

Nochba – App für Nachbarschaftshilfe

DIPLOMARBEIT

verfasst im Rahmen der

Reife- und Diplomprüfung

an der

Höheren Abteilung für Informatik

Eingereicht von:

Arsham Edalatkhah
Sandin Habibovic
Martin Hausleitner

Betreuer:

Rainer Stropek

Projektpartner:

IKT Linz GmbH - Open Common Linz

Leonding, März 2023

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Die Arbeit wurde bisher in gleicher oder ähnlicher Weise keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Die vorliegende Diplomarbeit ist mit dem elektronisch übermittelten Textdokument identisch.

Leonding, April 2022

S. Schwammal & S. Schwammal

Abstract

Brief summary of our amazing work. In English. This is the only time we have to include a picture within the text. The picture should somehow represent your thesis. This is untypical for scientific work but required by the powers that are. Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.



Zusammenfassung

Zusammenfassung unserer genialen Arbeit. Auf Deutsch. Das ist das einzige Mal, dass eine Grafik in den Textfluss eingebunden wird. Die gewählte Grafik soll irgendwie eure Arbeit repräsentieren. Das ist ungewöhnlich für eine wissenschaftliche Arbeit aber eine Anforderung der Obrigkeit. *Bitte auf keinen Fall mit der Zusammenfassung verwechseln, die den Abschluss der Arbeit bildet!* Suspendisse vel felis.

Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.



Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Aufgabenstellung	1
1.3	Zusammenfassung des Projektergebnisse	1
1.4	Projektvorgehensmodell	1
2	Technologien	2
2.1	Technologieevaluierung	2
2.2	Fazit	4
3	Systemarchitektur	5
3.1	Flutter	5
3.2	Firebase	5
3.3	Algolia Search	5
4	Umsetzung	6
4.1	Continuous Integration/Delivery	7
4.2	Mobile Anwendung	7
4.3	UI/UX Design	7
4.4	Backend	7
5	Evaluating getroffener Entscheidungen	8
6	Zusammenfassung	9
	Literaturverzeichnis	VI
	Abbildungsverzeichnis	VII
	Tabellenverzeichnis	VIII

Quellcodeverzeichnis

IX

Anhang

X

1 Einleitung

1.1 Motivation

1.2 Aufgabenstellung

1.3 Zusammenfassung des Projektergebnisse

1.4 Projektvorgehensmodell

2 Technologien

2.1 Technologieevaluierung

Im Rahmen dieser Diplomarbeit wurden umfangreiche Anforderungen an das Frontend und Backend der Anwendung gestellt, um eine hochwertige und funktionale Anwendung zu entwickeln. Für das Frontend wurden iOS- und Android-Apps gefordert, wobei die Möglichkeit einer zukünftigen Web-App berücksichtigt werden sollte. Das Backend sollte eine Echtzeit-Datenbank für den Chat und eine Datenbank mit Geofunktionen für den Radios-Filter bereitstellen. Dabei war es wichtig, dass das Backend einfach zu implementieren ist und eine gute SDK-Unterstützung für das Frontend bietet.

Neben diesen technischen Anforderungen war auch die Umsetzung der Business-Logik ein wichtiger Aspekt. Hierbei wurden Aspekte wie die Registrierung und Verifizierung von Nutzern berücksichtigt, um eine sichere und benutzerfreundliche Anwendung zu gewährleisten. Darüber hinaus war es wichtig, dass die Anwendung mit neuen und effizienten Technologien entwickelt wird, um eine schnelle und agile Anpassung an die sich ändernden Anforderungen des Marktes und der Nutzer zu ermöglichen.

2.1.1 Frontend

Nach ausführlicher Recherche wurden die folgenden plattformübergreifenden Frameworks in Betracht gezogen[1]: Flutter, React Native, Ionic und Xamarin. Jede Plattform hat ihre Vor- und Nachteile.

Nach langer Recherche und Evaluierung aller Faktoren wurde Flutter als Plattform für das Frontend ausgewählt. Insbesondere die schnelle Entwicklung und das einfache Debugging sowie die Integration in VS Code waren entscheidende Faktoren. Jeder aus unserem Team hat eine eigene Flutter Test App programmiert [2] um die Entscheidung zu festigen.

Insgesamt waren wir mit unserer Entscheidung, Flutter zu verwenden, sehr zufrieden. Während der Entwicklungszeit stellte sich heraus, dass die Klammernverwendung in Flutter am Anfang etwas ungewohnt war, aber dank der hervorragenden Unterstützung

durch die Flutter-Community und der umfangreichen Dokumentation auf Stack Overflow konnten wir alle Herausforderungen bewältigen. Außerdem wurde die Entwicklung durch die ständigen Verbesserungen und Updates von Flutter, wie der Einführung der Impeller-Rendering-Engine, beschleunigt. [3]

Insgesamt war die Wahl von Flutter für das Frontend der App eine gute Entscheidung.

