

Software Core Fundamental Project

Database and api created on spring boot

concept

- What did I want to do
- Why did I want to do it

CLUB

melee weapon (simple)

Category: Items

Damage: 1d4

Damage Type: Bludgeoning

Item Rarity: Standard

Properties: Light

Weight: 2

sprint

- The needed picture and the ideal picture
- Split into 2 parts stretch goals and needed

The screenshot shows a Jira sprint board for 'DWD Sprint 1' (12 Sep – 16 Sep, 25 issues). The board is divided into two main sections: 'Issues without epic' on the left and the main sprint board on the right. The left sidebar shows a list of issues categorized by 'required functionality' and 'stretch tasks'. The main board lists 25 issues, each with a status (e.g., 'DONE', 'TO DO'), a priority (e.g., 'REQUIRED FUNCTIONALITY', 'STRETCH TASKS'), and an assignee. The issues are organized into two columns: 'REQUIRED FUNCTIONALITY' and 'STRETCH TASKS'. The 'REQUIRED FUNCTIONALITY' column contains 10 issues, and the 'STRETCH TASKS' column contains 15 issues. The 'REQUIRED FUNCTIONALITY' column is further divided into 'DONE' and 'TO DO' sections. The 'STRETCH TASKS' column is further divided into 'TO DO' and 'DONE' sections. The 'DONE' section in the 'REQUIRED FUNCTIONALITY' column contains 10 issues, and the 'TO DO' section contains 0 issues. The 'TO DO' section in the 'STRETCH TASKS' column contains 15 issues, and the 'DONE' section contains 0 issues.

Issue ID	Description	Category	Status	Assignee
DWD-1	as a developer I want to store DnD items and weapons in a database for reference use	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-2	As a user I want to be able to view DnD items on the frontend	REQUIRED FUNCTIONALITY	TO DO	[User]
DWD-3	As a user I want to be able to add new items into the database	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-4	As a developer I want each type of entry categorized by either item or weapon	REQUIRED FUNCTIONALITY	TO DO	[User]
DWD-5	As a user I want to view a picture of the item I'm looking at on the database	STRETCH TASKS	TO DO	[User]
DWD-6	as a user I don't want to use more than the frontend to access the API	STRETCH TASKS	TO DO	[User]
DWD-7	as a user I want to delete unwanted items from the API	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-8	As a user I want a simple interface to control the API	REQUIRED FUNCTIONALITY	TO DO	[User]
DWD-9	As a user I want to update previous entries in the database	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-10	As a developer I want a log of entries by users	STRETCH TASKS	TO DO	[User]
DWD-11	Create the entity class	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-12	create the controller class	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-13	create the services class	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-14	create the repo	REQUIRED FUNCTIONALITY	DONE	[User]
DWD-15				
DWD-16				
DWD-17				
DWD-18				
DWD-19				
DWD-20				
DWD-21				
DWD-22				
DWD-23				
DWD-24				
DWD-25				

