TDL Project

Hamza Pervaiz

Introduction

About

Background in Computer Networks and Security. Previous experience with languages such as PHP, SQL, HTML/CSS.

Project Approach

The project specification was initially analyzed from requirements to deliverables checklist.

Used Cats + Houses API training as base starting point to understand functionality and flow of the application.

Training linking Cats and Houses via API also assisted.

Sprint Plan

Listed all requirements, user stories, acceptance criteria and prioritized using Moscow analysis. Utilization of Kanban to track and record (Jira/Trello). All completed prior to starting project implementation.

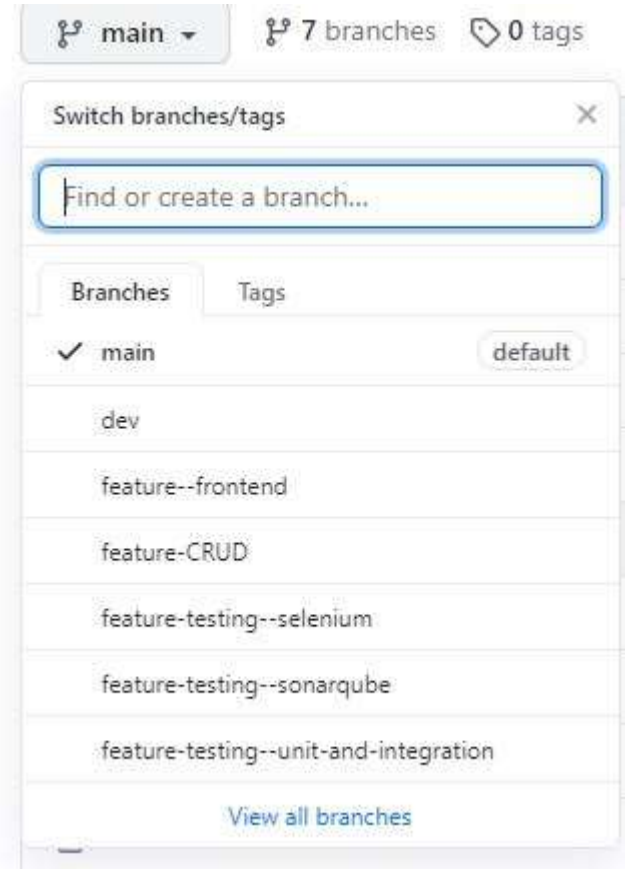
Consultant Journey

- Technologies learned and used for this project:

- 1. Java
- 2. Junit and Mockito (Unit and Integration Testing)
- 3. GitHub and Spring Tool Suite
- 4. Maven
- 5. SonarQube (Static Analysis)
- 6. Selenium (User Acceptance Testing)
- 7. Spring Boot
- 8. HTML/CSS/JS


Continuous Integration/ Version Control

- GitHub used for Version Control, using the feature-<concept> model. Features developed incrementally and added to each branch.



