Meine App

## Onlysteps

### Version 1.0.0, Datum: 9. Februar 2024

A screenshot of a cell phone

Description automatically generated

Abstract (Kurzbeschreibung)

Die App Onlysteps soll eine Erfassung der zurückgelegten Schritte darstellen. In der App können Tagesziele definiert werden. In einem Kalender kann der Benutzer einsehen, an welchen Tagen er seine Ziele erreicht hat. Zudem können Werte wie Kalorien, Entfernung oder Höhenmeter angezeigt werden. Die App ist als Darkmode entwickelt und in einem Schwarz-Lila Farbschema designed worden. Die App verfügt über eine Accountfunktion, bei der man sich ab- und anmelden kann und ein Profilbild hinterlegen kann.

user stories

Als Benutzer möchte ich die Gesamtanzahl meiner täglichen Schritte auf dem Startbildschirm sehen, um meine Aktivität im Auge zu behalten.

Als Benutzer möchte ich die zurückgelegte Distanz meiner Spaziergänge oder Läufe in Kilometern sehen, um meine Fortschritte zu verfolgen.

Als Benutzer möchte ich Informationen über die überwundenen Höhenmeter während meiner Aktivitäten sehen, um das Gelände und die Schwierigkeit meiner Routen zu verstehen.

Als Benutzer möchte ich eine detaillierte Übersicht meiner täglichen Schritte in einem Kalender abrufen, um Muster und Trends zu erkennen.

Mockups

A screenshot of a cell phone

Description automatically generated A screenshot of a calendar

Description automatically generated A screenshot of a phone

Description automatically generated

Die App Onlysteps verfügt über 3 Seiten: Start, Übersicht und Profil. Auf der Startseite wird der angemeldete Benutzer. Zuerst mit seinem Namen begrüsst. Darunter wird die Anzahl effektiver Schritte des aktuellen Tages mit dem Täglichen Ziel verglichen. Hier gibt es eine Kreisgrafik, welche sich schliesst, sobald das Tagesziel erreicht ist. Darunter werden drei weitere Werte angezeigt: Kalorien, Entfernung, Höhenmeter. Als nächstes wird ein Säulendiagramm mit einer Auswertung zum Wochenstreak angezeigt. Auf der Seite «Übersicht» befindet sich ein Kalender, in dem markiert ist an welchen Tagen der Benutzer sein tägliches Ziel erreicht hat. Auf der Seite «Profil» kann das Profilbild geändert werden sowie persönliche Daten wie Höhe oder Gewicht können geändert werden. Unten kann der Benutzer seine Ziele hinterlegen, einen Tracker verbinden (z.B Fitbit) oder die Datenschutzerklärung einsehen.

Technische Realisierung

Components:

Kalender:

Calendar-Daten: Die Variable calendarData enthält Informationen darüber, ob an einem bestimmten Tag ein Ereignis stattfindet oder nicht.

Rendering der Kalenderansicht: Die Funktion Calendar rendert eine Überschrift mit dem aktuellen Monat, einen Wochentagskopf, die tatsächliche Kalenderansicht basierend auf den calendarData-Daten und einige zusätzliche Textelemente.

Styling: Die Styles werden mit Hilfe von StyleSheet.create definiert, um das Aussehen der Container, Texte und des Kalenders anzupassen. Es gibt verschiedene Stile für den Monat, den Titel, den Wochentagskopf und die Kalenderzellen.

Verwendung von map: Die map-Funktion wird verwendet, um durch die calendarData zu iterieren und für jeden Tag eine CalendarCell-Komponente zu rendern.

Textelemente: Zusätzliche Textelemente, wie der Titel "Da geht noch was!" und die Information über die Anzahl der erreichten Ziele in diesem Monat, werden gerendert.

