

Hausarbeit

1.1 Beschreibung

Die Hausarbeit wird in diesem Semester anstelle einer schriftlichen SEP durchgeführt. Untenstehende Aufgaben müssen Sie in 5 Stunden lösen. Bitte lesen Sie zunächst die ganze Aufgabenstellung und teilen sich die Zeit entsprechend ein.

Die Aufgaben unter 1.2 werden bei der Bewertung gleich gewichtet. Wesentlich dabei: Wurde die Aufgabe erfüllt? Ist der geforderte Funktionsumfang umgesetzt? Und mit geringerem Gewicht: Ist die App gut gestaltet und gut benutzbar? Ist der Code lesbar und gut strukturiert. Wurden die Vorgaben zur Abgabe eingehalten? Wie ist die Qualität der Dokumentation?

Ihr Resultat senden Sie als ZIP-Datei zurück (siehe Abgabe 1.3 unten).

1.2 Aufgabenstellung

Sie erhalten den Auftrag, eine Online Shop App zu entwickeln. Basis Ihrer App ist eine JSON-Datei mit 50 Produkten. Diese Datei finden Sie mitgeliefert.

Bei der eingesetzten Technologie können Sie zwischen den beiden aus der Vorlesung bekannten wählen:

React Native : React Native mit Expo

SwiftUI : iOS Native mit SwiftUI

Für React Native benötigen Sie eine aktuelle Expo-Installation. Diese können Sie auf der Kommandozeile (s. <https://docs.expo.io>) installieren und starten:






```
# Create a project named my-app. Select the "blank" template
# when prompted
expo init my-app

# Navigate to the project directory
cd my-app

# Starting the development server
expo start
```

Falls Sie weder eine funktionierende Expo-Installation noch über eine gut verwendbare Xcode-Installation verfügen, können Sie als **Notlösung** eine Expo-Umsetzung im Online-Tool Snack () probieren.

Aufgabe 1: Basis

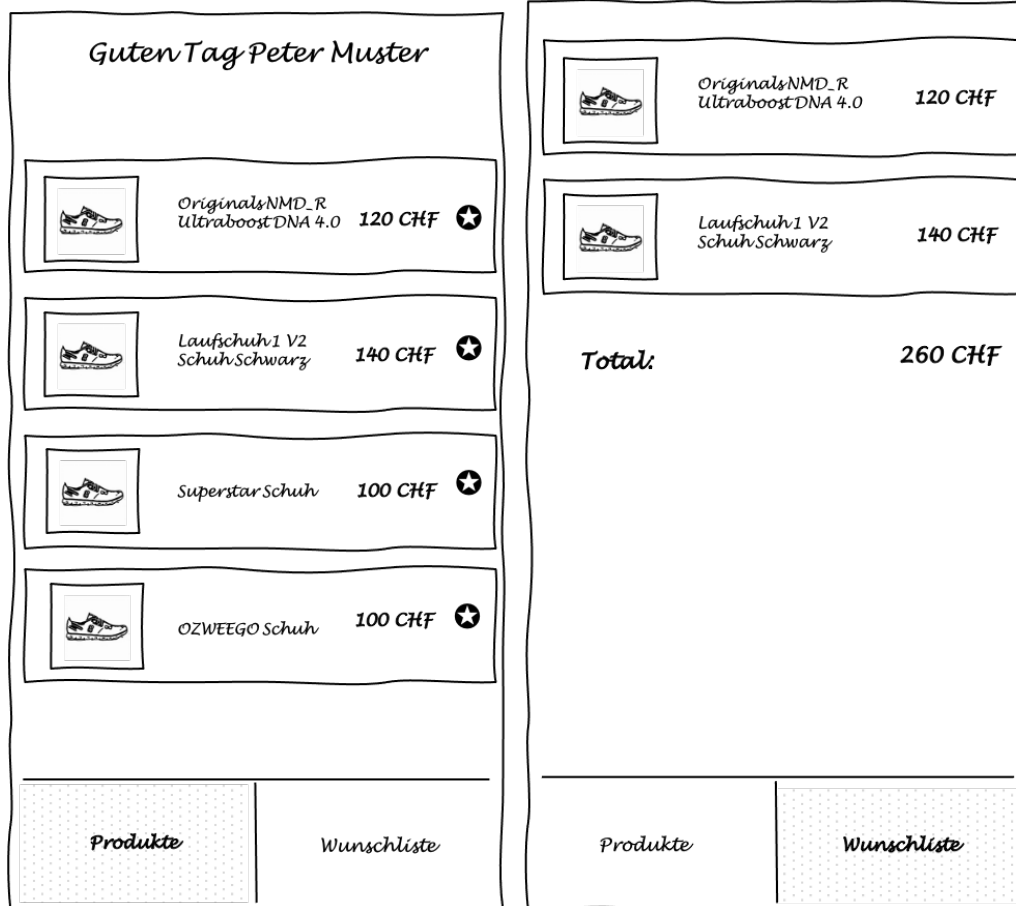
Guten Tag Peter Muster		
	Originals NMD_R Ultraboost DNA 4.0	120 CHF
	Laufschuh 1 V2 Schuh Schwarz	140 CHF
	Superstar Schuh	100 CHF
	OZWEEGO Schuh	100 CHF
	Forum Mid Schuh	190 CHF