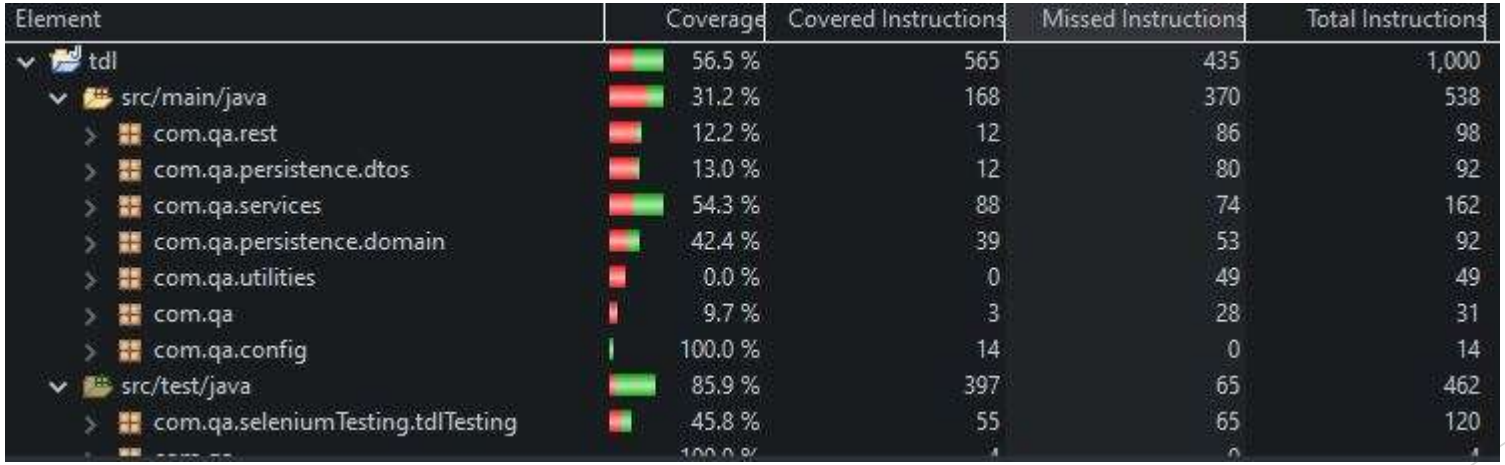
https://www.github.com/hamzapQA/tdl

Testing – Unit and Integration

