

Spickzettel: Testtypen & Testpyramide (Softwaretests)

Ziel

Verständnis für die verschiedenen Arten von Tests und deren Verhältnis im Projekt (Teststrategie).

Testarten im Überblick

Unit Tests

- Testen **einzelne Funktionen oder Klassen** isoliert
- Schnell, klein, unabhängig
- Beispiele: pytest, unittest

Integrationstests

- Testen Zusammenspiel mehrerer Komponenten
- Z. B. DB + API, Services + Datenzugriffe
- Benötigen oft Testdaten oder Mocks/Stubs

End-to-End-Tests (E2E)

- Simulieren **realistische Nutzerinteraktionen**
- Testen komplette Anwendung (Frontend ↔ Backend ↔ DB)
- Tools: Selenium, Playwright, Cypress

Manuelle Tests

- Exploratives Testen, UI, Akzeptanztests, Usability-Checks
 - Ergänzen automatisierte Tests
-

Testpyramide

End-to-End (wenige)
Integrationstests
Unit Tests (viele)

- Viele kleine Unit Tests
- Wenige, gezielte Integrationstests
- Sehr wenige, aber wichtige E2E-Tests

Ziel: Effizient testen mit guter Abdeckung bei vertretbarem Aufwand

Best Practices

- Schreibe zuerst Unit Tests für Logik
 - Integrationstests nur dort, wo Interaktion entscheidend ist
 - E2E-Tests für Kern-User-Flows (Login, Checkout, ...)
 - CI/CD: Unit + Integration bei jedem Push, E2E z. B. täglich
-

Die Testpyramide hilft, stabile und wartbare Teststrategien zu entwickeln – und Zeit beim Debugging zu sparen.