

FishAlarm - Konzept

By Nico Egli

# Inhalt

[Inhalt 1](#_Toc113546007)

[Sinn der App 2](#_Toc113546008)

[Zielgruppe 2](#_Toc113546009)

[Anforderungen 3](#_Toc113546010)

[Funktional: 3](#_Toc113546011)

[Minimum: 3](#_Toc113546012)

[Extras: 3](#_Toc113546013)

[Nicht funktional: 3](#_Toc113546014)

[Mockups 4](#_Toc113546015)

[Architektur 5](#_Toc113546016)

[Verwendete Technologien 5](#_Toc113546017)

[Testkonzept 6](#_Toc113546018)

# Sinn der App

Die App ist ein sehr simpler Wecker. Sie kann dazu genutzt werden sich aus dem Schlaf wecken zu lassen. Dadurch fällt es einem leichter morgens aufzustehen.

## Zielgruppe

Diese Applikation richtet sich an Personen der Gesellschaft, die Ihren Schlaf an einem zuvor festgelegten Zeitpunkt beenden müssen oder wollen.

# Anforderungen

Die Anforderungen der App können in die funktionalen Anforderungen und nicht funktionalen Anforderungen unterteilt werden. Funktional bedeutet es sind wichtige Funktionen in der App, nicht funktional sind

## Funktional:

### Minimum:

* Wecker
  + Zeit kann eingestellt werden
  + Wecker kann deaktiviert werden
  + Klingelton kann ausgewählt werden.

### Extras:

* NFC Integration
* Traumtagebuch
* Handy schütteln
* Glyph Interface (leider nicht möglich)

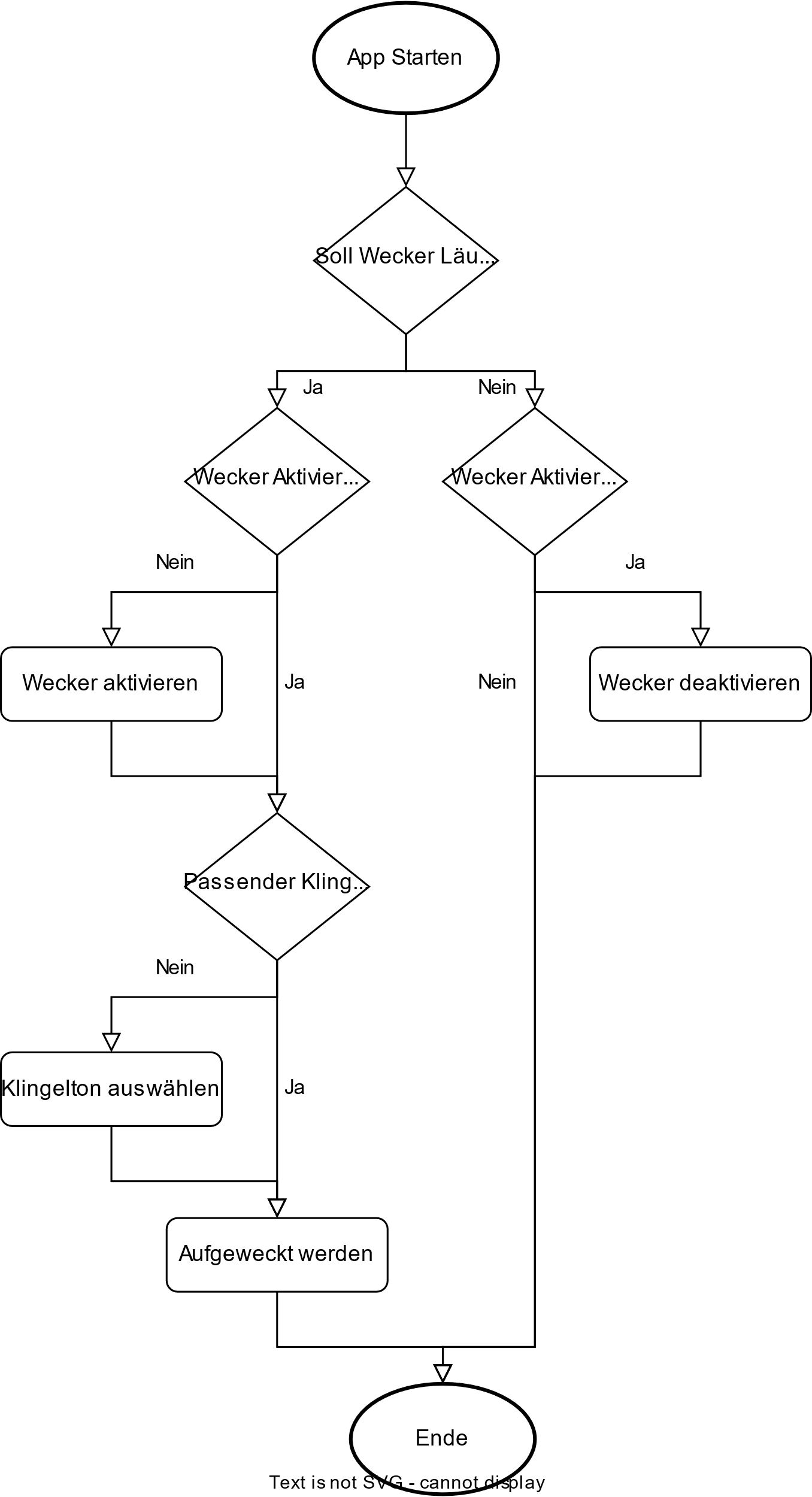
## Nicht funktional:

