Dokumentation

# Testprotokoll

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifikation** | **Vorbedingungen** | **Vorgehen beim Test** | **Erwartetes Resultat** |
| JSON importieren | Vorhandenes JSON mit vorgegebener Struktur & ein funktionierender Import | Ich werde den Pfad des erstellten JSON’s angeben und das JSON mit einem «console.log()» überprüfen. | Das JSON wird korrekt dargestellt. Keine Fehler. |
| Im Menu Kategorien sehen | JSON muss importiert worden sein & JSON braucht mind. einen Datensatz für eine Kategorie | JSON importieren, App starten und App-Verhalten überprüfen | Die App stellt die Kategorien richtig dar. |
| Eine Kategorie auswählen können | JSON muss importiert worden sein & JSON braucht mind. einen vollständigen Datensatz für eine Kategorie | JSON importieren, App starten und App-Verhalten überprüfen | Die App sollte mich auf eine weitere Seite leiten und mir alle Nahrungsmittel dieser Kategorie anzeigen. |
| Ein Nahrungsmittel auswählen und darstellen lassen können | JSON muss importiert worden sein & JSON braucht mind. einen vollständigen Datensatz einer Kategorie | JSON importieren, App starten und App-Verhalten überprüfen | Die App stellt die Attribute (Healthy, Energy & Macros) des Nahrungsmittels korrekt dar. |
| Funktionierender «Zurück» - Knopf | App muss funktionstüchtig sein, Knopf hat Referenz auf Seite & Knopf wird dargestellt für Benutzer | App starten, ein «Zurück»-Knopf Instanz suchen und draufklicken | Knopf leitet wie erwartet weiter auf nächste Seite. |

# Reflexion

In diesem ÜK konnte ich meine Kompetenz um einiges erweitern, wenn es um die Webtechnologie und das Programmieren einer App geht (Zwei Fliegen mit einer Klatsche!).  
Am Anfang hatte ich Probleme mich an das neue Framework zu gewöhnen, denn bisher hatte ich nur wenig Kontakt mit dem JavaScript Framework «Angular» und dem «Ionic»-Framework für das Erstellen einer «hybrid App».  
Nach etwa 2 Wochen nach Start des ÜK’s war ich in der Lage mit dem Framework sauber zu arbeiten und Änderungen sauber zu implementieren.  
Erst nachdem ich wirklich mit den beiden Frameworks nun korrekt arbeiten konnte, bekam ich einen Energieschub und arbeitete auf das fertige Produkt hin.   
Ich arbeitete an Wochenenden und auch unter der Woche, nicht weil der ÜK-Leiter es von uns verlangt hätte, aber weil ich mich bei so einer Gelegenheit nicht einfach zurücklehnen, sondern wirklich alles Wissen aufsaugen möchte.  
Die Zeit, die ich in meiner Freizeit aufgewendet habe, um am Projekt zu arbeiten war keine vergeudete Zeit, denn schlussendlich hatte ich es geschafft meine App zu realisieren und die Funktionalität zu garantieren, die ich am Anfang des ÜK’s in meinem Konzept beschrieben habe.

Leider hatten wir nur fünf ÜK-Tage, denn die App an sich war noch nicht ganz fertig. Die Funktionalität war zwar da, doch die Feinschliffe fehlten noch. Meistens sind es immer die Feinschliffe, die das Produkt dann wirklich glänzen lassen.  
Die App mag auf den ersten Blick primitiv sein, denn sie stellt ja nur ein paar Nahrungsmittel dar und deren «Werte», doch für mich als neuer Entwickler war dies alles andere als ein Spaziergang.

Die Funktionalität der App ist meiner Meinung nach wirklich cool, vom Import des JSON’s bis hin zur flexiblen Datendarstellung der Werte von den einzelnen Nahrungsmitteln.  
Mit der Hilfe des ÜK-Leiters konnte ich mich rasch verbessern und an der App so effizient wie möglich arbeiten. Der Leiter gab uns immer wieder nützlichen Input (Ideen), die die Funktionalität der App verbessern und die App gleichzeitig auch «nachhaltiger» gestalten würde («Easier maintenance»).

Rückblickend war der ÜK aus meiner Sicht ein voller Erfolg für meine Ausbildung als Applikationsentwickler!

# Vorbereitung zur Veröffentlichung

**Android**

1. «Google Play Developer Console» Konto erstellen
2. Das Developer Konto mit «Google Wallet» verlinken (App Käufe, …)
3. Im «Google Play Developer Console» eine neue Applikation hinzufügen
4. Details der App ausfüllen (Preis, Altersfreigabe, Kategorie, …)
5. Die App hochladen (App Bundle oder .apk)
6. Weitere App-Details festlegen (Geldverteilung, kleine Kontrolle von Google Play Store, …)
7. Und auf «Veröffentlichen» klicken!

**Apple**

1. «Apple Developer Program» Konto erstellen
2. App für Kontrolle vorbereiten (Beschreibung der App, Altersfreigabe, Preis, Bilder, …)
3. Eine Apple Store «Produktseite» erstellen (Titel des Produkts, Beschreibung, Screenshots, …) mit «App Store Connect»
4. App hochladen mit Hilfe von «Xcode»
5. App zur Kontrolle abschicken («Xcode»)

# Installationsanleitung

Für die Installation der App benötigen wir folgende Punkte:

* Ein passendes IDE (empfohlen wird «VS Code 2022»)
* Die Node.js Library
* Den Ionic CLI
* Zugang zum Repository

Das wären alle Punkte die Ihr benötigen würdet, um bei der Entwicklung der App mitzumachen.

## Download des IDE’s

Hier könnt ihr mein empfohlenes IDE herunterladen.

Link: <https://code.visualstudio.com/download>

## Zum installieren des Ionic CLI & des Node.js

Bitte folgt der offiziellen und super einfachen Anleitung vom Ionic Framework selbst!  
Sie werden euch instruieren, wie ihr das Ionic CLI und auch noch das Node.js installieren könnt.

Anleitung (Ionic CLI): <https://ionicframework.com/docs/intro/cli>

Anleitung (Node.js): <https://ionicframework.com/docs/intro/environment>

## Zugang zum Repository

Da mein Repository auf «public» gestellt ist und ich die Voreinstellungen so angepasst habe, dass jeder darauf zugreifen und Änderungen vornehmen kann, sollte die zukünftige Zusammenarbeit leichter fallen.

Link zum Repo: <https://github.com/SirFelge/health-eat>