

Exposé/ Projektarbeit/ Seminararbeit/ Bachelorarbeit

Internationale Hochschule Duales Studium

Studiengang: Informatik

**Generative Programmierung mittels künstlicher Intelligenz**

Müller Korbinian

Matrikelnummer: 102302316

Adresse

Lohwald Straße 59

86356 Neusäß

Betreuer/in (bei Bachelorarbeiten): XX

Abgabedatum: 12.03.

Inhalt

[Einleitung 3](#_Toc174474509)

[Theoretische Fundierung 3](#_Toc174474510)

[Begriffs Erklärung 3](#_Toc174474511)

[IDE 3](#_Toc174474512)

[Unit Test 3](#_Toc174474513)

[Merge Request 3](#_Toc174474514)

[Methodik 3](#_Toc174474515)

[Software Anforderungen 3](#_Toc174474516)

[Fallstudie 3](#_Toc174474517)

[Implementation 4](#_Toc174474518)

[Durch Entwickler umgesetzt 4](#_Toc174474519)

[Generative KI 4](#_Toc174474520)

[Analyse 4](#_Toc174474521)

[Implikationen 4](#_Toc174474522)

[Fazit 4](#_Toc174474523)

Einleitung

Theoretische Fundierung

Begriffs Erklärung

IDE

Die Integrated Development Environment, kurz IDE, ist eine der Handwerks mittel. Dabei handelt es sich um einen Texteditor für Programme. Dabei ist der Hauptaugenmerk bei dem Verwalten von Großen Software Projekten. Für die Technische Umsetzung wird die DIE PHP Storm von Jet Brains. Es wird die Professionelle Version verwendet.

Unit Test

Bei Unit Tests wird Software pragmatisch und sorgfältig getestet. Dabei werden insbesondere Klassen und Funktionen getestet. Die zu testenden Funktionen werden mit bekannten und Konstanten Werten aufgerufen. Die Ergebnisse werden mit den zu Erwartenden verglichen. Wenn diese Übereinstimmen gilt der Unit Test als erfolgreich. Erst wenn alle Test erfolgreich durchlaufen gilt der Unit Test als erledigt.

Merge Request

Bei einem Merge Request handelt es sich um ein Bestandteil im Arbeitsablauf. Dabei wird im Git einen neuen Branch angelegt. Auf dem Neuen Branch werden nun die Änderungen vorgenommen. Sobald alles Umgesetzt ist wird der Branch auf den Git Server geladen. Nun wird ein Merge Request gestellt. Dadurch wird beantragt die Änderungen in den Hauptstand zu übernehmen. Dabei werden die Änderungen von einem Kollegen überprüft. Sobald alles in Ordnung ist wird der Merge Request akzeptiert und die Änderungen gehen in die Produktion

Methodik

Fallstudie

Im Folgenden wird die Fallstudie zunächst einmal genauer beschrieben.

Voraussetzung

Die Grundlage für die Fallstudie ist ein Kundenprojekt bei dem Praxispartner. Der Kunde ist ein großer Obstbauernverband in Deutschland. Dieser beauftragte eine digitale Mitgliederverwaltung. Die Kunden, Bauern, werden hiermit erfasst. Neben der Anschrift, Firmenname und Kontaktdaten werden auch die Flächen registriert. Zu den Flächen wird dokumentiert wo sich diese Befinden und welche Obstsorte angebaut wird. Anhand dieser Information kann der Bauer Dienstleistungen buchen. Der Beitritt zu dem Verband ist für die Bauern kostenpflichtig. Dazu kommt noch eine Monatliche Gebühr. Eine der Dienstleistungen ist die Beratung. Dabei wird ein Termin ausgemacht und ein Beraten kommt vorbei. Diese Beratungen können individuell basierend auf den Früchten oder Flächen geschehen. Um den Beratern die Arbeit zu erleichtern wurde beschlossen eine Handy App für diese zu Erstellen. Diese Handy App ist die Grundlage für die Fallstudie.

Die Handy-App basiert auf dem Java-Script Framework React Native. Die Entscheidung ist auf dieses Gefallen da es ein sehr Modernes Framework ist. Ein weiterer Grund ist die Entwicklungsgeschwindigkeit. Das Projekt wird zu Nativen Apps für Android und IOS kompiliert und ist als solche für die Berater zum Download bereitgestellt.   
In der App werden die relevanten Daten nicht gespeichert. Dies Passiert alles Zentral in der Mitgliederverwaltung. Dazu wurde die Mitgliederverwaltung um einen Rest-Service erweitert. Die Kommunikation zwischen der App und dem Backend, geschieht über verschlüsselte HTTPS Anfragen, mit dem Package Axios.

Die App ist einfach gehalten. Der Berater loggt sich ein und kommt zum Dashboard. Hier findet die Hauptnavigation statt. Von hier aus gelangt er zu seinem Profil, einer Übersicht über alle verfügbaren Beratungstätigkeiten und der Maske zum Erfassen und Dokumentieren seiner Tätigkeit. Hier wird erfasst bei welchem Bauer, zu welcher Uhrzeit er war. Für die Tätigkeit selber wird Dokumentiert was er gemacht hat und wie lange es gedauert hat.

Einer der Wichtigsten Bestandteile der App ist die Stundenübersicht. Hier sieht der Berater welche Tätigkeiten er geleistet hat. Diese sind nach dem Datum, absteigen sortiert. Es gibt hier die Möglichkeit die einzelnen Elemente zu Sortieren nach Beratungstyp, Fruchttyp oder nach Bauer. Diese nach Bauer sortiere Ansicht ist die Grundlage für die Fallstudie. Hier soll am Oberen Bildschirmrand eine Kachel hinzugefügt werden. Die Kachel wird benutzt um die Daten des Bauern darzustellen. Dazu zählen: Name, Anschrift, Alle Flächen und Alle Früchte die Angebaut werden. Ebenso sollen zwei Butten hinzugefügt werden, die die Telefonnummer und die E-Mail Adresse anzeigen. Wenn auf die Button gepresst wird soll sich das Entsprechende Programm öffnen und die Daten übernemen.

Software Anforderungen

Anhand dieser Projekt Konstellation gehen folgende Anforderungen an die App hervor.

* Die App soll Native auf Mobilgeräten laufen, hier kommen die Betriebssysteme Android und vor
* Als Programmiersprache wird Javascript in Verbindung mit dem Framework React Native eingesetzt
* Die Kommunikation findet über das Internet mittels HTTPS Request statt.
* Die Beratungsstunden App interagiert mit der Rest Service der Mitgliederverwaltung

Implementation

Durch Entwickler umgesetzt

Generative KI

Analyse

Implikationen

Fazit