Meine App

## QuickScribble

### Version 1.0.0, Datum: 05. Juli 2024

Ein Bild, das Text, Screenshot, Software, Betriebssystem enthält.

Automatisch generierte Beschreibung

Abstract (Kurzbeschreibung)

QuickScribble ist eine benutzerfreundliche Notiz-App, die darauf abzielt, das Erstellen und Verwalten von Notizen so einfach wie möglich zu gestalten. Beim Start der App wird der Nutzer auf einer übersichtlichen Startseite begrüßt, wo er wählen kann, ob er bereits vorhandene Notizen ansehen oder neue erstellen möchte. Die App bietet vollständige CRUD-Funktionalitäten (Create, Read, Update, Delete), wodurch Nutzer ihre Notizen mühelos hinzufügen, anzeigen, bearbeiten und löschen können. Ein besonderes Merkmal von QuickScribble ist die Persistenz der Daten: Alle Notizen werden dauerhaft gespeichert. Das bedeutet, dass sie auch nach dem Beenden und Neustarten der App weiterhin verfügbar sind. Dies stellt sicher, dass keine wichtigen Informationen verloren gehen und der Nutzer jederzeit Zugriff auf seine Notizen hat. QuickScribble ist die ideale Lösung für alle, die ihre Gedanken und Ideen unkompliziert und zuverlässig festhalten möchten.

user stories

An dieser Stelle alle Anforderungen als User Stories auflisten:

* Als User möchte ich eine Notiz erfassen, damit ich mir wichtige Dinge merken kann.
* Als User möchte ich meine Notizen ansehen können, damit ich noch weiss, was notiert wurde.
* Als User möchte ich meine Notizen löschen können, damit ich nutzlose Notizen löschen kann.
* Als User möchte ich meine Notizen bearbeiten können, damit ich mögliche Fehler nachtragend bearbeiten kann.
* Als User möchte ich meinen Notizen Bilder anfügen können, damit ich wichtige Bilder ebenfalls speichern / notieren kann. (feature wenn alles andere steht)

Mockups

Ein Bild, das Text, Handy, Screenshot, Kommunikationsgerät enthält.

Automatisch generierte Beschreibung

Screen (oben links): Das ist der Hauptbildschirm unserer Applikation. Dort sieht man seine erstellten Notizen und kann bei bedarf für mehr Informationen noch draufklicken. In unserer App haben wir nachträglich noch eine Favorisierung- und lösch Funktion eingebaut.

Screen (oben 2. Von links): Das ist der Screen mit der Suchfunktion, bei der wir uns noch nicht ganz sicher sind, ob wir diese umsetzen werden, da vielleicht einfach die Zeit nicht reicht. Aber das Grundprinzip ist, das man einfach nach einer Notiz suchen kann. Das hilft der Organisation, wenn man viele Notizen hat.

Screen (oben 2. Von rechts): Dieser Screen dient dem Erstellen einer Notiz. Und ist eigentlich recht selbsterklärend. Man kann einen Titel und die eigentliche Notiz eingeben und diese dann speichern.

Screen (oben rechts): Dieser Screen zeigt die geplante Funktion das man ein Foto hochladen kann. Jedoch wird es noch eine Frage der verbleibenden Zeit, welche beantwortet, ob diese Funktion effektiv in den Final Release kommt oder dann ein späteres Update werden soll.

Screen (unten links): Dieser Screen zeigt die Detailansicht einer Notiz an. Wenn man auf den Titel einer Notiz klickt, kommt man dann auf die Detailansicht, in welcher man die ganze Notiz anschauen kann.

Screen (unten rechts): Dieser Screen zeigt die Bestätigung, welcher ein User geben muss, wenn er eine Notiz löschen möchte, ob es diese Funktion dann auch wirklich in den Final Release schafft, werden wir noch sehen. Bisher geht es auch ohne.

