Meine App

## Fifa Tracker

### Version 1.0.0, Datum: 12. März 2023

A screenshot of a phone

Description automatically generated

Abstract (Kurzbeschreibung)

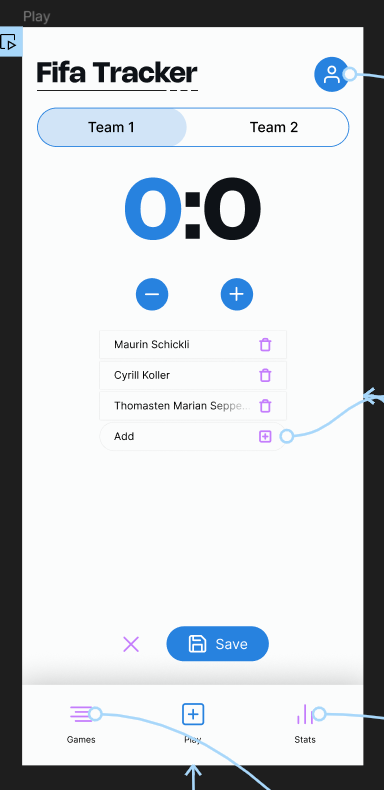
Die Anwendung Fifa Tracker dient zum Verfolgen von Fifa-Spielen. Sie verfügt über die Funktionen Spieler anlegen, Spiele verwalten, Spiele erstellen und Statistiken anzeigen. Statistiken sind Spiele gespielt, Spieler mit den meisten Siegen etc. Alle Daten werden in der Cloud gespeichert. Zu einem späteren Zeitpunkt soll noch eine Authentifizierung eingeführt werden.

user stories

An dieser Stelle alle Anforderungen als User Stories auflisten:

* Als Benutzer möchte ich Spiele mit Punktestand eingeben können, damit ich sehen kann, wer von meinen Kollegen der beste FIFA-Spieler ist.
* Als Benutzer möchte ich Spieler erfassen können, um sie zu Spielen hinzuzufügen.
* Als Benutzer möchte ich falsch erfasste Spiele löschen können um immer korrekte Daten zu haben.
* Als Benutzer möchte ich die Spielhistorie einsehen können, um mir "Historische" Spiele anzusehen.
* Als Benutzer möchte ich nach Spielern suchen können, um einfacher und schneller Spieler hinzufügen zu können.
* Als Benutzer, sollte ich nicht in der Lage sein, einen Spieler ohne Namen zu erstellen.

Mockups

An dieser Stelle den Prototypen deiner App beschreiben. Füge einen Printscreen deiner Screens ein und beschreibe jeden Screen einzeln.

Der Play Screen ist der erste Screen, den der Benutzer sieht, wenn er die Anwendung öffnet. In dieser Ansicht kann direkt ein neues Spiel erstellt werden. Rechts oben können die Spieler verwaltet werden.

Mit den beiden Reitern Team 1 und Team 2 können Spieler zu den Teams hinzugefügt werden. Mit dem Add Button gelangt man in ein Modal wo man neue Spieler hinzufügen kann.

A screenshot of a cell phone

Description automatically generated Wenn man auf «Add» klickt, kommt man auf diese Seite, wo man die Spieler pro Mannschaft verwalten kann. Man hat eine Suche zur Verfügung und sieht auch die bereits ausgewählten Spieler. Mit einem Klick auf den «Plus» Button kann ein Spieler zu einer Mannschaft hinzugefügt werden.

**A screenshot of a phone

Description automatically generated**Durch das klicken auf das «Person> Icon kommt man auf ein Modal auf dem alle Spieler verwaltet werden können. Hier können Spieler gesucht und gelöscht werden. Gleichzeitig können auch Spieler hinzugefügt werden.

A screenshot of a phone

Description automatically generatedÜber die untere Navigation gelangt man zum Games Screen. Hier kann man nach vergangenen Spielen suchen und diese auch löschen. Außerdem sieht man, wer in diesem Team gespielt hat und wann das Spiel stattgefunden hat.

