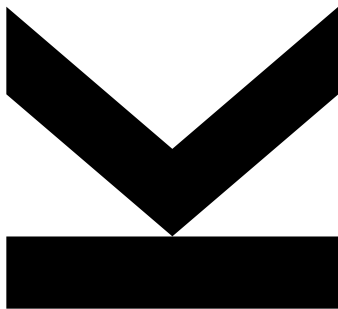


LERNVERTRAG - PROJEKTTITEL



Themenbereich

Gruppe 9

Projektleiter:

K11808821, Thomas Deutsch, deutsch-thomas@gmx.at

Teammitglieder:

K11818868, Stefan Bolda

K11818867, Stefan Hinterhölzl

Kompetenzdarstellung

Mithilfe der Kompetenzdarstellung in diesem Abschnitt soll nicht nur das Kompetenzniveau der einzelnen Teammitglieder, sondern vor allem auch das Kompetenzniveau im Team beschrieben werden. Basierend auf der Kompetenzdarstellung ist das Potenzial für eine mögliche Weiterentwicklung im Sinne der jeweiligen Grenzen abzuleiten.

Beispiel

Alle Teammitglieder haben Erfahrung in der Konfiguration unterschiedlichster Softwaresysteme zur Unterstützung kollaborativer Arbeit. Erfahrungen in der Konfiguration und Anwendung eines solchen Systems im Kontext verteilter Software-Entwicklung konnten bisher aber noch nicht gesammelt werden.

Leitfragen

- **Welche Fähigkeiten und Kompetenzen bringen die einzelnen Mitglieder bereits jetzt ins Projekt ein?** (Separate Darstellung der Kompetenzen der Mitglieder)
 - **Bolda**
 - Einführung in die Softwareentwicklung UE,VL
 - Softwareentwicklung 2 UE,VL
 - Software Engineering UE,VL
 - Communication Engineering UE,VL
 - Datenmodellierung UE,VL
 - Algorithmen und Datenstruktur UE,VL
 - Teilzeitangestellt als Softwareentwickler
 - Kenntnisse in Bitbucket & PRIMEFACES
 - **Deutsch**
 - Einführung in die Softwareentwicklung UE,VL
 - Softwareentwicklung 2 UE,VL
 - Software Engineering UE,VL
 - Communication Engineering UE,VL
 - Datenmodellierung UE,VL
 - Algorithmen und Datenstruktur UE,VL
 - Algorithmen und Datenstruktur 2 UE,VL
 - HTL Diplomarbeit Shoppinglist (Website + App + Database)
 - **Hinterhölzl**
 - Einführung in die Softwareentwicklung UE,VL
 - Softwareentwicklung 2 UE,VL
 - Software Engineering UE,VL
 - Communication Engineering UE,VL

- Datenmodellierung UE,VL
- Algorithmen und Datenstruktur UE,VL
- Algorithmen und Datenstruktur 2 UE,VL

- **Wie ergänzen sich die im Team vorhandenen Kompetenzen?** (Darstellung der Kompetenzen über die einzelnen Mitglieder hinaus)

Da alle Mitglieder einen sehr ähnlichen Studienverlauf vorweisen sind die Kompetenzen beinahe ident. Nichts desto trotz bringen die unterschiedlichen persönlichen Backgrounds, sowie die gemeinsame Begeisterung, für das Erlernen von Webdevelopment und anderen neuen Technologien, eine gute Basis für das gegenseitige Unterstützen und Lehren.

- **Wo liegen die Grenzen der im Team vorhanden Kompetenzen?** (Wohin wollen sich die einzelnen Mitglieder bzw. das Team entwickeln?)

Da Webdevelopment im bisherigen Studienverlauf eine geringe Rolle einnahm, sind die Kompetenzen in diesem Bereich noch sehr unausgeprägt und sollen gefestigt, beziehungsweise erlernt werden. Bisher erlernte Webdevelopment-Fähigkeiten sind lediglich, Grund-HTML-Kenntnisse, erstellen und benutzen von Restful-Webservices sowie das einbinden und benutzen von Thymeleaf.

Projektbeschreibung

Basierend auf den zuvor beschriebenen Kompetenzen bzw. Grenzen ist ein konkretes Projekt abzuleiten, welches die Weiterentwicklung oder Vertiefung konkreter Kompetenzen zum Ziel hat. Ein Bezug zu den in der Kompetenzdarstellung beschriebenen Umständen (Kompetenzen, Grenzen, etc.) ist herzustellen.

Leitfragen

- **Was ist die konkrete Projektidee?** (Kurzbeschreibung)

Ziel ist es durch Anwendung der praktischen und theoretischen Aspekte eine Website zu erstellen, welche eine URL verkürzt und bei Eingabe dieser im Browser den Benutzer auf die gewünschte Seite automatisch weiterleitet. Diese Webanwendung soll auf einem selbst bereitgestellten dedizierten Server zur Verfügung gestellt werden. Die verkürzten Links werden in einer Datenbank gespeichert. Mögliche weitere optionale Funktionen sind das Anlegen von Nutzerkonten sowie die Schaltung von Werbeanzeigen. Ein Beispiel eines Unternehmens, welches dies anbietet, wäre bit.ly.

