

Rahmenbedingungen Agiles Web-Programmierprojekt

Projektziel:

Entwicklung einer dynamischen Webseite mit mindestens 5-6 Seiten, die eine Benutzerregistrierung beinhaltet und mindestens 3-4 sinnvoll miteinander verbundene Datenbanktabellen umfasst. Es müssen SQL-Statements für SELECT, INSERT, UPDATE und DELETE implementiert werden. Das Projektthema wird mit einer zusätzlichen Funktionalität Ihrer Wahl realisiert, wie beispielsweise einer Kontaktverwaltung, Vereinsmanagement (z.B. Hobby- oder Sportvereine), einem Online-Shop oder einer Infodatenbank mit Such- und Filterfunktionen. Das Projekt wird agil umgesetzt, wobei regelmäßige Sprints zur Iteration und kontinuierlichen Verbesserung genutzt werden.

Technologie-Stack:

- **Frontend:** HTML, CSS, JavaScript
- **Backend:** PHP
- **Datenbank:** MySQL
- **SQL-Statements:** SELECT, INSERT, UPDATE, DELETE
- **Zusätzliche Anforderungen:**
 - Mindestens 5-6 Seiten (Startseite, Registrierung, Login, Dashboard, Profilseite, etc.)
 - Registrierungsmöglichkeit für Benutzer
 - Datenbank mit mindestens 4-5 miteinander verbundenen Tabellen
 - Includes für Header, Footer, Navigation und Datenbankankündigung
 - Verwendung von Kommentaren

Gruppenstruktur und Rollen:

- **Gruppengröße:** 5-6 Personen
- **Agiles Team:**
 - **Product Owner:** Verantwortlich für die Definition und Priorisierung der Anforderungen.
 - **Scrum Master:** Sicherstellung der agilen Prozesse und Teamkoordination.
 - **Entwickler (Frontend und Backend)**

Projektzeitrahmen:

- **Dauer:** 2 Wochen
- **Sprints:**
 - **Sprint 1 (Woche 1):** Planung, Design, erste Features (z.B. Registrierung, Login) und Datenbankstruktur.

- **Sprint 2 (Woche 2):** Fertigstellung der Webseite, zusätzliche Features (z.B. Profilbearbeitung).
- **Sprint Review und Retrospektive:** Am Ende jedes Sprints gibt es eine Review zur Überprüfung des Inkrements und der Arbeit und eine Retrospektive zur Verbesserung des Arbeitsprozesses.

Tägliche Treffen:

- **Zeiten:** Jeden Tag von 9:00 bis 12:30 vor Ort
- Diese täglichen Treffen dienen der täglichen Abstimmung, der Bearbeitung von Aufgaben und der Durchführung von Code-Reviews sowie Klärung der offenen Fragen. Hier werden auch Anpassungen und Aufgaben für den nächsten Sprint besprochen.

Verwendung von Trello:

- **Trello Board:** Zur Verwaltung und Organisation der Aufgaben während des gesamten Projekts.
 - **Sprint-Board:** Aufgaben, die für den aktuellen Sprint geplant sind.
 - **Done, Doing, To-Do**
- Trello hilft dabei, die Aufgaben klar zu verteilen, Fortschritte zu verfolgen und die Kommunikation im Team zu fördern.

Präsentationen:

- **Zwischenpräsentation (5-10 Minuten):** Vorführung der bisherigen Ergebnisse, einschließlich der ersten Sprints und implementierten Features. Feedback einholen und ggf. Anpassungen vornehmen.
- **Endpräsentation (20-25 Minuten):** Vorführung der fertigen Webseite. Präsentation der iterativen Entwicklung, der verwendeten Technologien und der Umsetzung der Anforderungen.
- Reflexion über den agilen Prozess und Teamarbeit.

Bewertungskriterien:

- **Funktionalität:** Erfüllung der grundlegenden Anforderungen, einschließlich der Datenbankintegration und Benutzerregistrierung.
- **Codequalität:** Sauberer, kommentierter Code, der die Best Practices der agilen Entwicklung widerspiegelt.
- **Benutzerfreundlichkeit:** Ansprechendes und intuitives Frontend. Agiler Prozess: Die Fähigkeit des Teams, agil zu arbeiten, mit regelmäßigen Reviews und Anpassungen.
- **Präsentationen:** Klarheit und Professionalität in der Darstellung des Projekts.

- **Engagement:** Aktive und kontinuierliche Mitwirkung unter Berücksichtigung der Arbeitsanweisungen.
- **Zeitmanagement:** Effiziente Planung und Einhaltung der Zeitvorgaben.
- **Teamfähigkeit:** Förderung der Zusammenarbeit und Sicherstellung der Arbeitsfähigkeit der Gruppe.
- **Respekt und Toleranz:** Aufmerksames Zuhören und sachlicher Umgang mit den Beiträgen anderer.
- **Konfliktbewältigung:** Offenheit für konstruktive Lösungs- und Vermittlungsvorschläge

Abgabematerialien:

- **PHP-Quellcode:** Der gesamte Code für das Backend, einschließlich der Datenbankinteraktionen.
- **SQL-Export-Datei:** SQL-Export-Datei als zip-Archiv zum Download.
- **GitHub-Repository:** Das gesamte Projekt muss auf GitHub gehostet werden, einschließlich aller Quellcodes und Dokumentationen.
- **Kurzbeschreibung:** Erstellung einer 1 bis 2 DIN-A4-Seiten umfassenden PDF-Datei mit einer Zusammenfassung des Konzepts, der verwendeten Techniken und der Datenbankstruktur.

Ressourcen:

- **Technische Ressourcen:** Webservice oder lokale Serverumgebung (z.B. XAMPP für lokale Entwicklung).