

Szenario 1 – Performance (Budgetverwaltung)

Quelle: Nutzer

Stimulus: Nutzer speichert eine neue Ausgabe

Artefakt: Budgetmodul / DB-Zugriff

Umgebung: Normalbetrieb

Antwort: Die Transaktion wird korrekt gespeichert, Schulden berechnet und Übersicht aktualisiert

Antwortmaß: Latenz < 1 Sekunde

Szenario 2 – Verfügbarkeit (WG-Zugriff) (NF04)

Quelle: WG-Mitglied

Stimulus: Öffnet WebApp

Artefakt: Gesamtsystem (Frontend/Backend/DB)

Umgebung: Normalbetrieb

Antwort: App ist erreichbar

Antwortmaß: 99 % Verfügbarkeit

Szenario 3 – Robustheit (Fehlerhafte Eingaben bei der Registrierung)

Quelle: Nutzer

Stimulus: Gibt ungültige E-Mail oder zu kurzes Passwort ein

Artefakt: Registrierungsformular / Validierungslogik

Umgebung: Normalbetrieb

Antwort: Nutzer bekommt sofort eine klare Fehlermeldung, ohne Reload

Antwortmaß:

- Validierungsfeedback in < 0,3 Sekunden
- Keine ungültigen Accounts werden erstellt
- Formular bleibt nutzbar, Eingaben gehen nicht verloren

Szenario 4 – Änderbarkeit / Erweiterbarkeit

Quelle: Entwickler

Stimulus: Hinzufügen eines neuen Features (z. B. Challenges)

Artefakt: Codebasis (Monolith)

Umgebung: Entwicklungsphase

Antwort: Neue Module lassen sich einfügen ohne bestehenden Code zu verändern

Antwortmaß: Implementationszeit \leq 2 Personentage

Szenario 5 – Usability (Einkaufsliste)

Quelle: Nutzer

Stimulus: Fügt neuen Artikel hinzu

Artefakt: UI & Einkaufsliste

Umgebung: Normalbetrieb

Antwort: Artikel wird unmittelbar sichtbar

Antwortmaß: Sichtbarkeit $< 0,5$ Sekunden, kein Reload nötig

Qualitätsbaum (Utility Tree) – FlatMate

Qualitätsattribut	Verfeinerung	Szenario	Business Value	Techn. Risiko
Performance	Latenz	S1: Ausgabe speichern $< 1\text{s}$	Hoch	Mittel
Verfügbarkeit	Betriebszeit	S2: 99% Uptime	Hoch	Mittel
Wartbarkeit	Modularität	S4: Feature in < 2 Tagen	Mittel	Niedrig
Robustheit	Validierungsfeedback	S3: Validierungsfeedback in $< 0,3\text{s}$	Mittel	Mittel
Usability	UI-Reaktionszeit	S5: Nutzerfeedback $< 0,5\text{s}$	Hoch	Niedrig