20DECSDET To-Do-List Project

Presented by: Arsalan Asad

Introduction

About me:

- Studied Chemical Engineering
- First job was in coding
- Developed passion for coding and how technology is advancing with coding
- I plan to use the knowledge I am constantly gaining to one day to something for myself.

The specification:

- From initial looks at the presentation I broke it down into the following steps.
- Plan – the priority was to write down a plan and visualise how I wanted my project to look
- Beginning – the next section was to look at my strengths and choose a starting point.
- Middle- this part was to make sure I began to tackle the challenging parts.
- End- To wrap everything up and have a finished product.

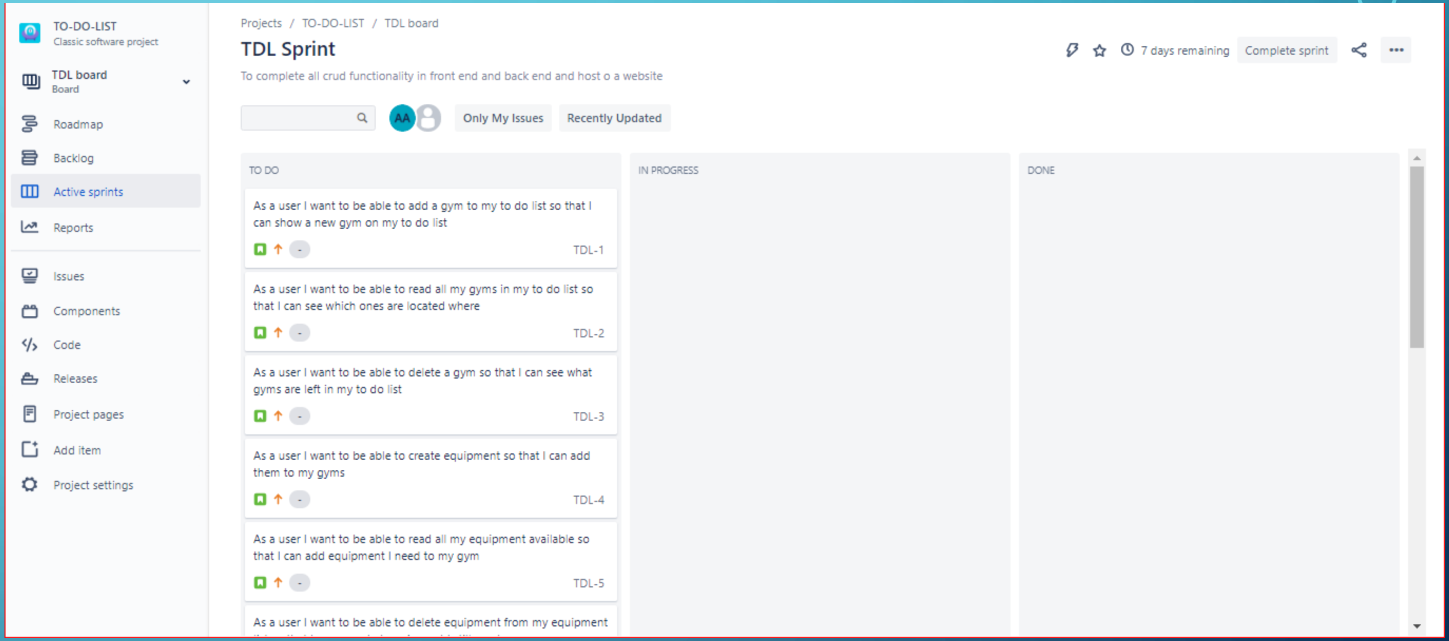
Concept

To start off I made sure that I understood the relationship between the two entities that I would be creating for my to-do-list. This was done to gain a better understanding of what would be required from me during implementation of code.

Following this, my key strengths from this project going in were the use of Java in the back end along with my sql, as well as testing. My more weaker points were the use of html and css to design a nice website. Therefore, I first started off with my strengths completing a working back end and complete tests for the back end and then tackled the html, javascript and css section along with automation testing for it.

Sprint Plan

- Has crud functionality been implemented?
- As a user…I want…so that…
- Cover all aspects of back end and front end
- Finally, a README and presentation task.



TDL Sprint

To complete all crud functionality in front end and back end and host o a website

As a user I want to be able to add a gym to my to do list so that I can show a new gym on my to do list — TDL-1

As a user I want to be able to read all my gyms in my to do list so that I can see which ones are located where — TDL-2

As a user I want to be able to delete a gym so that I can see what gyms are left in my to do list — TDL-3

As a user I want to be able to create equipment so that I can add them to my gyms — TDL-4

As a user I want to be able to read all my equipment available so that I can add equipment I need to my gym — TDL-5

As a user I want to be able to delete equipment from my equipment

Consultant Journey

This project has required me to use several new technologies allowing me to further build upon my knowledge and sharpen my skills required to be a consultant.