Technische Realisierung

Unsere Notizen-Applikation wurde mit modernen Webtechnologien und Frameworks entwickelt, um eine reaktionsschnelle und benutzerfreundliche Benutzeroberfläche zu bieten. Hier sind die Hauptkomponenten des Technologie-Stacks:

1. **React Native**: Ein Framework für die plattformübergreifende Entwicklung, das es ermöglicht, native Mobilanwendungen mit JavaScript und React zu erstellen. Dies bietet eine hervorragende Leistung und Benutzererfahrung sowohl auf Android- als auch auf iOS-Geräten.
2. **Expo**: Ein Open-Source-Toolchain, das den Entwicklungsprozess für React Native-Anwendungen beschleunigt. Expo bietet eine Vielzahl von Tools und Services, die die Entwicklung, das Testen und das Deployment erleichtern.
3. **AsyncStorage**: Ein einfacher, unverschlüsselter, asynchroner, dauerhafter Key-Value-Speicher, der von React Native bereitgestellt wird. Wir verwenden es zur Speicherung von Notizdaten lokal auf dem Gerät des Benutzers.
4. **React Navigation**: Ein flexibles und deklaratives Navigations-Framework für React Native. Es ermöglicht eine einfache Implementierung von Navigationsflüssen innerhalb der App.

### Datenmodell und Speicherung

Unsere Anwendung speichert Notizen lokal auf dem Gerät des Benutzers, um eine Offline-Verfügbarkeit zu gewährleisten. Jede Notiz besteht aus einem Titel, einem Text, einem Status (erledigt oder nicht), einem Favoritenstatus und einer eindeutigen ID, die beim Erstellen generiert wird. Das Datenmodell für eine Notiz sieht wie folgt aus:

{

    id: string,

    title: string,

    text: string,

    favourited: boolean

}

Wir verwenden «AsyncStorage», um diese Notizen zu speichern und abzurufen. Dies ermöglicht eine persistente Speicherung, die auch nach dem Schließen und erneuten Öffnen der App erhalten bleibt.

**Wichtige Komponente**

#### InputField

Das InputField-Komponent bietet ein anpassbares Eingabefeld für den Titel und den Text der Notiz. Es wurde so gestaltet, dass es groß genug ist, um bequem darin zu schreiben, ähnlich wie bei traditionellen Notiz-Apps.

import {ScrollView, StyleSheet, TextInput, } from "react-native"

export default function InputField({placeholder, value, onChangeText}) {

    return (

        <ScrollView style={styles.container}>

            <TextInput

                style={styles.input}

                value={value}

                onChangeText={onChangeText}

                placeholder={placeholder}

                multiline={true}

                textAlignVertical="top"

            />

        </ScrollView>

    );

}

const styles = StyleSheet.create({

    container: {

        width: '90%',

        marginBottom: 15,

        alignSelf: 'center',

    },

    input: {

        height: 650,

        width: 420,

        //marginTop: 18,

        //marginBottom: 1,

        fontSize: 18,

        borderWidth: 1,

        borderRadius: 10,

        padding: 10,

        backgroundColor: '#f9f9f9',

    },

})

### Fazit

Die technische Realisierung dieser Notizen-Applikation zielt darauf ab, eine benutzerfreundliche und reaktionsschnelle Erfahrung für die Benutzer zu bieten. Durch die Verwendung von React Native und Expo konnten wir eine plattformübergreifende App erstellen, die auf sowohl Android- als auch iOS-Geräten hervorragend funktioniert. AsyncStorage ermöglicht es uns, die Notizen lokal zu speichern, was eine nahtlose Offline-Erfahrung sicherstellt. Insgesamt stellt die Anwendung eine robuste und flexible Lösung für die Verwaltung von Notizen dar.

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 | Testfallnummer (ST = Systemtest) |
| Anforderungen | Welche Anforderungen werden durch diesen Testfall abgedeckt. (User Stories) |
| Vorbedingungen | Was muss gegeben sein, damit dieser Test durchgeführt werden kann? |
| Ablauf | Welche Schritte werden bei der Durchführung des Tests durchlaufen? |
| Erwartetes Resultat | Was sollte nun passiert sein? |

#### Testfall 1: Neue Notiz erstellen

**Ziel**: Überprüfen, ob eine neue Notiz erstellt und gespeichert werden kann.

1. **Testschritt**: Klicke auf das Eingabefeld für den Titel und gib einen Titel ein.
   * **Erwartetes Ergebnis**: Der eingegebene Titel wird im Feld angezeigt.
2. **Testschritt**: Klicke auf das Eingabefeld für die Notiz und gib den Inhalt ein.
   * **Erwartetes Ergebnis**: Der eingegebene Inhalt wird im Feld angezeigt.
3. **Testschritt**: Klicke auf die Schaltfläche "Save".
   * **Erwartetes Ergebnis**: Die neue Notiz wird zur Liste hinzugefügt und in der Liste der Notizen angezeigt.

