

**Continuous Integration (CI)** – handlar om att kontinuerligt integrera kod i ett centralt repo, och detta sker ofta. CI är för att säkerställa en snabb och kvalitativ kodleverans som då gör det enklare att upptäcka och korrigera fel.

Styrkor:

- Snabbare leveranser
- Högre kvalitet där ny kod automatiskt testas så upptäckten av buggar hittas innan den når prod
- Enklare att hantera förändringar i koden

Svagheter:

- Inledande installationstid och utbildning krävs för att bekanta sig med CI-servern
- Utveckling av lämpliga tester är avgörande
- Väntetider kan uppstå när flera utvecklare vill integrera sin kod samtidigt

## **Testautomatisering och koppling till CI/CD:**

Testautomatisering innebär att tester skrivs för att köras automatiskt vid varje kodändring.

I en CI/CD-pipeline kopplas testautomatisering in som ett steg direkt efter varje push eller pull request. Det ger snabb feedback om något gått sönder, vilket minskar risken att fel går ut i produktion.

## **Github med Github actions**

- Gå till projektets GitHub-repo och klicka på fliken Actions.
- Välj ett färdigt arbetsflöde eller skapa ett eget.
- Skapa en YAML-fil i `.github/workflows/`, t.ex. `maven.yml`.

Definiera:

- Trigger: t.ex. `on: push` eller `on: pull_request`.
- Steg: installera Java, köra `mvn test`, etc.

Tryck på Commit changes – din workflow funkar nu

Varje gång kod pushas körs testerna automatiskt av GitHub Actions.