* Die App soll einfach zu bedienen sein
* Die App soll verständlich sein
* Alle Schaltflächen und Texte sollten gross genug sein
* Passende Farben

# Mockups

Das Mockup der App wurde mit Moqups erstellt und kann unter dem folgenden Link eingesehen werden: <https://app.moqups.com/n7od7V1loXsC5z7gWRkzn8V1ZoeFjA1U/view/page/ad64222d5>

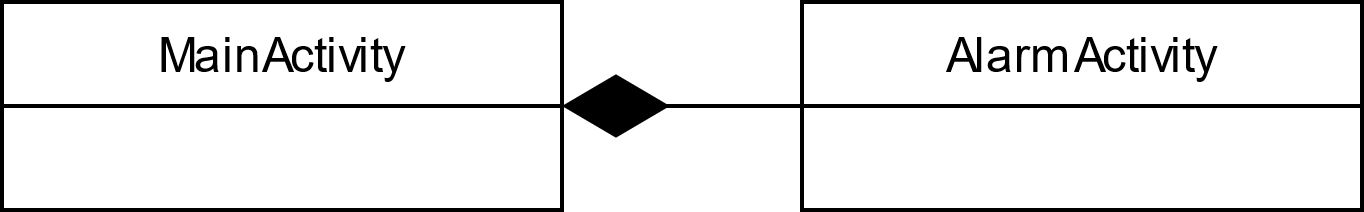
Folgendes Bild ist das Flowchart zu der App:



# Architektur

Die App besitzt sehr wenig Logik, eine eigene Klasse hätte sich nicht gelohnt. Deshalb wird die Logik ebenfalls in den UI-Klassen geschrieben.

Das Klassendiagramm sieht in etwa so aus:



## Verwendete Technologien

Die gesamte App wurde vollständig mit Kotlin geschrieben. Die Entwicklung wurde mit Android Studio umgesetzt. Ansonsten wurden keine weiteren Technologien verwendet.

# Testkonzept

| Identifikation | Vorbedingungen | Vorgehen | Erwartetes Resultat |
| --- | --- | --- | --- |
| FishAlarm-Test-01 | 1. App ist geöffnet 2. Wecker ist aktiviert | Zeit auswählen mit Time-Picker | Wecker geht um die angegebene Zeit los |
| FishAlarm-Test-02 | 1. App ist geöffnet 2. Zeit ist eingestellt | Wecker mit Switch aktivieren | Wecker geht um die angegebene Zeit los |
| FishAlarm-Test-03 | 1. App ist geöffnet 2. Zeit ist eingestellt | Wecker mit Switch deaktivieren | Wecker geht nicht um die angegebene Zeit los |
| FishAlarm-Test-04 | 1. App ist geöffnet 2. Wecker ist aktiviert 3. Zeit ist eingestellt | Klingelton wechseln | Wecker geht mit dem ausgewählten Klingelton los |
| FishAlarm-Test-05 | 1. App ist geöffnet 2. Wecker ist aktiviert 3. Zeit ist eingestellt 4. Wecker geht los | “Stop Alarm” drücken | Wecker stoppt |