

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

Software Anforderungen

Fallstudie

In der Fallstudie soll herausgefunden werden ob, die durch die Künstliche Intelligenz entwickelte Software funktioniert und technisch hohe Qualität aufweist.

Die erste Metrik die Ausgewertet wird ist die absolute Zeit die für die Lösung benötigt wird. Bei der KI Lösung ist der Ausgangspunkt ein Leerer Prompt. Die Zeit beginnt mit dem Eingeben der ersten Charakter in den Prompt. Auch die Manuelle Lösung beginnt mit einer Leeren Datei in der IDE. In beiden Fällen endet die Zeitnahme mit dem Erfolgreich Abgeschlossenen Unit Tests. Dieser Test ist nicht absolut und Akkurat. Jedoch liefert dieser bereits ein Erster Einblick in die Effektivität.   
Als sehr Effektiv wird das Resultat interpretiert wenn die Ki Zeit mehr als 25% Schneller war als die Manuell programmierte.

Die zweite Metrik ist die Ausführgeschwindigkeit. Dabei wird die Zeit gemessen die benötigt wird um die Komponente zu generieren. Dies geschieht ebenfalls in einem Unit Test. Jede der Implementationen wird mit drei verschiedenen Datensätzen aufgerufen. Der Test wird für jede Datensatz insgesamt 5 mal widerholt. Für jeden Daten Satz wird die Durchschnittliche Ausführzeit ermittelt. Der Test wird als Positiv erachtet, wenn die Zeiten, der durch die KI Implementierten Lösung, 10% schneller ist als die Menschlich Implementiertet.

Die dritte und letzte Metrik ist die qualitative Überprüfung. In dieser Prüfung werden die Beiden Code stücke einem Senior Entwickler in Form eines Merge Request vorgelegt. Es wird jedoch nicht erwähnt welches der beiden Programmstücke durch die KI entwickelt wurde. Der Merge Request wird gemeinsam durchgesprochen. Der Test wird als Erfolgreich angesehen, wenn der Senior Entwickler den Merge Request mit der KI Software Bevorzugt.

Implementation

Folgendes ist die Ausgangssituation für die Fallstudie. Der Kunde ist ein Obstbauernverband im Norddeutschland. Dieser hat eine online Mitgliederverwaltung in Auftrag gegeben. Dieses Projekt wurde in Typo3 mit PHP Umgesetzt. Nach erfolgreicher Lieferung wurde ein weiteres Projekt in Auftrag gegeben. Es wurde eine Handy App für die angestellten Berater beauftragt. Die App soll die geleisteten Beratungsstunden erfassen und verwalten.

**Der Projektaufbau**

Die Handy App besteht aus einem Frontend und einem Backend. Für die Datenspeicherung wird eine MariaDB verwendet. Diese wird auch von der Bestehenden Mitgliederverwaltung genutzt. Die Verwaltung wurde um einen Rest Service für die App erweitert. Dazu wird das Framework slim/http verwendet. Es werden Endpoints für das Einloggen in der App, das Bereitstellen von Nutzer Daten und das Verwalten von Beratungsstunden. Das Frontend basiert auf dem Javascript Framework React Native. Dieses ist eine Weiterentwicklung, des Frameworks React. Der größte Vorteil liegt darin, dass die Entwicklung schnell mittels Javascript und React Komponenten erledigt wird. Und im Anschluss in Native Android und IOS Apps übersetzt werden. Das Frontend nutz die sendet mittels Axios Fetch Requests an den Server um die Entsprechenden Daten zu Holen

**Die App Funktionalität**

Im Folgenden wird der Arbeitsablauf mit der App beschrieben. Es beginnt damit, dass ein Bauer eine Beratung stunde bei dem Verband bucht. Daraufhin findet ein Besuch durch den Berater stadt. Dabei wird der Bauer informiert und beratet zu den gemeldeten Flächen. Nach der Beratung geht es mit der App weiter. Der Berater logt sich ein. Dabei werden die Zugangsdaten an das Backend übermittelt und ausgewertet. Das Kernstück der App ist die Übersicht über alle geleisteten Beratungsstunden und die möglichkeit die gerate geleistete Beratungsstunden einzutragen. In der Übersicht mit den geleistenten Beratungsstunden ist es ferner möglich Nach dem Kunden, Berater, Zeitpunk, Beratungstätigket und angebaute Kulturen zu Filtern. Ebenso ist es möglich die Eigen Nutzerdaten zu bearbeiten.

Durch Entwickler umgesetzt

Generative KI

Analyse

Implikationen

Fazit