#### Testfall 2: Notizen anzeigen

**Ziel**: Überprüfen, ob alle gespeicherten Notizen korrekt angezeigt werden.

1. **Testschritt**: Öffne die Notizen-Applikation.
   * **Erwartetes Ergebnis**: Die Startseite der Applikation wird angezeigt.
2. **Testschritt**: Überprüfe die Liste der Notizen.
   * **Erwartetes Ergebnis**: Alle gespeicherten Notizen werden korrekt in der Liste angezeigt.

#### Testfall 3: Notiz favorisieren

**Ziel**: Überprüfen, ob eine Notiz als Favorit markiert und auf dem Favoriten-Tab angezeigt werden kann.

1. **Testschritt**: Wähle eine Notiz aus der Liste aus.
   * **Erwartetes Ergebnis**: Die ausgewählte Notiz wird hervorgehoben.
2. **Testschritt**: Klicke auf die Schaltfläche "Favorisieren" für die ausgewählte Notiz.
   * **Erwartetes Ergebnis**: Die Notiz wird als Favorit markiert (Stern-Icon wechselt den Zustand).
3. **Testschritt**: Wechsle zum Favoriten-Tab.
   * **Erwartetes Ergebnis**: Die favorisierte Notiz wird auf dem Favoriten-Tab angezeigt.

#### Testfall 4: Notiz löschen

**Ziel**: Überprüfen, ob eine Notiz gelöscht werden kann.

1. **Testschritt**: Wähle eine Notiz aus der Liste aus.
   * **Erwartetes Ergebnis**: Die ausgewählte Notiz wird hervorgehoben.
2. **Testschritt**: Klicke auf die Schaltfläche "Delete" für die ausgewählte Notiz.
   * **Erwartetes Ergebnis**: Die Notiz wird aus der Liste entfernt.

#### Testfall 5: Titel- und Notizfelder sind gross genug

**Ziel**: Überprüfen, ob die Eingabefelder für Titel und Notiz ausreichend groß sind.

1. **Testschritt**: Öffne die Seite zum Erstellen einer neuen Notiz.
   * **Erwartetes Ergebnis**: Die Eingabefelder für Titel und Notiz werden angezeigt.
2. **Testschritt**: Überprüfe die Größe der Eingabefelder.
   * **Erwartetes Ergebnis**: Die Felder sind groß genug, um bequem darin zu schreiben.

Testprotokoll

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

Hier ein Beispiel eines Eintrages:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Person / Datum | Erfolgreich | Bemerkungen |
| ST-01 | T. Brasser  05.07.2024 | Ja | Eine andere Tastatur wäre großartig da man im Moment noch keinen ö |
| ST-02 | S. Soares  05.07.2024 | Ja | Gutes UI-Design |
| ST-03 | D. Meister 05.07.2024 | Ja | Ok. |
| ST-04 | V. Witzmann 05.07.2024 | Ja | Funktioniert einwandfrei |
| ST-05 | T. Satheeswaran 05.07.2024 | Ja | Sehr gut |

Fazit

Hier kommt eure Reflexion zum Projekt.

* Was lief gut/schlecht?
  + Die Kommunikation im Team war sehr angenehm und einfach.
  + Die Umsetzung unseres Projektes war einfacher als gedacht und wir kamen gut mit der verfügbaren Zeit zurecht.
  + Die Ziele hätten wir erreichen können aber da wir ein paar Probleme hatten war für diese dann nicht mehr genügen Zeit
* Wie seid ihr mit dem Endergebnis zufrieden?
  + Wir sind recht zufrieden mit unserem Produkt. Wir denken das wir eine gute Arbeit geleistet haben, auch wenn die App vielleicht noch ein wenig ausbaufähig wäre
* Was habt ihr gelernt?
  + Wir haben gelernt, wie man eine Applikation mit React-native schreibt. Nebst dem haben wir mit den Inputs des Lehrers auch noch andere Wichtige Sachen gelernt
* War alles vorhanden oder was fehlte noch?
  + Ein paar zuvor gewünschte Features wurden zum Ende hin ein wenig vernachlässigt was dazu führte das diese nicht in den final release kommen werden. Aber alles in allem sind wir mit dem vorhandenen zufrieden