

## 1. ToDoServiceTest

**Lokacija:** `src/test/java/com/example/ToDo_App/service/ToDoServiceTest.java`

**Opis:**

Ta razred testira delovanje servisnega sloja aplikacije (`ToDoService`), ki uporablja repozitorij `IToDoRepo`.

Testi preverjajo shranjevanje, posodabljanje in pridobivanje opravil (`ToDo`), vključno z obravnavo napak in izjem.

Št.	Ime testa	Opis	Vrsta testa	Pričakovani rezultat
1	<b>saveOrUpdate_Success</b>	Preveri, ali se <code>ToDo</code> uspešno shrani, ko <code>save()</code> in <code>findById()</code> vrneta veljaven objekt.	Pozitiven	Metoda vrne <code>true</code> ; klicani sta <code>save()</code> in <code>findById()</code> .
2	<b>saveOrUpdate_Fails_WhenIdMissingAfterSave</b>	Preveri primer, ko se <code>ToDo</code> shrani, vendar repozitorij ne dodeli ID.	Negativen	Servis vrne <code>false</code> ; <code>save()</code> se kliče z <code>null</code> ID-jem.
3	<b>saveOrUpdate_Fails_WhenRepoThrows</b>	Simulira situacijo, ko metoda <code>save()</code> vrže izjemo v repozitoriju.	Negativen	Pričakuje se <code>RuntimeException</code> .
4	<b>getById_Success</b>	Testira pridobivanje <code>ToDo</code> elementa po obstoječem ID-ju.	Pozitiven	Vrne veljaven objekt z ustreznim ID in naslovom.
5	<b>getById_NotFound</b>	Preveri primer, ko <code>ToDo</code> z določenim ID-jem ne obstaja.	Negativen	Vrne <code>null</code> .
6	<b>getById_InvalidId</b>	Testira obnašanje, ko je ID 0 ali negativen.	Negativen	Vrne <code>null</code> ; metoda še vedno kliče repozitorij.
7	<b>getByStatus_Success</b>	Pridobivanje vseh opravil s statusom "Done".	Pozitiven	Seznam vsebuje samo elemente s statusom "Done".

8	<b>getByStatus_Empty</b>	Ko ni opravil z danim statusom.	Negativen	Vrne prazen seznam.
9	<b>getByStatus_Null_Throws</b>	Ko je status null.	Negativen	Pričakuje se <code>NullPointerException</code> .
10	<b>getByStatus_BlankOrSpaces_ReturnsEmpty</b>	Ko je status prazen niz ali vsebuje le presledke.	Negativen	Vrne prazen seznam.
11	<b>getByStatus_RepoThrows_Propagates</b>	Ko repozitorij vrže izjemo (npr. napaka v bazi).	Negativen	Pričakuje se <code>RuntimeException</code> .

## 2. TodoControllerTest

**Lokacija:** `src/test/java/com/example/ToDo_App/controller/ToDoControllerTest.java`

**Opis:**

Ta razred testira sloj kontrolerja (`ToDoController`) s pomočjo knjižnice Spring `MockMvc`.

Testi preverjajo HTTP poti, vedenje pri preusmeritvah, izpis pogledov (view-ov) in obravnavo napak iz servisa.

Št.	Ime testa	Opis	Vrsta testa	Pričakovani rezultat
1	<b>testViewAllToDoItems_Success</b>	Testira GET <code>/viewToDoList</code> – preveri, ali pot deluje in vrne 200 OK.	Pozitiven	Status 200 (OK).
2	<b>testViewAllToDoItems_NotFound</b>	Klic neobstoječe poti ( <code>/nonExistentPath</code> ).	Negativen	Status 404 (Not Found).
3	<b>testAddToDoItem_View</b>	GET <code>/addToDoItem</code> – prikaz obrazca za dodajanje novega opravila.	Pozitiven	Status 200 in pogled "AddToDoItem".
4	<b>testUpdateToDoStatus_Success</b>	GET <code>/updateToDoStatus/{id}</code> , ko servis uspešno posodobi status.	Pozitiven	Preusmeritev na <code>/viewToDoList</code> ; flash sporočilo "email sent".
5	<b>testUpdateToDoStatus_Failure</b>	GET <code>/updateToDoStatus/{id}</code> – ko servis ne uspe posodobiti statusa.	Negativen	Preusmeritev na <code>/viewToDoList</code> ; flash sporočilo "Error marking task as complete."
6	<b>testViewAllToDoItems_ServiceThrows</b>	Ko <code>ToDoService</code> vrže <code>RuntimeException</code> med GET <code>/viewToDoList</code> .	Negativen	Pričakuje se <code>ServletException</code> (propagirano naprej).