Erstellen Sie eine App, die zunächst eine Liste mit sämtlichen Produkten aus der JSON-Datei anzeigt. Für jedes Produkt sollen:

- der Name
- der Preis
- ein Foto als Thumbnail

ausgegeben werden. Oberhalb der Produktliste kann der Benutzer persönlich begrüsst werden ("Guten Tag Peter Muster")

Aufgabe 2: Wunschliste



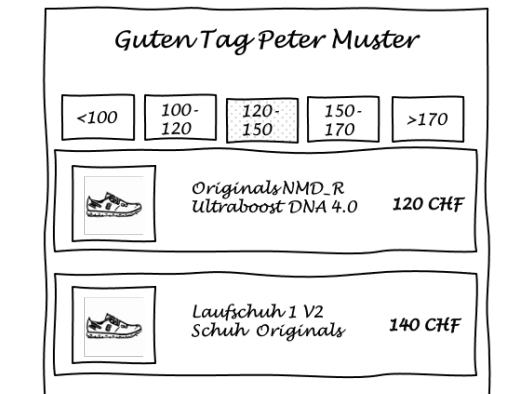
Implementieren Sie eine TabView (Reiteranzeige) mit zwei Reitern:

- Produkte
- Wunschliste

Implementieren Sie eine Funktionalität, so dass ein Produkt via Button in die Wunschliste gelegt werden kann. Zeigen Sie die Wunschliste im Reiter „Wunschliste“ an.

Zeigen Sie unter den Produkten den Gesamtbetrag an.

Aufgabe 3: Preisfilterung



Implementieren Sie eine Preisfilterung, bei der oberhalb der Produktliste eine Leiste mit Knöpfen für folgende Preise angezeigt wird:

- <100
- 100 – 120
- 120 – 150
- 150 – 170
- >170

Beim Drücken auf einen der Knöpfe sollen nur die Produkte angezeigt werden, deren Preis im angegebenen Bereich liegt.

Aufgabe 4: Frage zu iOS

Beantworten Sie diese Frage nur, wenn Sie als Technologie für die App React Native gewählt haben!

Das folgende Programm erstellt einen Zähler mit 2 Knöpfen. 5 Teile fehlen, numeriert von 1–5. Ergänzen Sie diese Teile. Schreiben Sie die Lösung in die Kurzdokumentation (siehe Aufgabe 6).

```
struct ContentView: View {
    ① var counter : ② = 10
    var body: some View {
        VStack {
            Spacer()
            Text(③)
                .padding().font(
                    .system(size: 120))

            Spacer()
            HStack {
                Button(action: {
                    ④
                }, label: {
                    Text("Minus")
                        .frame(maxWidth: .infinity)
                }).padding()
                    .border(Color.blue)

                Button(action: {
                    counter += 1
                }, label: {
                    Text("Plus")
                        .frame(maxWidth: .infinity)
                }).padding()
                    .border(Color.blue)
            }.padding()
        }
    }
}
```



Aufgabe 5: Frage zu React Native

Beantworten Sie diese Frage nur, wenn Sie als Technologie für die App SwiftUI gewählt haben!