- **Was beinhaltet der theoretische Teil der Arbeit?**

- Festigen der RPC JSON Kommunikations-Kenntnisse
- Festigen der REST Kenntnisse
- Webservice durch eigenen dedizierten Server anbieten erlernen

- Verschlüsselte Datenübertragung kennenlernen (SHA-256)

- **Was beinhaltet der praktische Teil der Arbeit?**

- HTML
- CSS
- JavaScript
- Angular JS
- Erstellen einer Datenbank
- Restful web service
- API (Wahl und Einsatz)
- Swagger UI für die API Dokumentation erstellen
- Verschlüsselung

- **Welche Kompetenzen werden im Zuge des Projekts gestärkt und/oder neu entwickelt?**

Ziel ist es, dass jedes Teammitglied erlernt, wie man eine grundlegende Webapplikation erstellt, welche auf einem eigens bereitgestellten dedizierten Server läuft, und aufgerufen werden kann sowie Daten speichert und verarbeitet. Weiters ist es Ziel weitere Entwicklungsumgebungen, Plugins und Frameworks kennen zu lernen.

Ressourcen

In diesem Abschnitt sind die für die Umsetzung des Projekts notwendigen Ressourcen zu identifizieren bzw. zu beschreiben. Ressourcen bezeichnen hierbei sowohl Hardware- als auch Software-Ressourcen.

Leitfragen

- **Welche Ressourcen sind für die Umsetzung des Projekts notwendig?**
 - Visual Code als IDE
 - GitHub als Repository
 - AngularJS für Logik der Website
 - Restful-Webservices für Kommunikation
 - Bootstrap und Materialize als Stylinghilfe (CSS Framework)
- **Welche Ressourcen werden vom Projektteam zur Verfügung gestellt?**
 - Dedizierter Server

Meilensteine

Meilensteine dienen zur grundlegenden Strukturierung von Projekten und bieten die Möglichkeit, das Projekt anhand definierter Qualitätskriterien zu evaluieren bzw. zu prüfen. In diesem Abschnitt gilt es das Projekt in grundsätzlich 4 Phasen (3 Meilensteine) zu strukturieren und anhand entsprechender Deliverables zu definieren. Auf Basis der zugeordneten Deliverables soll das Projekt zum definierten Zeitpunkt (siehe Meilensteine) bewertet werden können.

Leitfragen

- **Welche Aufgaben und Ergebnisse sind den Meilensteinen zuzuordnen?**
 - Meilenstein 1
 - UI Prototype der Webapplikation
 - Logischer Entwurf der Datenbank
 - Einrichten des Git-Repositories
 - Vertraut machen mit den aufgelisteten Tools und Frameworks
 - Meilenstein 2
 - Ausführbares Webprojekt mit grundlegender Funktion des URL Verkürzens
 - Lauffähige API
 - Meilenstein 3
 - Externer Zugriff auf die Webapplikation
 - Fertiges Design der Webapplikation
 - Login und Registrierung von Nutzern und zuweisen der jeweiligen Datenbankeinträge
 - Dokumentation der API

- **Welche Deliverables (Protokolle, Berichte, Dokumentationen, Entscheidungsprozesse, Modelle, Softwaresysteme, Handbücher, etc.) werden zu den Meilensteinen abgegeben?**
 - Dokumentation der API (Swagger UI)
 - UML Darstellung der Datenbank
 - UI Prototype
 - Geklonetes Git-Repository
 - Zeitaufzeichnung
 - Theoretische Ausarbeitung

Meilenstein #1 17. April 2020 (23:59)

Titel:	Theoretische Ausarbeitung
Deliverables:	Theoretische Ausarbeitung
	Zeitaufzeichnung
	UI Prototype

Meilenstein #2 15. Mai 2020 (23:59)

Titel:	
Deliverables:	Zeitaufzeichnung
	Geklonetes Git-Repository

Meilenstein #3 12. Juni 2020 (23:59)

Titel:	
Deliverables:	Zeitaufzeichnung
	Geklonetes Git-Repository
	Dokumentation der API (Swagger UI)
	URL zum Webservice

Ergebnissicherung

Die Ergebnissicherung gewährleistet eine entsprechende Dokumentation des Projekts im Sinne eines Projektmanagements (Ablaufdokumentation, Zeitaufzeichnungen, etc.) bzw. einer inhaltlichen Projektbeschreibung (Deliverables, erreichte Ergebnisse, Funktionalitäten, etc.). Die einzelnen Dokumente der Ergebnissicherung stellen die Grundlage zur Beurteilung des Projekts dar. Die Möglichkeiten bzw. Qualitäten eines erfolgreichen Projektmanagements erhöhen sich je genauer und detaillierter die Ergebnisse in ihrer Struktur bzw. ihren Anforderungen dargestellt werden.

Leitfragen

- **Welche Deliverables (Protokolle, Berichte, Dokumentationen, Entscheidungsprozesse, Modelle, Softwaresysteme, etc.) werden am Ende des Projekts abgegeben?**
 - Dokumentation der Datenbank
 - Dokumentation der API
 - Geklonetes Git-Repository der fertigen Web-Applikation
 - Zeitaufzeichnung
 - URL um auf den Webservice zuzugreifen
 - UI Prototyp
- **Welchen Sinn und Zweck erfüllen die einzelnen Deliverables?**
 - Verstehen der Datenbank um Erweiterbarkeit und Wartung zu gewährleisten
 - Verstehen der API um Erweiterbarkeit der Website durch bereits existierende Abfrage zu ermöglichen
 - Nachvollziehbarkeit des Produzierten Codes, sowie bieten der Möglichkeit die Website selbst zu hosten
 - Nachweis der erbrachten Zeitaufwände
 - URL um auf den Webservice zuzugreifen, testen und nutzen zu können
 - Vergleich des Ursprünglichen Prototyps mit dem tatsächlich entwickelten Produkt