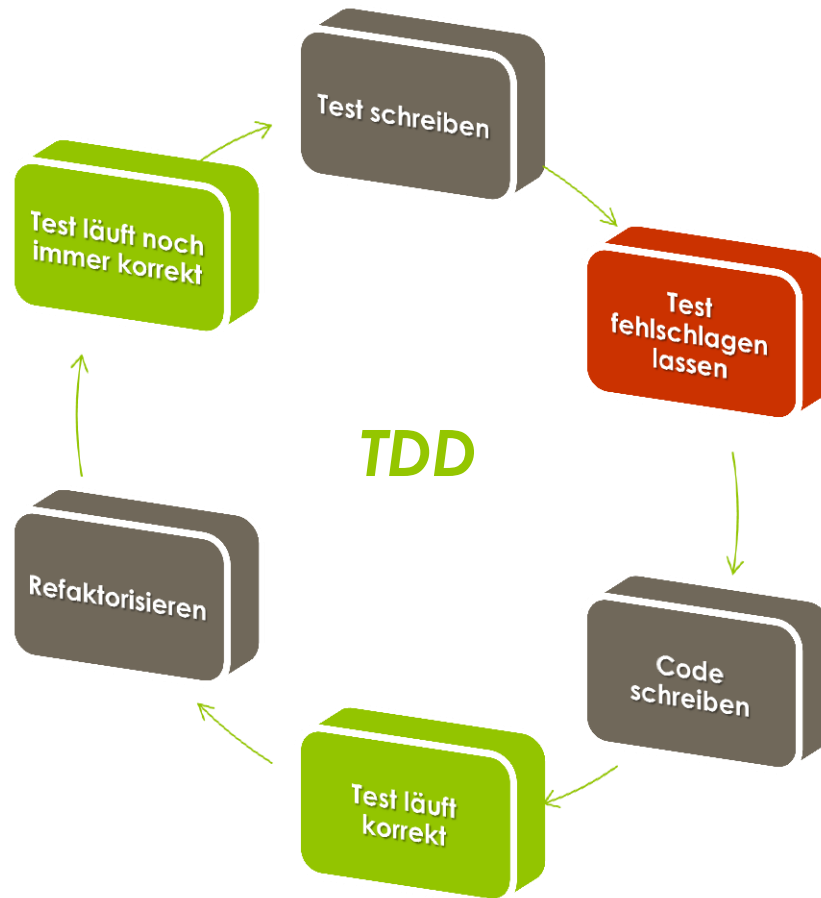
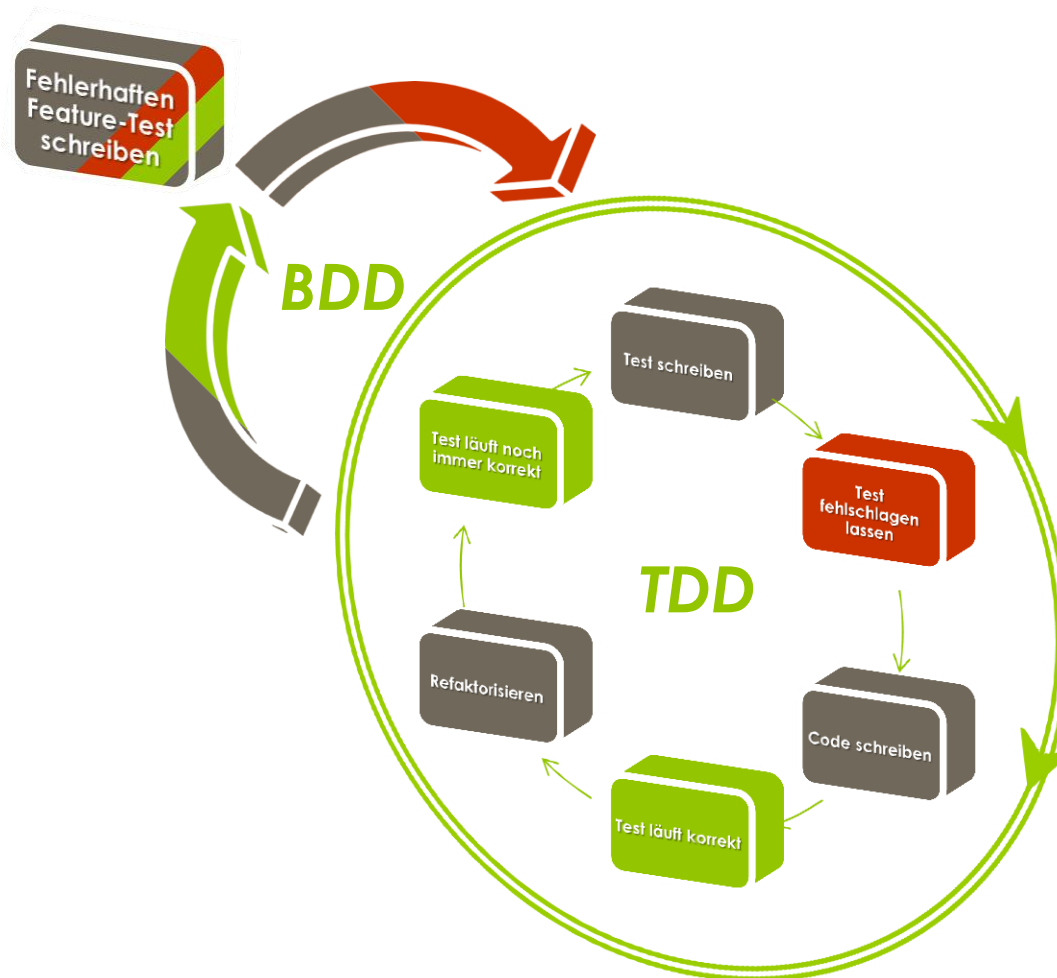