The following were used in this project:

- Java
- Javascript
- HTML
- CSS
- Swagger
- Postman
- SQL
- Spring
- Selenium

CI

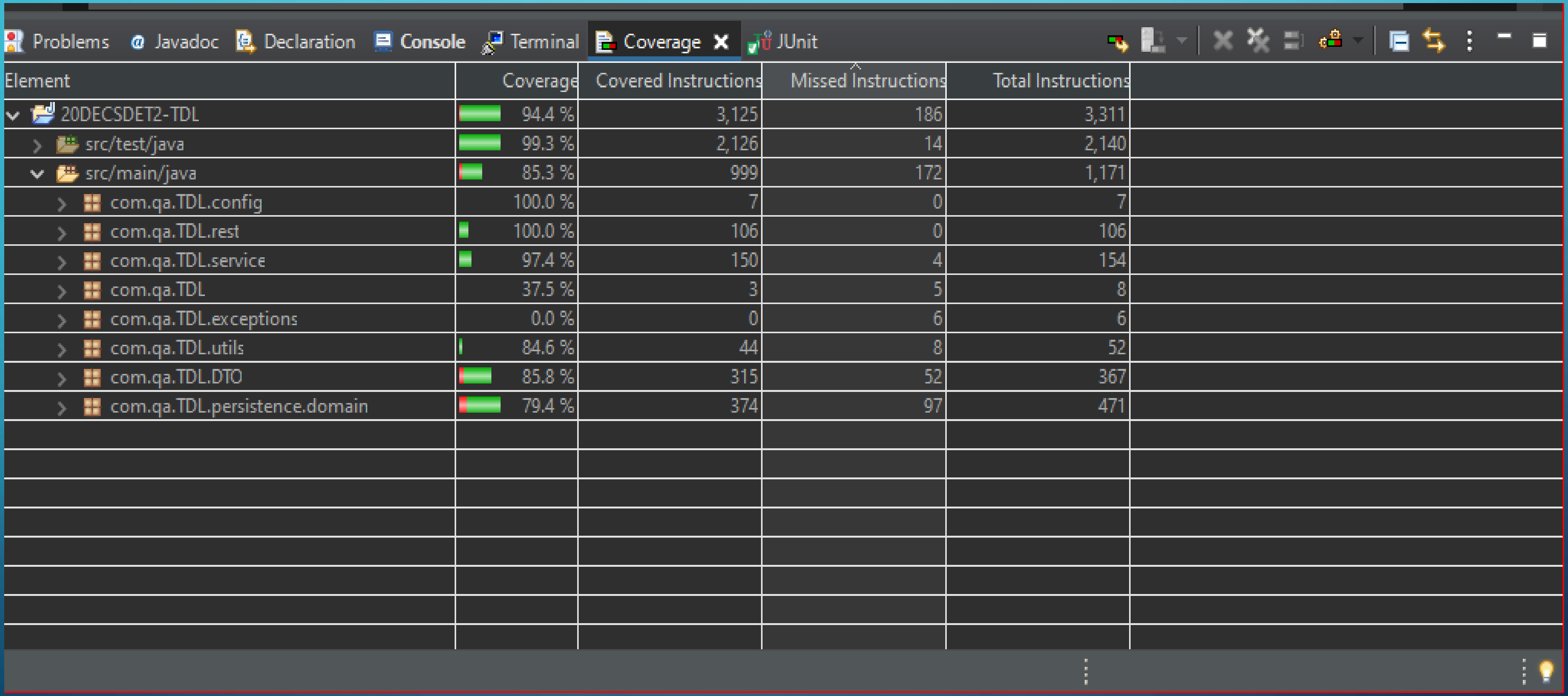
The CI, or version control, used throughout this project was Git. Git is used in conjunction with GitHub so that I could split my work into multiple feature branches in which one can see the stages of my project and what had been completed during each stage.

The approach was done by first creating a repository in my GitHub account and then cloning that repo into my project and initialising my project as a git project.

