



Zukunftsorientierte Webanwendung mit automatisierter Qualitätssicherung

Projektarbeit im Bachelor-Studiengang Praktische Informatik
PIB-PA Sommersemester 2025

Motivation und Hintergrund

In der dynamischen Welt der Softwareentwicklung zählt Qualität von Anfang an:

„Inspection does not improve the quality, nor guarantee quality. Inspection is too late. The quality, good or bad, is already in the product. Quality cannot be inspected into a product or service; it must be built into it.“ (W. Edwards Deming in Out of the Crisis)

Sie entwickeln eine moderne, interaktive Webanwendung, die dank kontinuierlicher und automatisierter Qualitätssicherung stets höchsten Ansprüchen genügt. Dabei minimieren Sie die Entstehung von „Cruft“¹ und fördern eine nachhaltige, agile Entwicklung. Wenn Sie sich für innovative Webtechnologien und agile Arbeitsmethoden begeistern, ist dieses Projekt Ihre Chance, Ihr Können in der Praxis unter Beweis zu stellen.

Projektbeschreibung

Sie realisieren im Team eine **dynamische Webanwendung** mit **NestJS** – einem modernen serverseitigen Framework mit TypeScript-Unterstützung. Neben der Entwicklung von Kernfunktionalitäten (wie Benutzerverwaltung, Datenpersistenz und API-Design) integrieren Sie von Anfang an automatisierte Qualitätssicherungsprozesse:

Statische Codeanalyse: Einsatz von Tools wie ESLint.

Automatisierte Tests: Unit-Tests (z. B. mit Jest), Integrationstests und End-to-End-Tests (z. B. mit Playwright).

CI/CD-Pipeline: Automatisiertes Bauen, Testen und Veröffentlichen (z. B. via GitHub Actions).

¹Cruft ist Fachjargon für alles, was übrig, überflüssig und hinderlich ist. Der Begriff wird insbesondere für fehlerhafte, überflüssige, nutzlose, überflüssige oder dysfunktionale Elemente in Computersoftware verwendet. Siehe auch: martinfowler.com/articles/is-quality-worth-cost.html

Darüber hinaus führen Sie das Projekt agil, also iterativ und inkrementell, durch. Wählen Sie eigenverantwortlich ein geeignetes Vorgehensmodell (z. B. Scrum oder Kanban) und strukturieren Sie Ihre Arbeit in regelmäßigen Zyklen.

Aufgabenstellung

So stellen Sie Ihre Kompetenz unter Beweis:

Konzept und Architektur

- Entwickeln Sie ein Gesamtkonzept, das funktionale und nicht-funktionale Anforderungen abdeckt.
- Planen Sie eine modulare Architektur, die Frontend, Backend und Qualitätssicherung klar trennt.
- Setzen Sie Ihr Projekt mit einem agilen Entwicklungsprozess um (z. B. Scrum oder Kanban).

Umsetzung

- Entwickeln Sie eine dynamische Webanwendung mit NestJS.
- Integrieren Sie Funktionalitäten wie Benutzerverwaltung, Datenpersistenz, API-Design und Session-Management.
- Realisieren Sie optional ein ansprechendes Frontend.

Qualitätssicherung

- Richten Sie eine statische Codeanalyse ein.
- Implementieren Sie umfassende automatisierte Tests.
- Bauen Sie eine CI/CD-Pipeline auf, die den gesamten Build- und Deployment-Prozess steuert.

Dokumentation und Reflexion

- Dokumentieren Sie Ihre Architekturentscheidungen, Implementierungsschritte und Qualitätssicherungsmaßnahmen.
- Verfassen Sie einen abschließenden Erfahrungsbericht, in dem Sie unter anderem auf die agile Arbeitsweise und deren Vorteile eingehen.

Voraussetzungen und Erwartungen

Sie sollten bereits Erfahrung in der Webentwicklung mit HTML, CSS, JavaScript und Git haben, um in diesem Projekt erfolgreich zu sein. Grundkenntnisse in TypeScript, Node.js und serverseitigen Frameworks sind wünschenswert. Sie sollten sich für agile Arbeitsmethoden begeistern und bereit sein, diese eigenverantwortlich auszuwählen und umzusetzen.

Prof. Dr.-Ing. Martin Burger

Software Engineering und Software Quality Assurance

martin.burger@htwsaar.de

sesqa.de

**ingenieur
wissenschaften
htw saar**