

INF202 Meilenstein #4

Verantwortliche/r:

Teoman Turan, e180503040@stud.tau.edu.tr

Bariş Emre Yaşar, e180503038@stud.tau.edu.tr

1)Systemüberblick

Tests sind ein integraler Bestandteil des Reservierungs-Webprojekts und sollen sicherstellen, dass alle Features und Funktionalitäten wie erwartet funktionieren und ein nahtloses Benutzererlebnis bieten. Unsere Teststrategie umfasst mehrere Testmethoden und Testebenen, um jede Komponente der Anwendung sowie das System als Ganzes zu validieren. Dafür benutzen wir Restful Testing..Für End-to-End-Tests wird „Cypress“ verwendet. Mit Cypress ist es möglich, sowohl UI-Tests als auch Komponententests durchzuführen.

2)Systemtestesfälle

Testfall	Test der Einloggen
Anfangsbedingungen	Benutzer hat sich noch nicht eingeloggt
Test Schritten	<ol style="list-style-type: none">1. Benutzer öffnet die Applikation.2. Benutzer klickt auf die Login-Schaltfläche.3. Person gibt einen Benutzernamen und ein Passwort ein, die gültig sind.
Erwartetes Ergebnis	<ul style="list-style-type: none">• Benutzer eingeloggt zum System
Tatsächliche Ergebnis	<ul style="list-style-type: none">• Benutzer eingeloggt zum System

Testfall	Test der Registrierung mit E-Mail und Benutzername
Anfangsbedingungen	Benutzer hat sich noch nicht registriert

Test Schritten	<ol style="list-style-type: none"> 1. Nutzer öffnet die Applikation. 2. Benutzer klickt auf "Create Account" 3. Der Benutzer füllt das Registrierungsformular mit dem gewünschten regulären Ausdruck aus (diese werden dem Benutzer als Info während des Ausfüllens des Formulars angezeigt) 4. Benutzer klickt nach dem Ausfüllen des Formulars auf die Schaltfläche "Sign Up"
Erwartetes Ergebnis	<ul style="list-style-type: none"> • Der Benutzer kann sich mit den Informationen anmelden, die er bei der Registrierung eingegeben hat.
	<ul style="list-style-type: none"> • Der Benutzer kann sich mit den Informationen anmelden, die er bei der Registrierung eingegeben hat.

3) Rückverfolgbarkeit

****Traceability in the Reservation Web Project****

In unserem Projekt achten wir auf ein hohes Maß an Rückverfolgbarkeit und stellen sicher, dass jede Funktion und Komponente direkt mit einer bestimmten Anforderung oder User Story verknüpft ist. Hier ist eine mögliche Aufschlüsselung der Rückverfolgbarkeit in verschiedenen Teilen des Projekts:

1. ****Requirements Traceability****: Wir verknüpfen jedes Feature und jede Funktionalität mit einer spezifischen Anforderung oder User Story. Dadurch stellen wir sicher, dass wir alle Anforderungen erfüllen und erleichtern die Auswirkungsanalyse, wenn sich eine Anforderung ändert. Diese Rückverfolgbarkeit ermöglicht es uns, die Teile des Systems zu identifizieren, die aktualisiert werden müssen.

2. ****Code Traceability****: Code-Implementierungen sind mit spezifischen Anforderungen verknüpft, die sie erfüllen, und mit Design Dokumenten, auf denen sie basieren. Diese Rückverfolgbarkeit stellt sicher, dass der Code einem klaren Zweck dient, und es wird einfacher, das System zu lokalisieren und zu ändern, wenn sich Anforderungen oder Designs ändern.

3. ****Test Traceability****: Testfälle stehen in direktem Zusammenhang mit den Anforderungen, die sie validieren sollen, und stellen so sicher, dass alle Anforderungen angemessen getestet werden. Darüber hinaus hilft die Rückverfolgbarkeit fehlgeschlagener Tests auf ihre Anforderungen dabei, potenziell nicht erfüllte Anforderungen zu identifizieren.

4. **Issue Traceability**: Es bestehen Verbindungen zwischen allen Problemen, Fehlern oder Verbesserungen und den Systemteilen, die sie betreffen, einschließlich spezifischer Anforderungen, Code und Testfälle. Diese Rückverfolgbarkeit hilft, Probleme schnell zu beheben und ihre möglichen Auswirkungen auf das System zu verstehen.

5. **Deployment Traceability**: Wir führen Aufzeichnungen über die Code-, Konfigurations- und Dateiversionen, die in jeder Systemversion enthalten sind. Diese Rückverfolgbarkeit stellt sicher, dass jede Systemversion zu Fehlerbehebungs- oder Prüfzwecken reproduziert werden kann.

Betrachten wir eine hypothetische Reihe von Anwendungsfällen, Webdienst-Endpunkten und Testfällen, um deren Rückverfolgbarkeit zu veranschaulichen:

Use Case	Web Service Endpoint	Test Case
User creates a new reservation	POST /reservations	TestReservationCreation
User modifies an existing reservation	PUT /reservations/{id}	TestReservationModification
User cancels a reservation	DELETE /reservations/{id}	TestReservationCancellation
User views a specific reservation	GET /reservations/{id}	TestReservationView
User browses all their reservations	GET /reservations	TestReservationBrowsing

Diese Tabelle veranschaulicht die Rückverfolgbarkeit zwischen jedem Anwendungsfall, seinem entsprechenden Webdienst-Endpunkt und dem Testfall, der seine Funktionalität überprüft. Wenn anwendungsfall bezogene Änderungen oder Probleme auftreten, können wir diese auf den entsprechenden Endpunkt und Testfall zurückführen und so die Problemisolierung und Lösungsimplementierung vereinfachen.