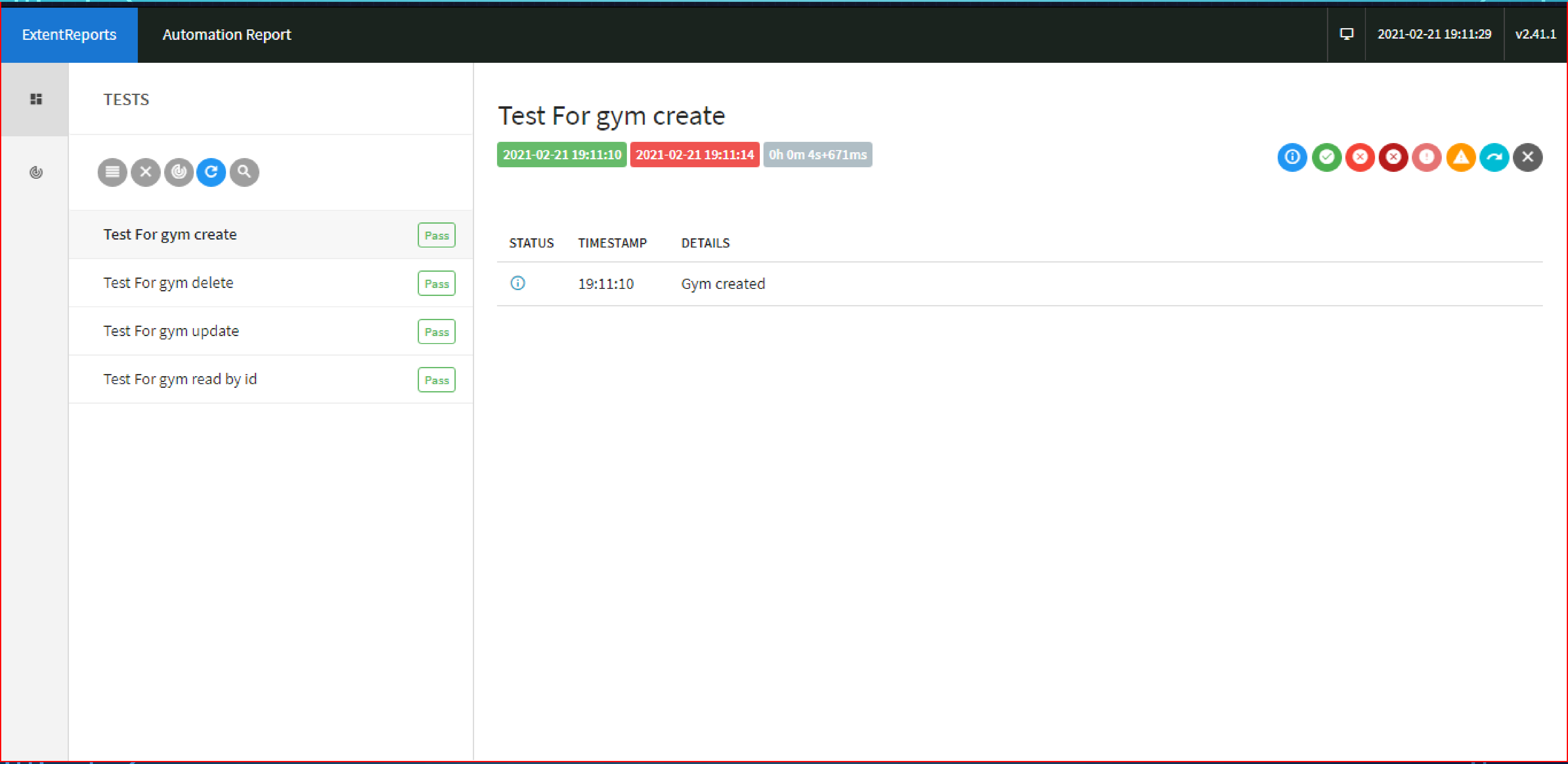
Following this, the next stages were creating feature branches when each of my user stories on Jira were complete and pushing all these changes up to my GitHub.

Testing

Use of Junit, selenium and extent reports.



Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
20DECSDET2-TDL	94.4 %	3,125	186	3,311
src/test/java	99.3 %	2,126	14	2,140
src/main/java	85.3 %	999	172	1,171
com.qa.TDL.config	100.0 %	7	0	7
com.qa.TDL.rest	100.0 %	106	0	106
com.qa.TDL.service	97.4 %	150	4	154
com.qa.TDL	37.5 %	3	5	8
com.qa.TDL.exceptions	0.0 %	0	6	6
com.qa.TDL.utils	84.6 %	44	8	52
com.qa.TDL.DTO	85.8 %	315	52	367
com.qa.TDL.persistence.domain	79.4 %	374	97	471

