

# Dokumentation: Symcon-App



Ann-Kathrin Harms

1.10.2021

## Inhalt

Beschreibung des Auftrages:.....	1
Beschreibung der Ausführung:.....	1
Auseinandersetzung mit Flutter:.....	1
Planung des Programmaufbaus: .....	1
Schreiben des Programms:.....	2
Fazit: .....	3

## Beschreibung des Auftrages:

Die Aufgabe ist es die Startseite einer App mit dem Framework „Flutter“ zu entwickeln. Auf dieser Seite soll das Logo der Firma, zwei Zeilen Text, wobei die erste Zeile dick gedruckt und größer als die zweite Zeile ist, sowie am unteren Rand ein Button mit einem Icon und einem Schriftzug sein. Weitere Vorgaben sind die Hintergrundfarbe #111C2D, die Akzentfarbe #4FAD9B und die Schriftart Poppins.

## Beschreibung der Ausführung:

### Auseinandersetzung mit Flutter:

Da ich vorher noch nicht mit Flutter gearbeitet hatte, musste ich dieses erstmal installieren. Also habe ich die SDK <sup>1</sup>heruntergeladen und mit Hilfe des Leitfadens dieses und die IDE<sup>2</sup> Android Studio installiert. Mit dem Cookbook von Flutter und einigen Tutorials probierte ich einige Codebeispiele aus, um mich in der Sprache und dem Aufbau zurecht zu finden.

### Planung des Programmaufbaus:

Nachdem ich ein paar Proben geschrieben hatte, setzte ich mich daran, das Programm für diesen Auftrag zu planen, also wie ich das vorgegebene Layout umsetzen könnte. Da Flutter mit Containern arbeitet, habe ich mich dazu entschieden, das Logo, die Texte und den Button in einzelne Container zu setzen.

Jedes Objekt ist in Flutter ein Widget, so ist das Logo ein anders Objekt als die Überschrift, der Text oder der Button.

Struktur:

Scaffold

|--> Container

    |-> Column

        |->Row mit dem Firmenlogo

        |-> Row mit der Überschrift

        |-> Row mit dem Text

        |-> Row mit dem Button

---

<sup>1</sup> Software Development Kit

<sup>2</sup> Integrated Development Environment

## Schreiben des Programmes:

Zuerst habe ich den Beispielcode, welchen Flutter beim Erstellen eines neuen Programmes bereitstellt, zu großen Teilen gelöscht, um dann den Unterbau der App zu schreiben. In der Klasse `MyApp` übergab ich dem Attribut `home`, welches die Startseite einer App aufruft, als Parameter die Klasse `HomeScreen`.

In der Klasse `HomeScreen` richtete ich zuerst die Struktur aus der Planung ein. Also ein Scaffold, welcher Parameter für die `backgroundColor` und den `body` übergeben bekommen hat. Der `body` wiederum bekam das Widget `Column`, welches wiederum vier Kindwidget vom Typ `Row` hat. Die erste `Row` beinhaltet das Logo der Firma SYMCON, das Zweite die Überschrift, das Dritte den Text und das letzte den Button.

Es stellte sich allerdings heraus, dass sich der Button nicht wie gewünscht am unteren Rand des Gerätes befand, sondern direkt unter dem Text, da das Widget des Textes nicht groß genug war. Auch waren die Abstände der einzelnen Texte nicht wie gewünscht. Nach einiger Recherche wurden die `Column` mit dem Widget `Container` umwickelt, um die Abstände zum äußeren Rand des Gerätes mit dem Attribut `margin` festzulegen. Die Widgets des Logos und der Überschrift wurden einzeln mit `Padding` umwickelt. Der Text wird mit `Expanded` ummantelt, um den Button an den unteren Rand zu schieben.

Um das Logo der Firma anzuzeigen, fotografierte ich das Logo zuerst von der Seite und bearbeitete dieses über das Programm „Gimp“. Um es dann in der Datei „pupsec.yaml“ einzubinden und im Programm mit der Klasse `Image` zu verwenden. Dort legte ich über das Bild mit dem Attribut `color` die Akzentfarbe.

Weiter ging es zu dem Button, genauer gesagt dessen Aufschrift, welcher aus einem Icon und einem Text besteht. Dem Button wurden also das Widget `Row` mit zwei Kindwidgets von `Padding` übergeben. In einem ist das Icon, in dem anderen der Text.

Um die Texte in der vorgegebenen Schriftart anzeigenzulassen, band ich das Package „Google\_Fonts“ ein und verwendete die Schriftart „Poppins“. Daraus definierte ich in der Datei „main.dart“ in der Klasse `myApp` die Stile für die verschiedenen Texte und den Button. So bekam der Button einen weißen Hintergrund und eine schwarze Schriftfarbe. Die Überschrift die Schriftfarbe „Weiß“, die Größe „24“ und das Gewicht von „600“. Der Bodytext bekam ebenfalls die Farbe „Weiß“, die Größe ist hier „20“ und das Gewicht „normal“.

Da mir die Bearbeitung des Logos noch nicht gefiel, suchte ich mir im Internet ein Tool um einen SVG Pfad, in welchem das Logo auf der Firmenseite angegeben ist, in das Dateiformat PNG zu transformieren. Dieses war schnell gefunden und ich tauschte das Bild im Programm aus.

Somit war die Aufgabe eigentlich beendet. Allerdings fiel beim erneuten durchlesen des Codes auf, das einige Widgets vom Typ `Row` lediglich ein Kind hielten, weshalb ich beschloss diese zu entfernen.

Die finale Struktur sieht also wie folgt aus:

Scaffold

```
|-> SafeArea
  |-> Container
    |-> Column
      |-> Padding (für das Logo)
        |-> Image
      |-> Padding (für die Überschrift)
        |-> Text
      |-> Padding (für den Bodytext)
        |-> Text
      |-> TextButton
        |-> Row
          |-> Padding (für das Icon)
            |-> Icon
          |-> Padding (für den Text)
            |-> Text
```

## Fazit:

Mein Fazit dieses Programms ist es, dass ich viel über Flutter und die Darstellung von verschiedenen Komponenten auf einer Oberfläche erfahren habe. Durch verschiedene Codebeispiele und vorherige Erfahrung im Webdesign, ist es mir möglich gewesen die einzelnen Komponenten der Darstellung schnell auseinander zu schneiden und in den Widgets anzuordnen. Einzig die Einbindung des Firmenlogos, hat für mich zunächst eine Herausforderung dargestellt. Eigentlich habe ich mir vorgenommen dieses direkt als SVG zu nutzen, schließlich habe ich mich dazu entschieden es als PNG einzubinden.

Genutzte Programme:

- Android Studio
- Gimp

Genutzte Packages:

- Google Fonts

Farblegende:

Orange: Widget

Grün: Klasse

Blau: Attribute