- Carried out on following classes:

TDL (Domain)

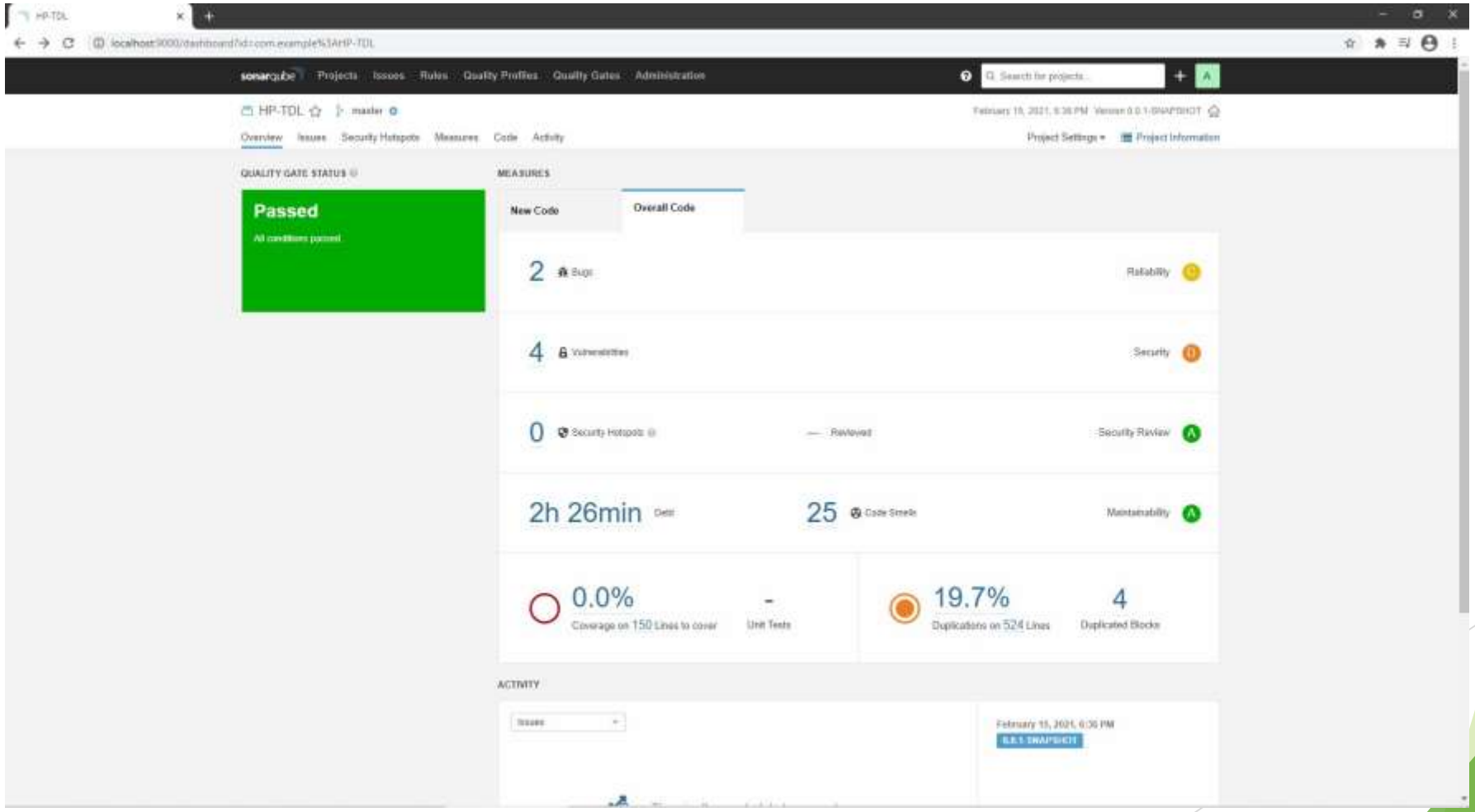
TDL (Services)



Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
tdl	56.5 %	565	435	1,000
src/main/java	31.2 %	168	370	538
com.qa.rest	12.2 %	12	86	98
com.qa.persistence.dtos	13.0 %	12	80	92
com.qa.services	54.3 %	88	74	162
com.qa.persistence.domain	42.4 %	39	53	92
com.qa.utilities	0.0 %	0	49	49
com.qa	9.7 %	3	28	31
com.qa.config	100.0 %	14	0	14
src/test/java	85.9 %	397	65	462
com.qa.seleniumTesting.tdlTesting	45.8 %	55	65	120

Testing – Static Analysis - SonarQube

- Code scanned with SonarQube result:



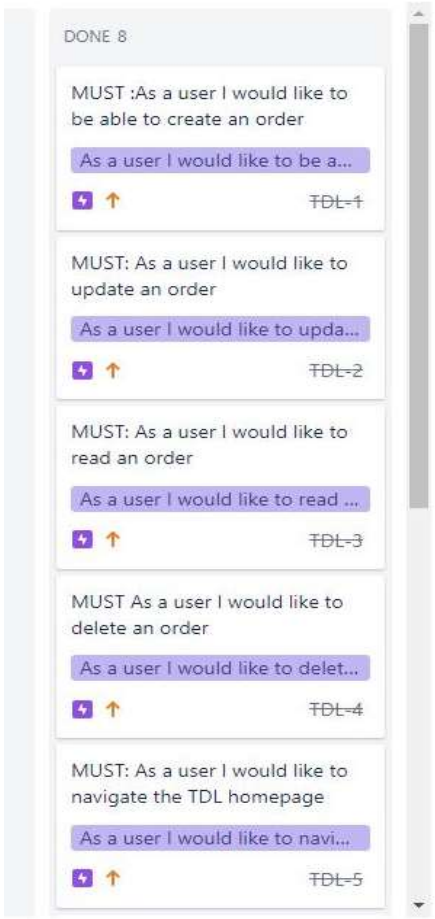
Testing – User Acceptance - Selenium

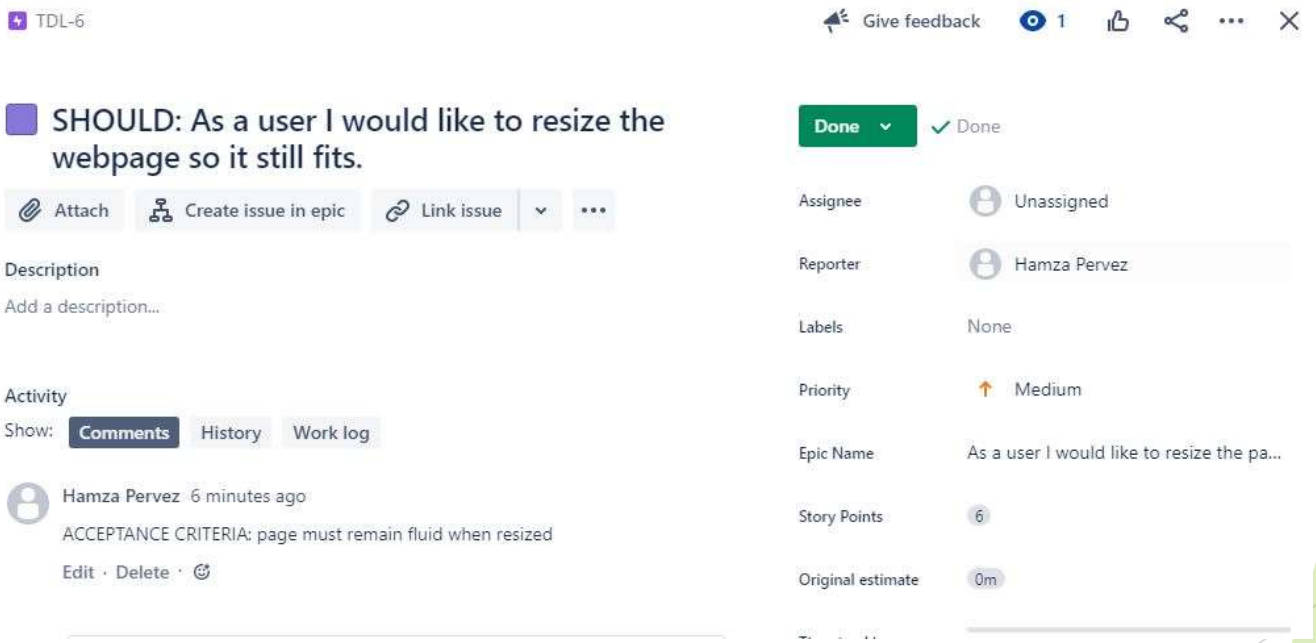
- Sample test:

```
@Test
public void createToDo() {

    test = report.startTest("Create To Do Test");

    driver.get(URL);
    //clicking create button via xpath
    targ = driver.findElement(By.xpath("/html/body/button"));
    targ.click();

    //entering example data
    targ = driver.findElement(By.id("toDoTitle"));
    targ.sendKeys("CREATE - Selenium User Acceptance Test");

    targ = driver.findElement(By.xpath("//*[@id=\"createbutton\"]/div/div/div[3]/button"));
    targ.click();

    // checking if it was successful
    targ = new WebDriverWait(driver, 5).until(ExpectedConditions.presenceOfElementLocated(By.xpath("//*[@id=\"createConfirm\"]/p")));
    String result = targ.getText();
    String expected = "has been created. Head over to the Read page and click Read all to view the task ID";

    if(result.contains(expected)) {
        test.log(LogStatus.PASS, expected);
    }else {
        test.log(LogStatus.FAIL, "failed creation");
    }

    //assertions
    assertThat(result.concat(expected));
}
```