A screenshot of a phone

Description automatically generatedEin weiterer Klick in der Bottom Navigation führt zum Reiter Stats. Hier werden die aktuellen Statistiken angezeigt.

Technische Realisierung

An dieser Stelle beschreiben, wie die technische Realisierung der App erfolgt ist. Welche Komponenten gibt es und wie werden die Daten gespeichert resp. von wo werden diese geladen. Wenn es sich um komplizierte Abläufe handelt, werden diese am besten mit UML Diagrammen beschrieben.

Eine Fachperson, die das Kapitel liest, sollte schnell nachvollziehen können, wie die App realisiert wurde.

## A diagram of a serviceÜbersicht

Unsere App besteht aus einer Bottom Navigation von der aus man auf die drei Hauptseiten gelangen kann. Die verschiedenen Tabs beziehen ihre Daten von unseren Services die dann Funktionen implementiert haben um auf die DB zuzugreifen. Dadurch haben wir Präsentation Logik und DB Logik getrennt obwohl wir kein «richtiges» Backend haben.

## Play

Im Play Tab verwenden wir mehrheitlich selber gemachte Components, wie TabButton, Result, Teamselection. Ausserdem verwenden wir einen selbst gemachten Button Component mit verschiedenen Varianten. Der TabButton brauchen wir, um zwischen den zwei Teams zu wechseln, um dann die jeweilige Toranzahl und Mitspieler auszuwählen. Der Result Component zeigt das Ergebnis des Spiels an und der Teamselection Component, dient dazu die Spieler einem Team zuzuteilen.

## History

Die index.tsx-Datei im history-Ordner ist eine React-Komponente, die eine Liste von Spielen darstellt. Sie nutzt den React-State und useFocusEffect-Hook, um Spiele zu verwalten und anzuzeigen. Beim Start werden games, eine Liste der Spiele, und isFetching, ein Indikator für Datenabruf, initialisiert. gamesService ist eine Instanz des GamesService, der Interaktionen mit einer Spiele-Datenquelle ermöglicht.

Die Funktion updateGames ruft Spiele ab und aktualisiert den Spielzustand. Sie setzt isFetching zuerst auf true, um laufenden Abruf anzuzeigen, ruft dann getGames vom gamesService ab. Bei Erfolg werden Spiele im Zustand gespeichert, bei Fehler wird eine Toast-Nachricht angezeigt. Danach wird isFetching auf false gesetzt.

Die useFocusEffect-Hook ruft updateGames beim ersten Rendern auf, um Spiele zu laden und anzeigen.

Die Render-Methode zeigt entweder eine Nachricht für keine Spiele oder eine FlatList-Komponente für Spiele. FlatList nutzt games als Datenquelle und updateGames als Aktualisierungs-Handler. Jedes Spiel wird als GameDetail-Komponente dargestellt, die Spielinformationen zeigt.

## Statistics

Der Statistics Tab ist der einfachste der drei. Er hat eine Funktion getStats, die alle Statistiken vom StatsService holt und diese in einen useState schreibt. Gerendert werden 3 StatsContainer und einen Button. Die StatsContainer sind selbstgemachte Container, die die verschiedenen Statistiken anzeigen, die in der oben beschriebenen Funktion geholt werden. Der Button wird verwendet, um die Statistiken zu aktualisieren, falls sie dies nicht von alleine getan haben.

## Data Services

Es gibt drei Services, von denen es jeweils immer nur eine Instanz gibt, da sie als Singleton definiert sind. Die drei Services sind GamesService, StatsService und PlayerService. Diese Services verwalten jeweils alle Games beziehungsweise alle Player und Stats. Dazu gehört eine Suche nach IDs, Erstellung beziehungsweise Löschung eines Objekts und die Aktualisierung eines Objekts.

Testing

In diesem Kapitel definiert ihr die manuellen Tests, die durchgeführt werden. Es müssen mind. 4 manuelle Tests definiert werden.