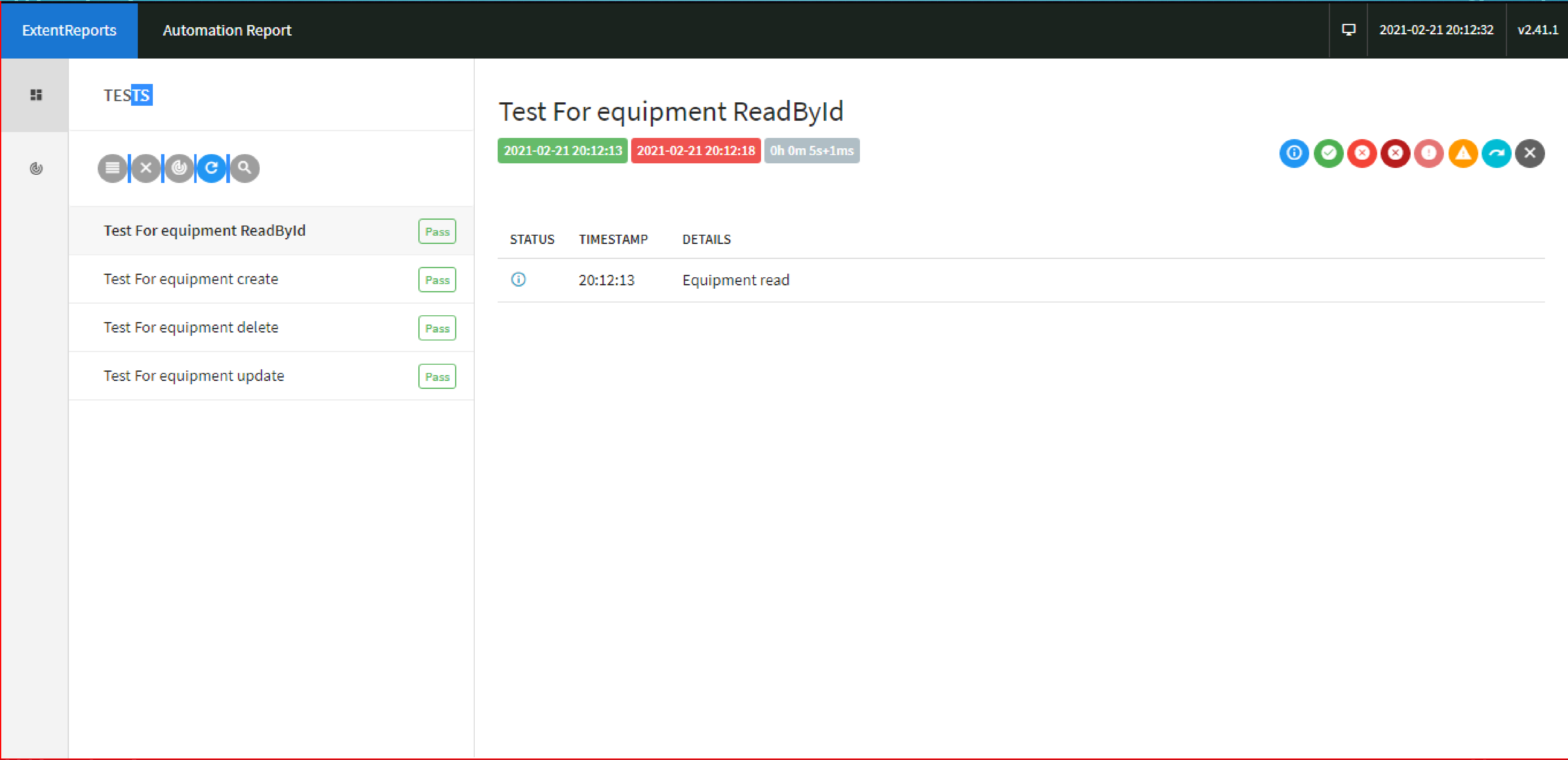


Test For gym create	Pass
Test For gym delete	Pass
Test For gym update	Pass
Test For gym read by id	Pass