Diagramm Startbildschirm:

Dieser React-Code erstellt Diagramme für Schritte basierend auf einer Pedometer-Funktionalität. Hier sind die Schlüsselpunkte:

useState und useEffect: Verwendung des useState-Hooks, um den vorherigen und aktuellen Schrittzähler zu speichern. Der useEffect-Hook wird verwendet, um das Abonnement für den Schrittzähler zu starten und zu beenden.

Pedometer-Funktionalität: Die Funktion subscribe wird verwendet, um die Schritte für die vergangene Woche zu erhalten und das aktuelle Schrittzähler-Abonnement einzurichten.

Diagramme: Verwendung von verschiedenen Diagrammtypen aus der react-native-chart-kit-Bibliothek, wie ProgressChart für den aktuellen Schrittzähler und BarChart für die Schritte der vergangenen Tage.

Styling: Die Styles werden mit Hilfe von StyleSheet.create definiert, um das Aussehen des Containers und der Diagramme anzupassen.

chartConfig: Konfiguration für das Aussehen der Diagramme, einschließlich Farbverläufen, Farben und Strichstärken.

Profile Page:

Hier haben wir eine tabellarische Ansicht zu den persönlichen Daten. Hier wird folgendes angezeigt. Name, Grösse, Gewicht, Alter. Hierfür wurde ein listContainer verwendet. Weiter unten haben wir eine View, welche als btnContainer implementiert ist. Hier sind die Buttons «Meine Ziele», «Tracker verbinden» und «Datenschutzrichtlinien. Unterhalb gibt es einen listContainer um sich abzumelden oder das Konto zu löschen. Hier wurden eigentlich 3 Views in einer grossen View (Maincontainer) verschachtelt. Jeder dieser drei untergeordneten Views enthalten ihre eigentlichen Komponenten, welche wiederum untergeordnet sind.

Testing

|  |  |
| --- | --- |
| **Abschnitt** | **Inhalt** |
| ID | ST-01 |
| Anforderungen | Schritte werden im Startbildschirm angezeigt |
| Vorbedingungen | Sensor muss funktionieren und die Anzahl Schritte muss angezeigt werden. |
| Ablauf | App auf Startbildschirm öffnen |
| Erwartetes Resultat | Eine Zahl welche die Anzahl Schritte anzeigt soll dargestellt werden. |

|  |  |
| --- | --- |
| **Abschnitt** | **Inhalt** |
| ID | ST-02 |
| Anforderungen | Im Kalender wird angezeigt, an welchem Tag wie viele Schritte zurückgelegt wurden. |
| Vorbedingungen | Der Benutzer muss an Tagen in der Vergangenheit Schritte zurückgelegt haben. |
| Ablauf | 2. Seite mit Kalender öffnen und Werte analysieren und vergleichen. |
| Erwartetes Resultat | Ein Trend mit Übersicht zu den zurückgelegten Schritten an vergangenen Tagen. |

|  |  |
| --- | --- |
| **Abschnitt** | **Inhalt** |
| ID | ST-03 |
| Anforderungen | Verbrennte Kalorien werden im Startbildschirm angezeigt |
| Vorbedingungen | Es müssen Schritte zurückgelegt worden sein und oberhalb angezeigt werden, |
| Ablauf | Kalorien mit den Schritten analysieren und vergleichen |
| Erwartetes Resultat | Kalorien werden angezeigt und stehen in einem Verhältnis zu den zurückgelegten Schritten. |
| Abschnitt | Inhalt |
| ID | ST-04 |
| Anforderungen | Im Kalender werden Tage, an denen das Tagesziel erreicht ist, sollen grün angezeigt werden. Tage, an denen das Tagesziel nicht erreicht worden ist, sollen rot angezeigt werden. |
| Vorbedingungen | Ein Tagesziel muss hinterlegt sein. |
| Ablauf | Zurückgelegte Schritte mit Tagesziel vergleichen und Kalender analyisiern |
| Erwartetes Resultat | Ist das Tagesziel erreicht, soll der Eintrag im Kalender grün sein. Ist das Tagesziel nicht erreicht, soll der Eintrag im Kalender rot sein. |

