeit, wobei Yakup etwa 60% und Felix etwa 40% der Arbeit übernahm. Wir haben eng zusammengearbeitet, um sicherzustellen, dass der Code den Anforderungen entspricht und alle gewünschten Funktionen enthält.

Die Planung des Projekts wurde von beiden Teammitgliedern gemeinsam durchgeführt. Wir haben Ideen ausgetauscht, die Anforderungen diskutiert und die Aufgaben strukturiert. Diese gemeinsame Planung ermöglichte es uns, ein klares Verständnis für den Umfang des Projekts zu entwickeln und effektiv daran zu arbeiten.

Auch der Projektbericht wurde größtenteils gemeinsam verfasst. Felix war dabei für etwa 60% und Yakup für etwa 40% der Arbeit verantwortlich. Wir haben uns regelmäßig abgestimmt, um den Fortschritt zu besprechen und sicherzustellen, dass alle relevanten Informationen im Bericht enthalten sind.

# Fazit

Das Programmieren der Time4You-App war eine anspruchsvolle Aufgabe, die mitunter viel Zeit und Aufwand erforderte.

Glücklicherweise standen uns zahlreiche Online-Tutorials zur Verfügung, die uns bei der Entwicklung unterstützten. Dennoch haben wir unterschätzt, wie viel Zeit das gesamte Projekt in Anspruch nehmen würde. Trotz der Herausforderungen war es äußerst interessant, einen Einblick in die Welt der App-Programmierung zu bekommen.

Wir haben wertvolle Erfahrungen gesammelt und sind stolz auf das erreichte Ergebnis.