

## Team-Rollenbeschreibung

### Team-Rollenbeschreibung

Im Rahmen des Projekts „Hochschulbewerbung“ haben wir als vierköpfiges Team bestehend aus

- Luca Siemsen,
- Aran Götz,
- Omar Hamandi,
- Sehmus Ilhan

gemeinsam an der Umsetzung einer vollständigen Webanwendung gearbeitet.

Die Bearbeitung erfolgte arbeitsteilig, wobei sich die Aufgaben entlang individueller Interessen und Stärken verteilten. Das Frontend wurde hauptsächlich von Aran, Omar und Shemus gestaltet. Dazu gehörten die Entwicklung der Benutzeroberfläche mit Nuxt 3 und Vue.js, das Design der Formulare, die Navigation sowie das Einfügen von statischen Seiteninhalten wie FAQ und Startseite. Auch auf das responsive Design und eine benutzerfreundliche Darstellung wurde geachtet.

Luca übernahm die Umsetzung des Backends. Seine Aufgaben umfassten die Integration von Supabase für die Speicherung der Bewerbungs- und Kontaktdaten, das Anlegen der Datenbankstruktur, die Entwicklung der API-Endpunkte sowie die Umsetzung des Dateiuploads. Darüber hinaus verantwortete er den Login mit Rollenverwaltung und die serverseitige Validierung sowie den Versand automatischer E-Mails mit nodemailer.

Alle Teammitglieder waren regelmäßig in die Abstimmungen eingebunden, haben Ideen eingebracht und bei der Fehlersuche unterstützt. Die Zusammenarbeit erfolgte in engem Austausch, wodurch ein funktionsfähiges und technisch solides Ergebnis realisiert werden konnte.

## Beurteilung des KI-Einsatzes (ChatGPT)

### Beurteilung des KI-Einsatzes (ChatGPT)

Im Rahmen des Projekts wurde ChatGPT als unterstützendes Werkzeug bei der Entwicklung der Webanwendung "Hochschulbewerbung" eingesetzt. Die KI kam dabei in nahezu allen Projektphasen zum Einsatz – von der Ideenfindung und Erklärung technischer Zusammenhänge bis hin zur konkreten Code-Generierung.

Die Unterstützung durch die KI erwies sich insgesamt als sehr hilfreich. Besonders bei der Erklärung von Fehlermeldungen, der Umsetzung typischer Vue/Nuxt-Funktionalitäten oder dem Formulieren von Supabase-Anfragen konnte ChatGPT den Entwicklungsprozess beschleunigen. Auch die allgemeine Strukturierung der Anwendung, Vorschläge für API-Logik oder Mailversand mit nodemailer wurden auf Basis von KI-Eingaben erstellt und anschließend von Hand angepasst, erweitert und an die Projektanforderungen angepasst.

Allerdings zeigte sich auch, dass die KI in bestimmten Fällen wiederholend oder unpräzise antwortete, insbesondere bei komplexeren Fehlermeldungen oder Nuxt-spezifischen Konfigurationen. In solchen Situationen war zusätzliche Recherche über YouTube, Foren oder die offizielle Dokumentation erforderlich. Zudem war es notwendig, den generierten Code kritisch zu hinterfragen und anzupassen, da nicht alle Vorschläge direkt funktional oder optimal waren.

Besonders im Zusammenhang mit Rollen- und Rechteverwaltung über Supabase kam es häufiger zu unklaren oder widersprüchlichen Antworten. Die KI schlug zum Teil mehrfach dieselben Lösungsschritte vor, obwohl sie zuvor nicht funktioniert hatten. In einigen Fällen wurde eine Lösung vorgeschlagen, später aber widerrufen oder als ungünstig bezeichnet, nur um anschließend doch wieder darauf zurückzukommen. Dies führte zu unnötigem Zeitaufwand und zusätzlichem Debugging, insbesondere im Backend-Bereich.

Trotz dieser Schwierigkeiten war die KI eine sinnvolle Ergänzung – vor allem als „Sparringspartner“ für Ideen oder erste Lösungsvorschläge. Der endgültige Code entstand jedoch durch eigenes Verständnis, kritisches Prüfen und aktives Anpassen an das konkrete Projekt.

Insgesamt ist der Einsatz von ChatGPT als Unterstützung für Programmieraufgaben praxisnah und effizient, solange die Nutzung reflektiert erfolgt. Der generierte Code diente in vielen Fällen als produktive Grundlage, wurde jedoch stets von Hand überarbeitet und verstanden. Die KI war somit ein sinnvolles Hilfsmittel, ersetzte jedoch nicht das eigenständige Entwickeln und Problemlösen.