2.1.2 Backend

Für die Implementierung einer mobilen App in Flutter für die Diplomarbeit von Martin Hausleitner wurden verschiedene Backend-Technologien evaluiert. Dabei wurden folgende Anforderungen an die Technologie gestellt: Skalierbarkeit, Realtime-Datenbank für Chat, Business-Logik für den Beitragsradius, sichere Authentifizierung und Verschlüsselung sowie eine einfache und schnelle Entwicklungsmöglichkeit.

Die drei Favoriten für die Backend-Technologie waren Supabase [4], Firebase[?] und Appwrite[?]. Diese wurden alle zum damaligen Zeitpunkt (Februar 2022) mit einem Flutter-SDK unterstützt.

Alle drei Technologien bieten einen guten User-Authentifizierungs-Layer sowie eine Datenbank mit Geo-Queries für das Hauptfeature des Beitragsradius. Auch das Speichern von Beitragsfotos ist in allen Technologien möglich.

Jedoch hatte Supabase zu diesem Zeitpunkt keine Cloud Functions, was die anderen beiden Technologien hatten und somit konnte Supabase für die Umsetzung der Business-Logik, wie beispielsweise den Registrierungsprozess, nicht verwendet werden. Obwohl im April 2022 experimentelle Deno-Edge-Functions veröffentlicht wurden, war es in einer produktiven App nicht sinnvoll, diese zu nutzen.

Letztendlich blieben Firebase und Appwrite übrig. Da Firebase bessere Flutter-SDKs und mehr Features, insbesondere von Google für Flutter, hatte, wurde es als Backend-Technologie für die Diplomarbeit ausgewählt.

Eine Tabelle mit Vor- und Nachteilen beider Technologien ist wie folgt:

Insgesamt bietet Firebase aufgrund der umfangreichen Features und der einfacheren Integration mit anderen Google-Tools Vorteile gegenüber Appwrite. Allerdings kann App

Column 1	Column 2	Column 3
Row 1, Cell 1	Row 1, Cell 2	Row 1, Cell 3
Row 2, Cell 1	Row 2, Cell 2	Row 2, Cell 3
Row 3, Cell 1	Row 3, Cell 2	Row 3, Cell 3

Tabelle 1: Example Table

2.2 Fazit

Die Wahl von Flutter als Frontend-Plattform und Firebase als Backend-Plattform erwies sich als gute Entscheidung für die Anforderungen dieser Diplomarbeit. Die Einfachheit der Syntax, das schnelle Debugging und die umfangreiche Dokumentation und Unterstützung durch die Community machten Flutter zu einer einfachen und effizienten Plattform für die Entwicklung der mobilen Apps. Firebase bot eine gute Lösung für die Echtzeit-Datenbank und die Geodatenbank und eine einfache Integration mit Flutter.

Insgesamt haben die gewählten Technologien dazu beigetragen, dass die Entwicklung der Anwendung schnell und effizient verlief und die Anforderungen der Diplomarbeit erfüllt wurden.

3 Systemarchitektur

3.1 Flutter

3.2 Firebase

3.2.1 Firebase Authentication

Security Rules

3.2.2 Cloud Firestore

Datenmodel

3.2.3 Cloud Storage

3.2.4 Firebase Cloud Functions

3.3 Algolia Search

Firebase Cloud Function

4 Umsetzung

4.1 Continuous Integration/Delivery

4.1.1 GitHub Actions

4.1.2 Fastlane

4.1.3 Firebase App Distribution

4.2 Mobile Anwendung

4.2.1 Dateistruktur

4.2.2 State Management

GetX

4.2.3 Authentifizierung

Registrieren

Login

4.2.4 Feed

4.2.5 Beiträge

Kategorien

Info

Kommentare

Beitrag Melden

4.2.6 Filter

4.2.7 Suche

4.2.8 Chat

4.2.9 Profil

Profil Melden

4.2.10 Benachrichtigungen

5 Evaluating getroffener Entscheidungen

6 Zusammenfassung

Aufzählungen:

- Itemize Level 1
 - Itemize Level 2
 - Itemize Level 3 (vermeiden)
- 1. Enumerate Level 1
 - a. Enumerate Level 2
 - i. Enumerate Level 3 (vermeiden)

Desc Level 1

Desc Level 2 (vermeiden)

Desc Level 3 (vermeiden)

Literaturverzeichnis

- [1] S. Srivastava, „Top 10 Best Cross-Platform App Development Frameworks,” <https://appinventiv.com/blog/cross-platform-app-frameworks/>, February 2023.
- [2] M. Hausleitner, „Creating Test Apps with Flutter and Firebase,” <https://github.com/Martin-Hausleitner/flutter-firebase-test-01>, 2022.
- [3] Flutter, „Impeller,” <https://github.com/flutter/flutter/wiki/Impeller>, 2022.
- [4] „Supabase,” <https://supabase.io/>.

Abbildungsverzeichnis

Tabellenverzeichnis

1 Example Table 4

Quellcodeverzeichnis

Anhang