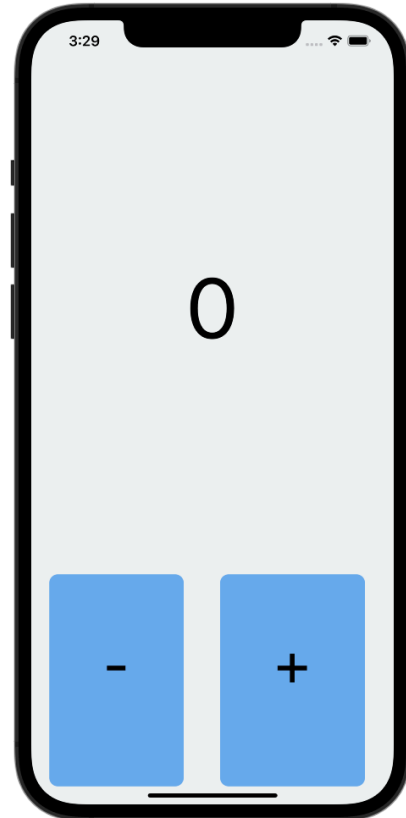
Das folgende Programm erstellt einen Zähler mit 2 Knöpfen. 5 Teile fehlen, numeriert von 1–5. Ergänzen Sie diese Teile. Schreiben Sie die Lösung in die Kurzdokumentation (siehe Aufgabe 6).

```
import React, { ① }
  from 'react'
import { Text, TouchableOpacity,
  View, StyleSheet }
  from 'react-native';
import Constants from
  'expo-constants';

export default function App() {

  const [counter, setCounter] = ②

  return (
    <View style={styles.container}>
      <View style={styles.counter}>
        <Text style={styles.countertext}>
          ③
        </Text>
      </View>
      <View style={styles.buttons}>
        <TouchableOpacity
          style={styles.button}
          onPress={() => {
            ④
          }} >
          <Text style={styles.buttontext}><-</Text>
        </TouchableOpacity>
        <TouchableOpacity style={styles.button}
          onPress={() => {
            ⑤
          }} >
          <Text style={styles.buttontext}>>+</Text>
        </TouchableOpacity>
      </View>
    </View>
  );
}
```



Aufgabe 6

Erstellen Sie eine Kurzdokumentation als PDF. Diese sollte mindestens folgendes enthalten:

- Zwei Screenshots
- Ein paar Angaben zur Implementierung (welche Aufgaben wurden gelöst / nicht gelöst, welche Schwierigkeiten gab es, ...)
- Ihre Lösung der Aufgabe 4 oder 5

1.3 Abgabe

Für die Abgabe der Prüfung erstellen Sie eine ZIP-Datei mit Ihren Ergebnissen:

React Native mit Expo: Erstellen Sie ein Zip-Archiv des Projektverzeichnisses inklusive Doku-PDF **aber ohne das `node_modules`** Verzeichnis.

React Native mit Snack: Laden Sie Ihr Projekt als Zip-Datei von Snack herunter und ergänzen Sie das Archiv um das Doku-PDF. **Wichtig:** Erwähnen Sie am Anfang der Doku, dass Sie Snack verwendet haben. Geben Sie ausserdem einen Link zur Snack-Installation an.

SwiftUI: Komprimieren Sie Ihr Projektverzeichnis inkl. Doku-PDF in ein ZIP-Archiv.

Nennen Sie das ZIP-Archiv `<ihr-kürzel>-<ihre-technologie>.zip`, zum Beispiel `stme-swiftui.zip` oder `bkrt-react-native.zip`. Laden Sie diese ZIP-Datei via MS-Teams (Teamname MOBA 2 FS21) hoch (unter **Aufgaben** -> **Abgabe SEP**).

Die Abgabe muss bis spätestens 13:05 Uhr erfolgt sein, zu spät zurückgesendete Arbeiten werden als „verspätete Abgabe“ mit 1.0 bewertet.

Bei technischen Problemen oder Fragen sind wir die ganze Zeit per E-Mail oder MS-Teams Chat erreichbar.

Sollten weitergehende Probleme auftreten (Internet fällt aus, Rechnercrash, etc.) nehmen Sie bitte umgehend mit uns telefonisch Kontakt auf:

- Gerrit Burkert: 079 571 69 72
- Henrik Stormer: 079 860 10 95