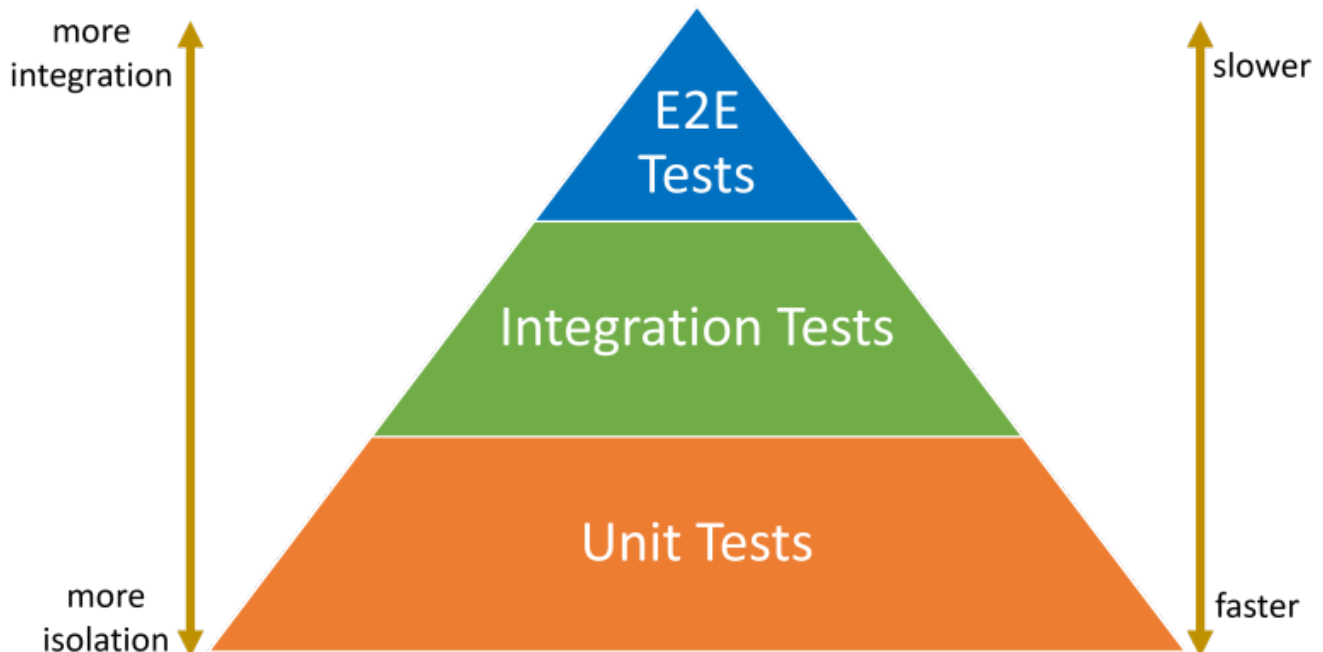


# LU05a - Teststufen

Es gibt unzählige Grafiken von Testpyramiden mit den unterschiedlichsten Begriffen, Achsen, Stufen etc. Das Essenzielle haben sie aber alle gemeinsam:

- Schnelle/günstige Tests sind Unten und sollten in der grössten Anzahl geschrieben werden.
- Langsame/teure Tests sind Oben und sollten sparsam eingesetzt werden.



© <https://juliocasal.com/blog/Should-You-Write-Unit-Tests>

Mögliche weitere Stufen wären zum Beispiel: „Smoke Tests“, „Manuelle Tests“, „User Acceptance Tests“, „System Tests“.

Wir konzentrieren uns jedoch vorerst auf die folgenden 3.

## Unit-Testing

Ein Unit-Test testet idR eine einzelne Funktion (die natürlich andere Funktionen aufrufen kann). Das wichtigste Abgrenzungsmerkmal ist aber, dass ein Unit-Test keine Datenbank, Datei, Storage oder dergleichen verwendet. Dadurch ist er schnell und unabhängig und kann *en masse* eingesetzt werden.

Unit-Tests eignen sich auch für Testing First einsetzen, was bei komplexeren Tests teilweise nicht mehr möglich ist.

## Integration-Testing

Die Integration-Tests überprüfen in der Regel das Zusammenspiel mehrerer Einheiten. Dies kann

kann zum Beispiel ein Service sein. Umsysteme, Datenbanken, Dateien etc. können bei Bedarf auch gemockt werden.

## E2e-Testing

End-to-end-Tests rufen direkt die Benutzeroberfläche auf und testen zum Beispiel einen Use-Case oder gar einen Prozess. Die Eingaben sind dabei oft einem Benutzer nachempfunden. Anstatt Mocks werden oftmals Testinstanzen verwendet. Dadurch kann man überprüfen, ob der gesamte Stack (Datenbank, Backend und Frontend) als Ganzes funktioniert.

From:  
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:  
<https://wiki.bzz.ch/de/modul/ffit/3-jahr/cicd/learningunits/lu05/a>

Last update: **2026/03/10 10:17**