Test For gym create

2021-02-21 19:11:10 2021-02-21 19:11:14 0h 0m 4s+671ms

STATUS	TIMESTAMP	DETAILS
ⓘ	19:11:10	Gym created



Test For equipment ReadById Pass

Test For equipment create Pass

Test For equipment delete Pass

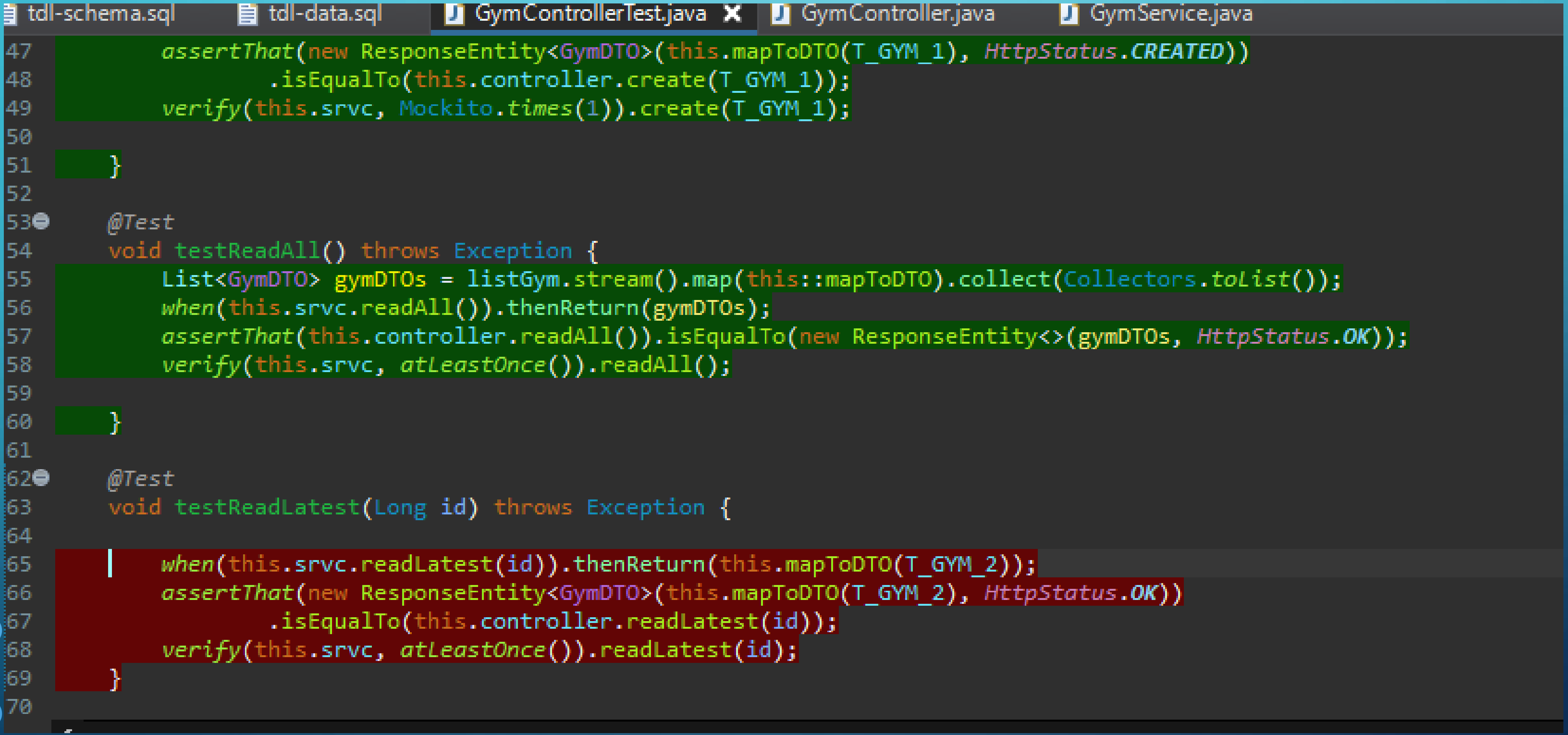
Test For equipment update Pass

Test For equipment ReadById

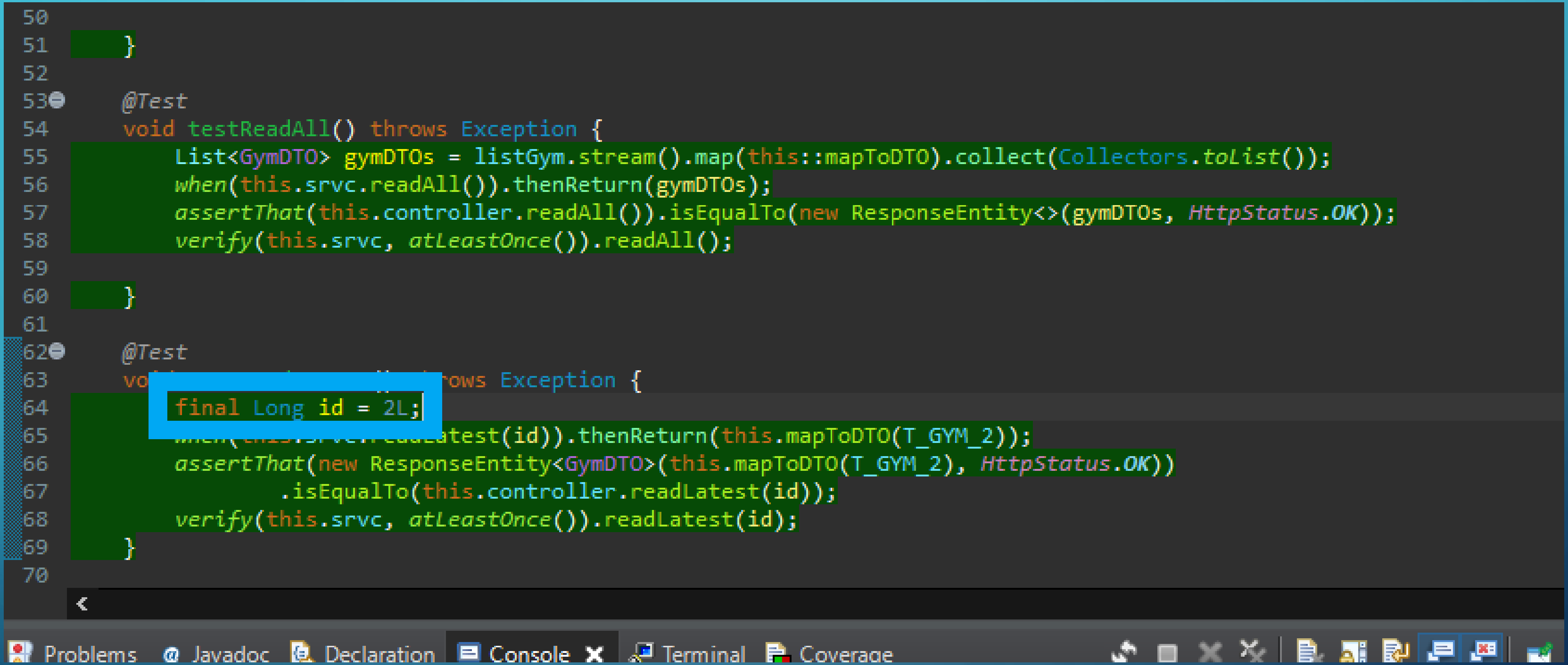
2021-02-21 20:12:13 2021-02-21 20:12:18 0h 0m 5s+1ms

STATUS	TIMESTAMP	DETAILS
ⓘ	20:12:13	Equipment read

Testing Error Handling