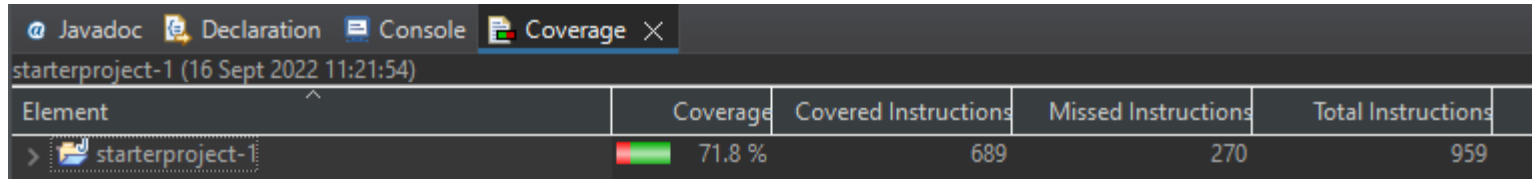
technology

- Spring
- Postman
- MYSQL
- Eclipse



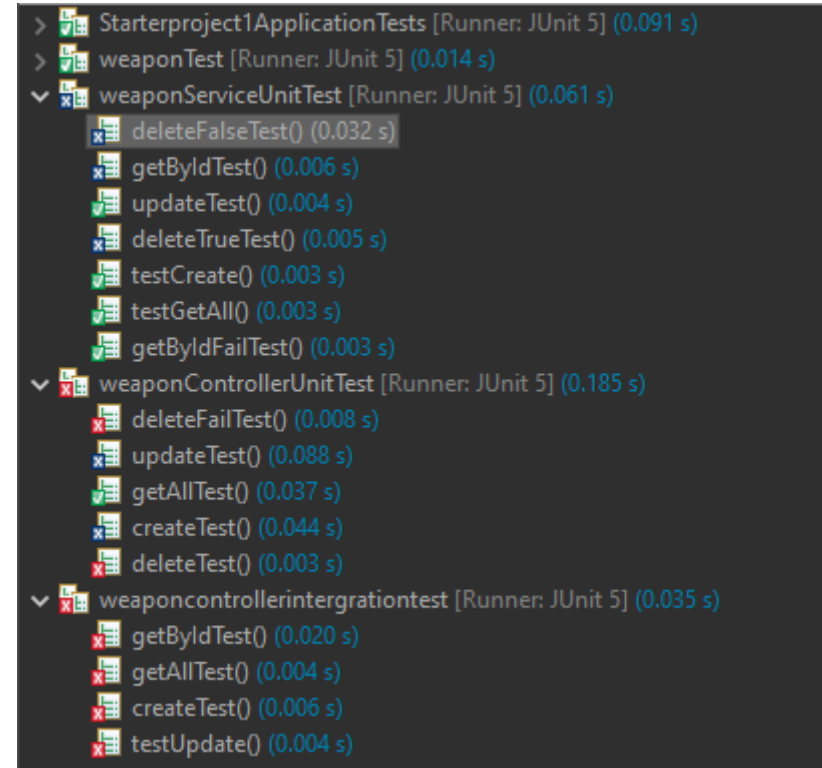
Testing

- Full testing is still required
- Small tests preformed for functionality have gone well on postman



The screenshot shows the Coverage window in an IDE. The title bar includes tabs for Javadoc, Declaration, Console, and Coverage. The main area displays a table for 'starterproject-1 (16 Sept 2022 11:21:54)'. The table has columns for Element, Coverage, Covered Instructions, Missed Instructions, and Total Instructions. The 'starterproject-' element is highlighted with a red and green progress bar indicating 71.8% coverage.

Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
> starterproject-	71.8 %	689	270	959



The screenshot shows the Test Results window in an IDE. It lists several test cases with their status (pass/fail) and duration. The tests are grouped under 'Starterproject1ApplicationTests', 'weaponTest', 'weaponServiceUnitTest', 'weaponControllerUnitTest', and 'weaponcontrollerintergrationtest'.

Test Case	Runner	Duration	Status
Starterproject1ApplicationTests	JUnit 5	0.091 s	Pass
weaponTest	JUnit 5	0.014 s	Pass
weaponServiceUnitTest	JUnit 5	0.061 s	Pass
deleteFalseTest()		0.032 s	Fail
getByIdTest()		0.006 s	Fail
updateTest()		0.004 s	Pass
deleteTrueTest()		0.005 s	Fail
testCreate()		0.003 s	Pass
testGetAll()		0.003 s	Pass
getByIdFailTest()		0.003 s	Pass
weaponControllerUnitTest	JUnit 5	0.185 s	Pass
deleteFailTest()		0.008 s	Fail
updateTest()		0.088 s	Fail
getAllTest()		0.037 s	Pass
createTest()		0.044 s	Fail
deleteTest()		0.003 s	Fail
weaponcontrollerintergrationtest	JUnit 5	0.035 s	Pass
getByIdTest()		0.020 s	Fail
getAllTest()		0.004 s	Fail
createTest()		0.006 s	Fail
testUpdate()		0.004 s	Fail

Sprint review

- Could have been better
- Should have been reordered better

conclusions

- Should have asked more questions
- Organization should have been better
- Next steps
- Enjoyed the project