Planning – User Stories

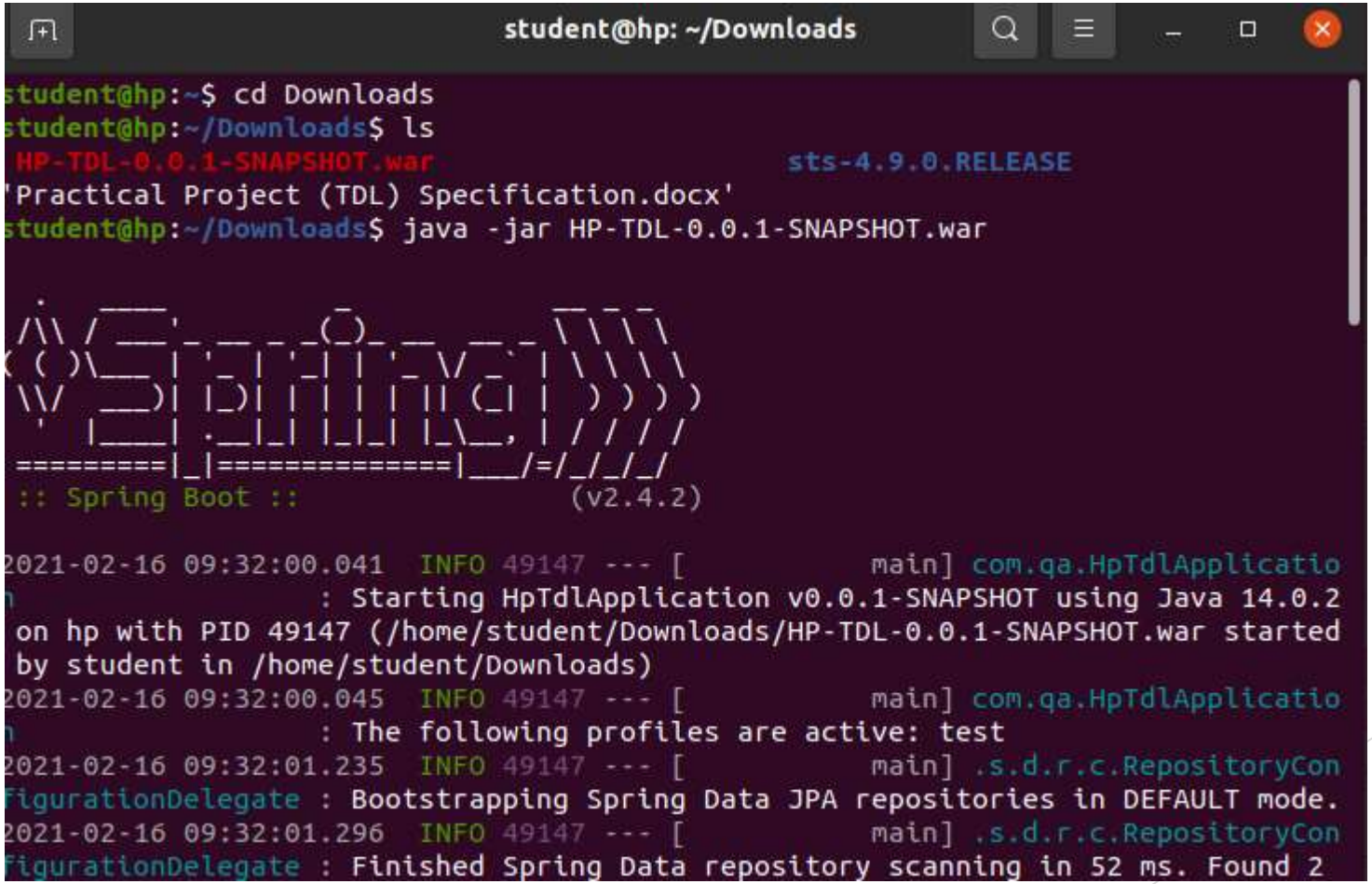
- Analysis of project specification, obtaining requirements and creating defined user stories with acceptance criteria.





Demonstration

- User Stories



Sprint Review

Completed:

- CRUD functionality – back end.
- CRUD functionality – front end
- GitHub repo and branches used.
- Codebase completed.
- .WAR file which can be deployed via the command line.

Not completed:

- - Further testing.
- - CRUD with second database entity.
- - GCP MySQL instance integration.

Sprint Retrospective

- IMS-Demo starting point helped a lot as it allowed to understand initial functionality of the package structure.
- Cat/Domain API training assisted with understanding.
- Band-API musician task assisted with front-end development, DOM attributes etc.
- GitHub version control helped to easily revert to amend errors.
- Access to recorded material on MS Teams helped.

Could Improve:

- Web Design – add more functionality
- Help Documentation
- Encryption – logging into DB.
- Include more database entities.
- More attributes in database associated with task.

Conclusion

- Project overall was successful as CRUD functionality achieved.
- Technologies and experiences learned will be valuable when developing new applications.
- Now able to develop an API with H2 database in Spring
- Now able to develop a working front-end.
- Project allowed understanding of how to plan future projects, (requirements, time management, resources).
- Will continue building on project in spare time to deepen knowledge and improve functionality.

Thank You