Test Driven Development | TDD



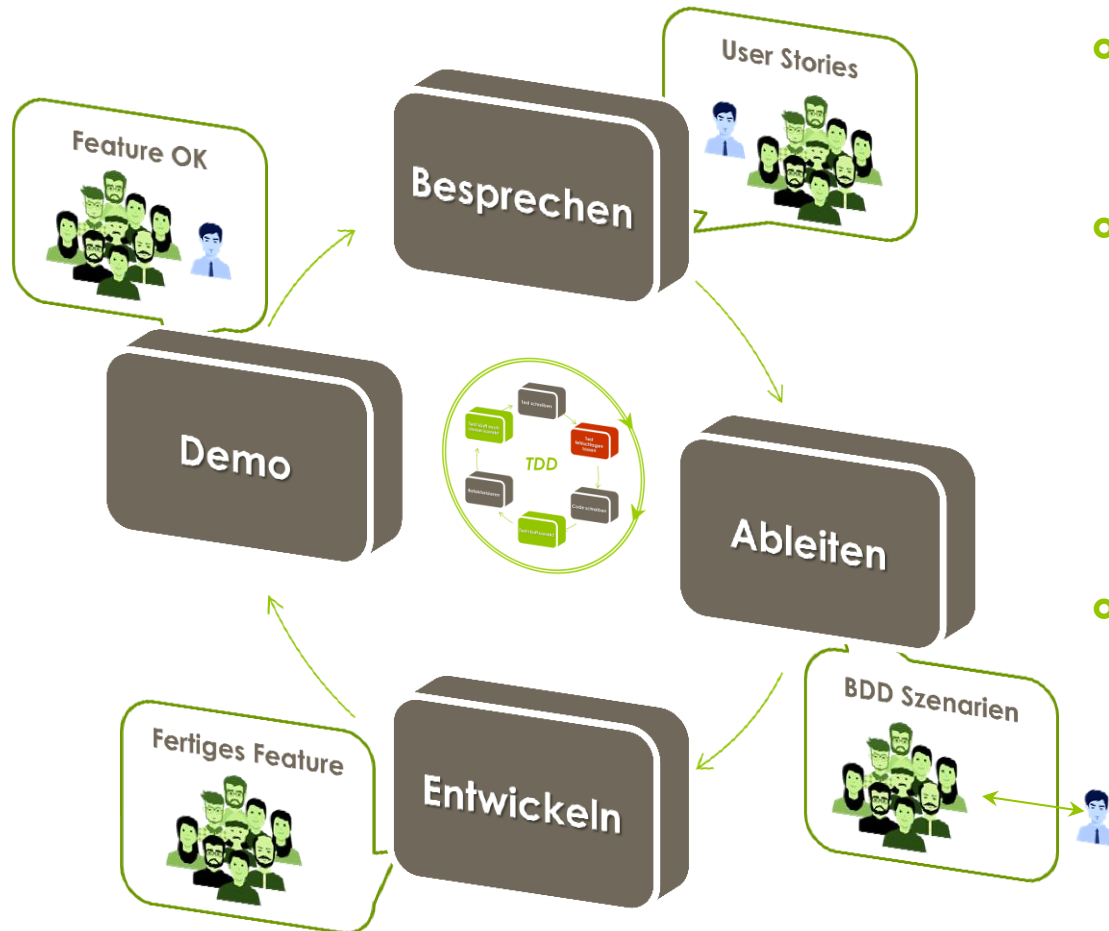
- Den meisten wird der Zyklus zu TDD bekannt sein.
- Doch aus Sicht der QA sollte verstärkt darauf geachtet werden, dass das „Refactoring“ tatsächlich stattfindet, um eben später nicht vor dem Problem schlecht wartbarer Tests zu stehen.
- In erster Linie obliegt TDD noch immer den Entwicklern, doch regelmäßige Reports (automatisiert) durch die QA helfen allen den TDD-Ansatz kontinuierlich einzusetzen.

Behavior Driven Development | BDD



- Das BDD TDD als Fundament nutzt, sollte bekannt sein.
- Hierbei sollte die QA auf die Idee „Collaboration“ besonders achten, da die End-User-Sicht (Kunde / Product Owner) zum tragen kommt und für die Umsetzung eines korrektes Features ungemein hilft.
- Wenn bspw. PO und DEV-Team (QA Rolle) das Verhalten der Software gemeinsam konzipieren, steht dem Umsetzungserfolg meist wenig entgegen.

Acceptance Test Driven Development | ATDD



- ATDD konzentriert sich auf die Erfassung von Anforderungen in Akzeptanztests.
- Die QA sollte verstärkt darauf achten, dass die verschiedenen Perspektiven: Kunde / Product Owner, DEV-Team und QA (QA Rolle im DEV-Team) präzise verstehen, was zu tun und umzusetzen ist.
- Ziel sollte sein das die Akzeptanzkriterien bzw. deren Tests (z. B. via BDD) bestätigen, dass die Funktionalität getan („done“) ist.