Hier ein Beispiel eines Tests:

|  |  |  |
| --- | --- | --- |
| **Abschnitt** | **Inhalt** | |
| ID | 0 | |
| Anforderungen | * Als Benutzer möchte ich falsch erfasste Spiele löschen können um immer korrekte Daten zu haben. | |
| Vorbedingungen | Aktuellste Version der App, Expo Go und Internetverbindung. Bereits ein erfassten Spieler. | |
| Ablauf | * Klicke auf das «Person» icon oben rechts * Klicke auf das «trash» Icon | |
| Erwartetes Resultat | Der Spieler sollte nun nicht mehr verfügbar sein. Man kann ihn nicht mehr zu Spielen hinzufügen. | |
| Abschnitt | | Inhalt |
| ID | | 1 |
| Anforderungen | | * Als Benutzer möchte ich Spiele mit Punktestand eingeben können, damit ich sehen kann, wer von meinen Kollegen der beste FIFA-Spieler ist. |
| Vorbedingungen | | Aktuellste Version der App, Expo Go und Internetverbindung. |
| Ablauf | | * Öffnen der App * Öffnen des «Add Player» Modal * Hinzufügen eines beliebigen Spielers * Switchen zu Team 2 * Hinzufügen eines beliebigen Spielers * Save |
| Erwartetes Resultat | | Im Tab «History» sollte jetzt das neu erfasst Spiel mit den ausgewählten Spielern zu sehen sein. |
| Abschnitt | | Inhalt |
| ID | | 2 |
| Anforderungen | | * Als Benutzer möchte ich Spieler erfassen können, um sie zu Spielen hinzuzufügen. |
| Vorbedingungen | | Aktuellste Version der App, Expo Go und Internetverbindung. |
| Ablauf | | * Öffnen der App * Klick oben rechts auf das Person Icon * Eingabe des Spieler Namens im Inputfeld * Drücken auf das Plus links im Inputfeld |
| Erwartetes Resultat | | Es sollte ein neuer Spieler mit diesem Namen hinzugefügt worden sein. Der Spieler sollte nun verwendet werden können um Spiele zu erstellen. |

|  |  |
| --- | --- |
| **Abschnitt** | **Inhalt** |
| ID | 3 |
| Anforderungen | * Als Benutzer, sollte ich nicht in der Lage sein, einen Spieler ohne Namen zu erstellen. |
| Vorbedingungen | Aktuellste Version der App, Expo Go und Internetverbindung. |
| Ablauf | * Öffnen der App * Klick oben rechts auf das Person Icon * Drücken auf das Plus links im Inputfeld ohne etwas im Feld eingegeben zu haben |
| Erwartetes Resultat | Es sollte kein Spieler erstellt werden, da kein Name eingegeben worden ist. |

Testprotokoll

In diesem Kapitel wird die Testausführung protokolliert. Welche Tests wurden wann von wem mit welchem Resultat durchgeführt.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Person / Datum | Erfolgreich | Bemerkungen |
| ST-00 | M. Schickli 12.03.2024 | Ja | Der Testfall war erfolgreich. |
| ST-01 | C.Koller  12.03.2024 | Ja | Der Testfall war erfolgreich. |
| ST-02 | Damjan  12.03.2024 | Ja | Der Testfall war erfolgreich. |
| ST-04 | Maurin  12.03.2024 | Ja | Der Testfall war erfolgreich. |

fazit

## Maurin

Ich fand den Fifa-Tracker ein sehr lehrreiches Projekt. Die Arbeit mit React Native hat mir sehr viel Spass gemacht. Wir hatten einen guten Zeitplan und konnten viele Features implementieren. Außerdem fand ich den Aufbau des Kurses sehr interessant.

## Cyrill

Ich fand die Projektphase sehr cool und interessant, da ich sehr viel neues dadurch gelernt habe, dass ich es direkt selbst implementieren musste. Wir haben unsere Zeit gut eingeteilt und die Teamdynamik war super. Ich weiss nun, dass mir Mobile App Development sehr viel Spass bereitet.