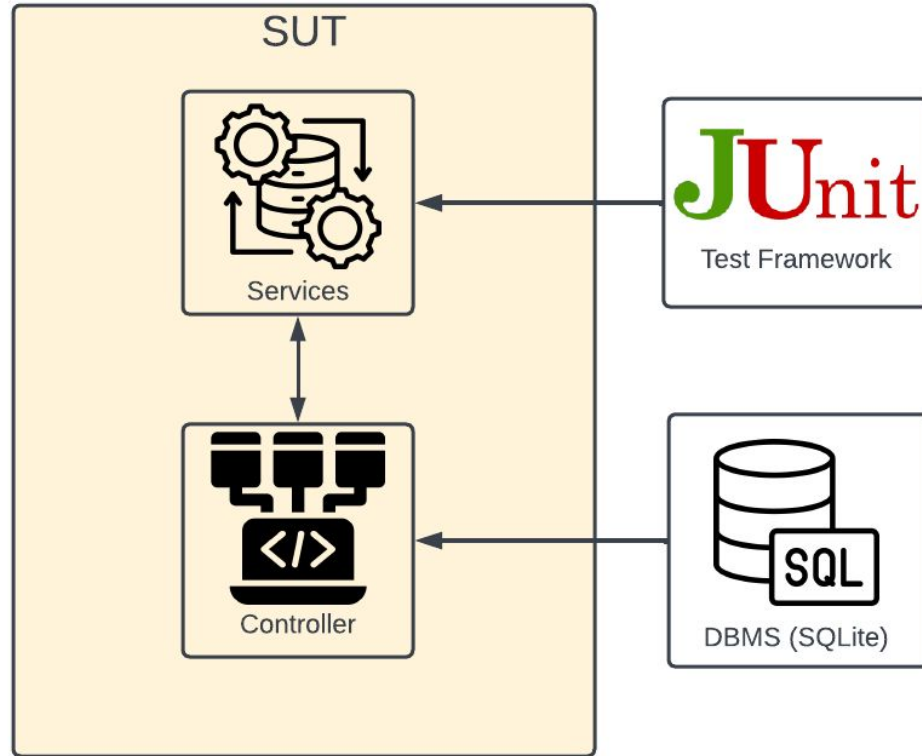


STOCKWELL

MS#4
Systemtest

BURAK ARSLAN 200503051
ÇAĞLAR KABACA 200503053

Systemüberblick



Systemtestfall

System Test ID Name	ID001 - Test um ein Produkt im Lager hinzuzufügen und zu überprüfen ob es im Lager sichtbar ist.
Preconditions	<ul style="list-style-type: none">• LagerService<ul style="list-style-type: none">◦ Request -> AddProductLagerDB(LagerProduct product) und GetProductImLager(String uuid)◦ Response -> das LagerProduct, das schon hergestellt wurde.<ul style="list-style-type: none">▪ Data "LagerProduct"<ul style="list-style-type: none">• String uuid• Product product• Date date• int amount• int cost• LagerProduct product<ul style="list-style-type: none">◦ uuid: unique◦ product: <Test Produkt>◦ Date: "20 May 2020: 12:31"◦ amount: 30◦ cost: 50
Test Steps	<ol style="list-style-type: none">1. Stellt ein Produkt im Lager mithilfe des AddProductLagerDB Services her.2. Nimmt das LagerProduct vom Lager mithilfe des GetProductImLager Services ein.3. Prüft diese zwei Produkte über, die gesendet und eingenommen werden, ob sie genauso sind.
Post Condition	Ein LagerProduct im LagerDB - LagerProduct product uuid: unique product: <Test Produkt> Date: "20 May 2020: 12:31" amount: 30 cost: 50

Systemtestfall

Test Data	none
Expected Result	<ul style="list-style-type: none">• LagerProduct product<ul style="list-style-type: none">◦ uulD: unique◦ product: <Test Produkt>◦ Date: "20 May 2020: 12:31"◦ amount: 30◦ cost: 50
Actual Result	<ul style="list-style-type: none">• LagerProduct product<ul style="list-style-type: none">◦ uulD: unique◦ product: <Test Produkt>◦ Date: "20 May 2020: 12:31"◦ amount: 30◦ cost: 50
Verdict (Pass/Fail)	Pass
Verified UC & Req. IDs	UI-4 / Lagerverwaltung

Integrationstestfall

System Test ID Name	DB_ID001 Test um eine Transaktion zu erstellen und diese Transaktion im Database hinzufügen.
Preconditions	<ul style="list-style-type: none">• Database Table "transaktionen" enthält eine Zeile für<ul style="list-style-type: none">◦ uulD (String): unique◦ productID (String): unique◦ kundeID (String): unique◦ mitarbeiterID (String): unique◦ type (String): "Test Type"◦ amount (Integer): 10◦ cost (Integer): 200◦ date (String): "20 May 2020: 12:31"• Stellen Sie sicher, dass das SpringFramework richtig läuft.
Test Steps	<ol style="list-style-type: none">1. Stellen Sie sicher, dass das SpringFramework richtig läuft.2. Junit Test Case ruft die createNewTransaktion(Transaktion transaktion) Funktion mit dem Wert transaktion, der die Werte uulD, productID, kundeID, mitarbeiterID, type, amount, cost und date auch darin enthält.3. Das Response gibt ein Boolean Wert zurück, das "true" sei, wenn die Funktion ohne Fehler läuft, sonst "falsch".
Post Condition	none
Test Data	Transaktion (<unique uulD>, <ID der TestProduct>, <ID der TestKunde>, <ID der TestMitarbeiter>, "Test Type", 10, 200, "20 May 2020: 12:31")
Expected Result	true
Actual Result	true
Verdict (Pass/Fail)	Pass
Verified UC & Req. IDs	UI-5 / Einkäufe und Verkäufe

Testbeispiele

✓ Tests passed: 5 of 5 tests – 679 ms	
✓ DatabaseTester (org.atom.stockwell.db)	679 ms
✓ createMitarbeiter	481 ms
✓ createTransaktion	67 ms
✓ addProductLager	110 ms
✓ createPerson	9 ms
✓ createProduct	12 ms

Rückverfolgbarkeit