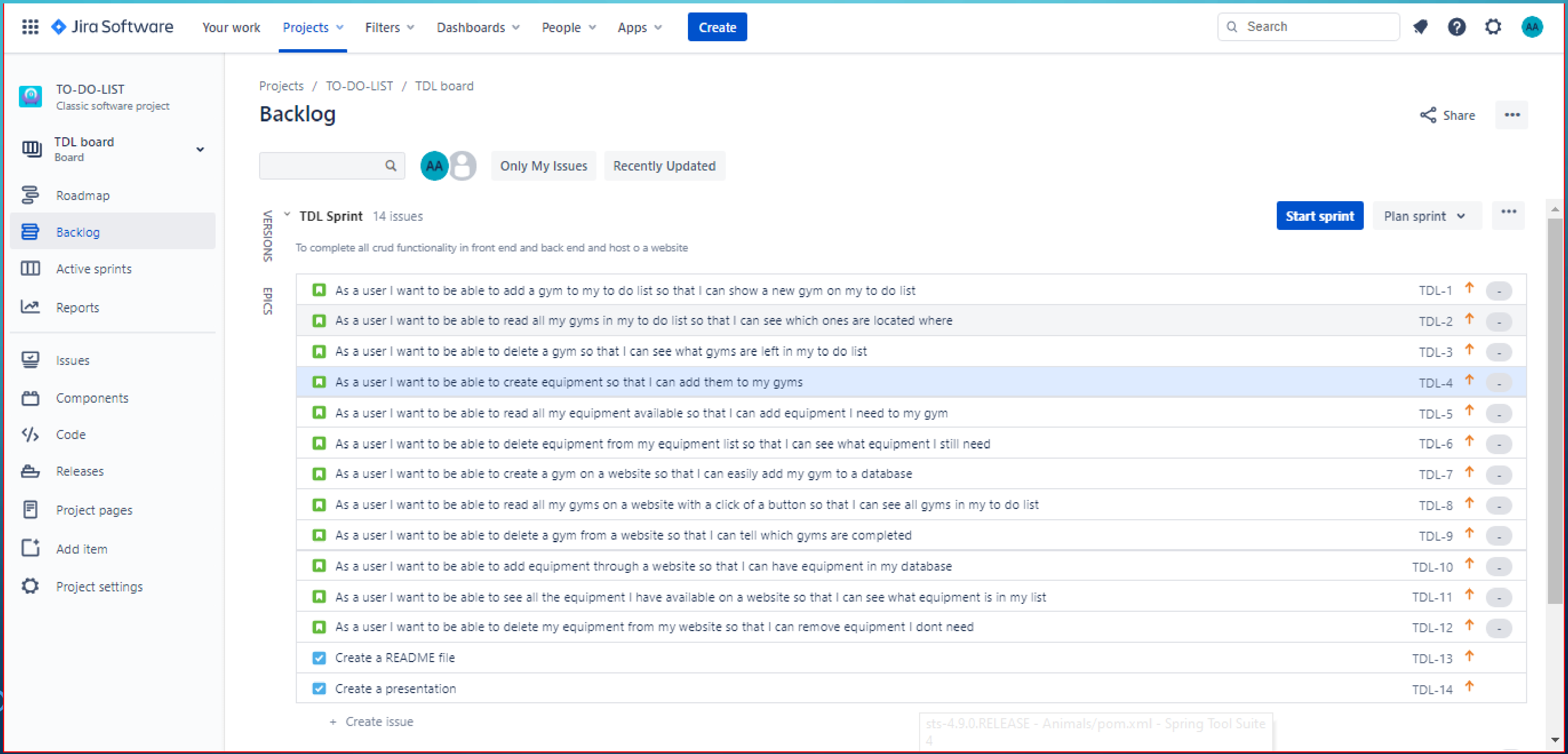
```
        assertThat(new ResponseEntity<GymDTO>(this.mapToDTO(T_GYM_1), HttpStatus.CREATED))
                .isEqualTo(this.controller.create(T_GYM_1));
        verify(this.srvc, Mockito.times(1)).create(T_GYM_1);

    }

    @Test
    void testReadAll() throws Exception {
        List<GymDTO> gymDTOs = listGym.stream().map(this::mapToDTO).collect(Collectors.toList());
        when(this.srvc.readAll()).thenReturn(gymDTOs);
        assertThat(this.controller.readAll()).isEqualTo(new ResponseEntity<>(gymDTOs, HttpStatus.OK));
        verify(this.srvc, atLeastOnce()).readAll();

    }

    @Test
    void testReadLatest(Long id) throws Exception {

        when(this.srvc.readLatest(id)).thenReturn(this.mapToDTO(T_GYM_2));
        assertThat(new ResponseEntity<GymDTO>(this.mapToDTO(T_GYM_2), HttpStatus.OK))
                .isEqualTo(this.controller.readLatest(id));
        verify(this.srvc, atLeastOnce()).readLatest(id);
    }