Testprotokoll

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Person / Datum | Erfolgreich | Bemerkungen |
| ST-01 | Skala Larissa 09.02.2024 | Ja | Der Test war erfolgreich, die Schritte konnten angezeigt werden. |
| ST-02 | Matt Fabio  10.02.2024 | Ja | Der Test war teilweise erfolgreich, der Kalender zeigt die Schritte an |
| ST-03 | Rüegg Gregor  11.02.2024 | Ja | Der Test war erfolgreich, die verbrennten Kalorien werden angezeigt. |
| ST-04 | Matt Fabio  11.02.2024 | Ja | Der Test war erfolgreich, das Tagesziel wird richtig angezeigt. |

fazit

Fabio Matt:

* Was lief gut/schlecht?

Die Aufgabenaufteilung war vom Thema her gut. Jeder machte vor allem das, was er am besten konnte, trotzdem hatte jeder an allem gearbeitet. Gregor hat sich auf das Backend fokussiert, Larissa das Design entworfen und ich die Dokumentation + Tests erarbeitet und teilweise durchgeführt. Wir hatten zudem sehr effizient gearbeitet. Bei uns hatte aber zeitweise nicht alles direkt funktioniert. Vor allem das Routing hat uns zu viel Zeit gekostet. Die Zeit war für so ein grosses Projekt aber nicht genügend, so dass wir einiges am Wochenende erledigen mussten,

* Wie seid ihr mit dem Endergebnis zufrieden?

Wenn man beachtet, dass ich noch nicht oft mit React Native gearbeitet hatte, bin ich zufrieden. Auch wenn die App nicht die Beste ist, war der Lerneffekt da. Der Code hält sich an die Clean Code Prinzipien, das Design ist in Ordnung und die Dokumentation beinhaltet die Informationen, welche sie braucht.

* Was habt ihr gelernt?

Ich hatte das erste mal an einer Mobile Applikation gearbeitet. Bisher hatte ich nur mit React gearbeitet. Die Vorgaben für Mockups (Abstände, Grössen etc.) waren ebenfalls neu für mich. Das Arbeiten mit Sensoren war ebenfalls interessant. Ich hatte zudem auch das erste mal mit Android Studio gearbeitet.

* War alles vorhanden oder was fehlte noch?

Schlussendlich war nicht ganz alles vorhanden.

Larissa Skala:

* Was lief gut/schlecht?

Gut lief das Designen in Adobe XD. Die Umsetzung des Projektes war eine schwierig, da man von uns erwartet, in 2 Tagen eine neue Programmiersprache zu beherrschen und in zwei Tagen eine komplette Mobile App auf die Beine zu stellen. Dazu kommen noch nicht passende Anleitungen, Falschinformationen seitens Kursleitung (zB Library xy funktioniere, ist aber unbrauchbar etc.)

* Wie seid ihr mit dem Endergebnis zufrieden?

Design ist ansehlich

* Was habt ihr gelernt?

Grundlagen von React native.

* War alles vorhanden oder was fehlte noch?

Das Design stimmt nicht, Funktionalität (stand 12.02.2024, 10:18) ist nicht vorhanden, Pages werden nicht gerendert

Gregor Rüegg::

* Was lief gut/schlecht?

Am Anfang hat vieles nicht Funktioniert wie z.B. der Emulator vom Handy. Als sich das ganze geklärt hat und man sich eingearbeitet hat ging eigentlich alles recht gut.

* Wie seid ihr mit dem Endergebnis zufrieden?

An sich bin ich ganz zufrieden.

* Was habt ihr gelernt?

React-Native kennengelernt und ein paar Sachen im Zusammenhang mit Apps wie z.B. das Publishing und die ganzen Anforderungen.

* War alles vorhanden oder was fehlte noch?

Mehr oder weniger ist alles Vorhanden