```



```java

    }

    @Test
    void testReadAll() throws Exception {
        List<GymDTO> gymDTOs = listGym.stream().map(this::mapToDTO).collect(Collectors.toList());
        when(this.srvc.readAll()).thenReturn(gymDTOs);
        assertThat(this.controller.readAll()).isEqualTo(new ResponseEntity<>(gymDTOs, HttpStatus.OK));
        verify(this.srvc, atLeastOnce()).readAll();

    }

    @Test
    vo                    rows Exception {
        final Long id = 2L;
        when(this.srvc.readLatest(id)).thenReturn(this.mapToDTO(T_GYM_2));
        assertThat(new ResponseEntity<GymDTO>(this.mapToDTO(T_GYM_2), HttpStatus.OK))
                .isEqualTo(this.controller.readLatest(id));
        verify(this.srvc, atLeastOnce()).readLatest(id);
    }

```

Sprint Review

Overall, I completed all the requirements of my sprint and user stories created through Jira. The next few slides will depict the start through to the end of my sprint.



Projects / TO-DO-LIST / TDL board

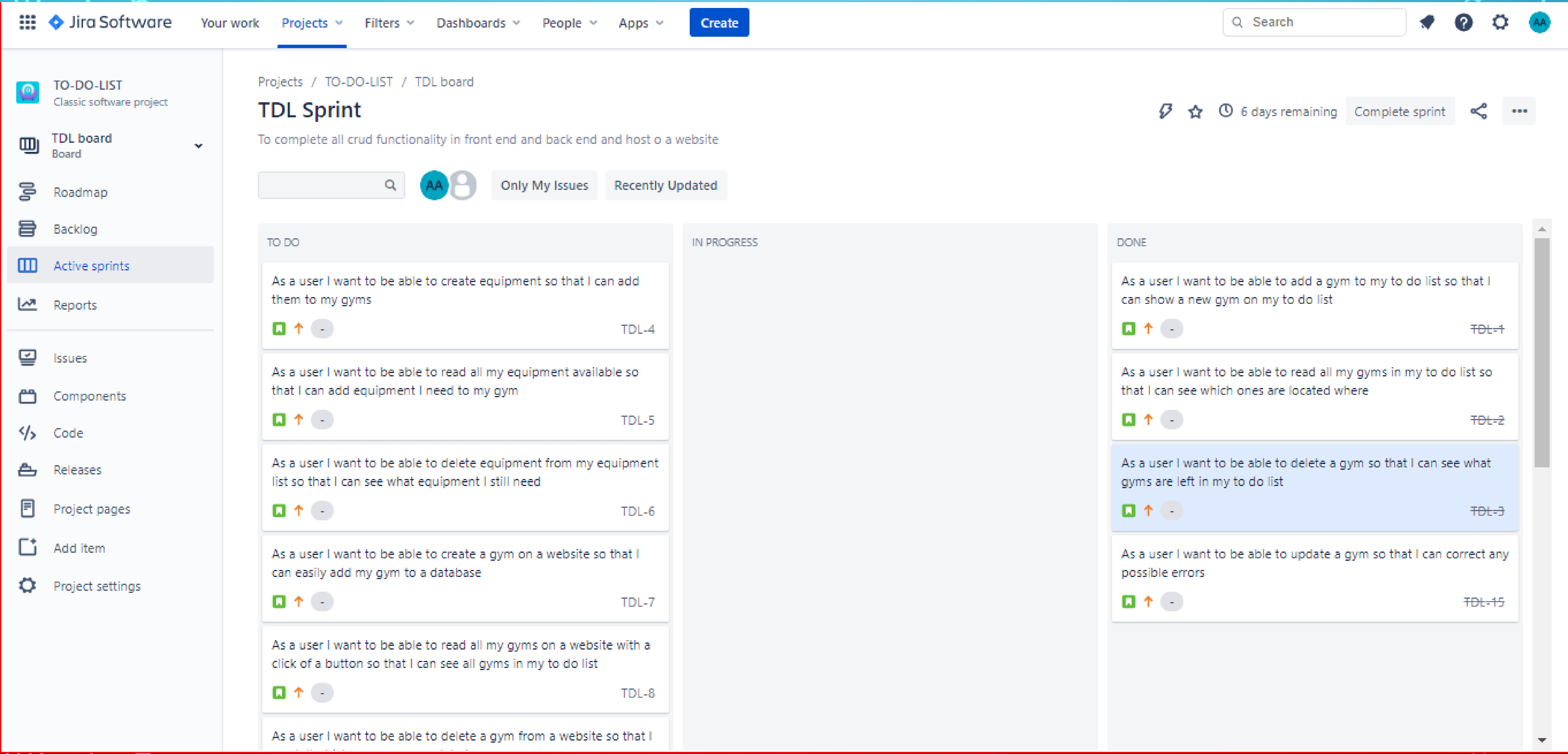
Backlog

TDL Sprint 14 issues

To complete all crud functionality in front end and back end and host o a website

As a user I want to be able to add a gym to my to do list so that I can show a new gym on my to do list	TDL-1
As a user I want to be able to read all my gyms in my to do list so that I can see which ones are located where	TDL-2
As a user I want to be able to delete a gym so that I can see what gyms are left in my to do list	TDL-3
As a user I want to be able to create equipment so that I can add them to my gyms	TDL-4
As a user I want to be able to read all my equipment available so that I can add equipment I need to my gym	TDL-5
As a user I want to be able to delete equipment from my equipment list so that I can see what equipment I still need	TDL-6
As a user I want to be able to create a gym on a website so that I can easily add my gym to a database	TDL-7
As a user I want to be able to read all my gyms on a website with a click of a button so that I can see all gyms in my to do list	TDL-8
As a user I want to be able to delete a gym from a website so that I can tell which gyms are completed	TDL-9
As a user I want to be able to add equipment through a website so that I can have equipment in my database	TDL-10
As a user I want to be able to see all the equipment I have available on a website so that I can see what equipment is in my list	TDL-11
As a user I want to be able to delete my equipment from my website so that I can remove equipment I dont need	TDL-12
Create a README file	TDL-13
Create a presentation	TDL-14

sts-4.9.0.RELEASE - Animals/pom.xml - Spring Tool Suite 4



TDL Sprint

To complete all crud functionality in front end and back end and host o a website

TO DO

As a user I want to be able to create equipment so that I can add them to my gyms TDL-4

As a user I want to be able to read all my equipment available so that I can add equipment I need to my gym TDL-5

As a user I want to be able to delete equipment from my equipment list so that I can see what equipment I still need TDL-6

As a user I want to be able to create a gym on a website so that I can easily add my gym to a database TDL-7

As a user I want to be able to read all my gyms on a website with a click of a button so that I can see all gyms in my to do list TDL-8

As a user I want to be able to delete a gym from a website so that I

IN PROGRESS

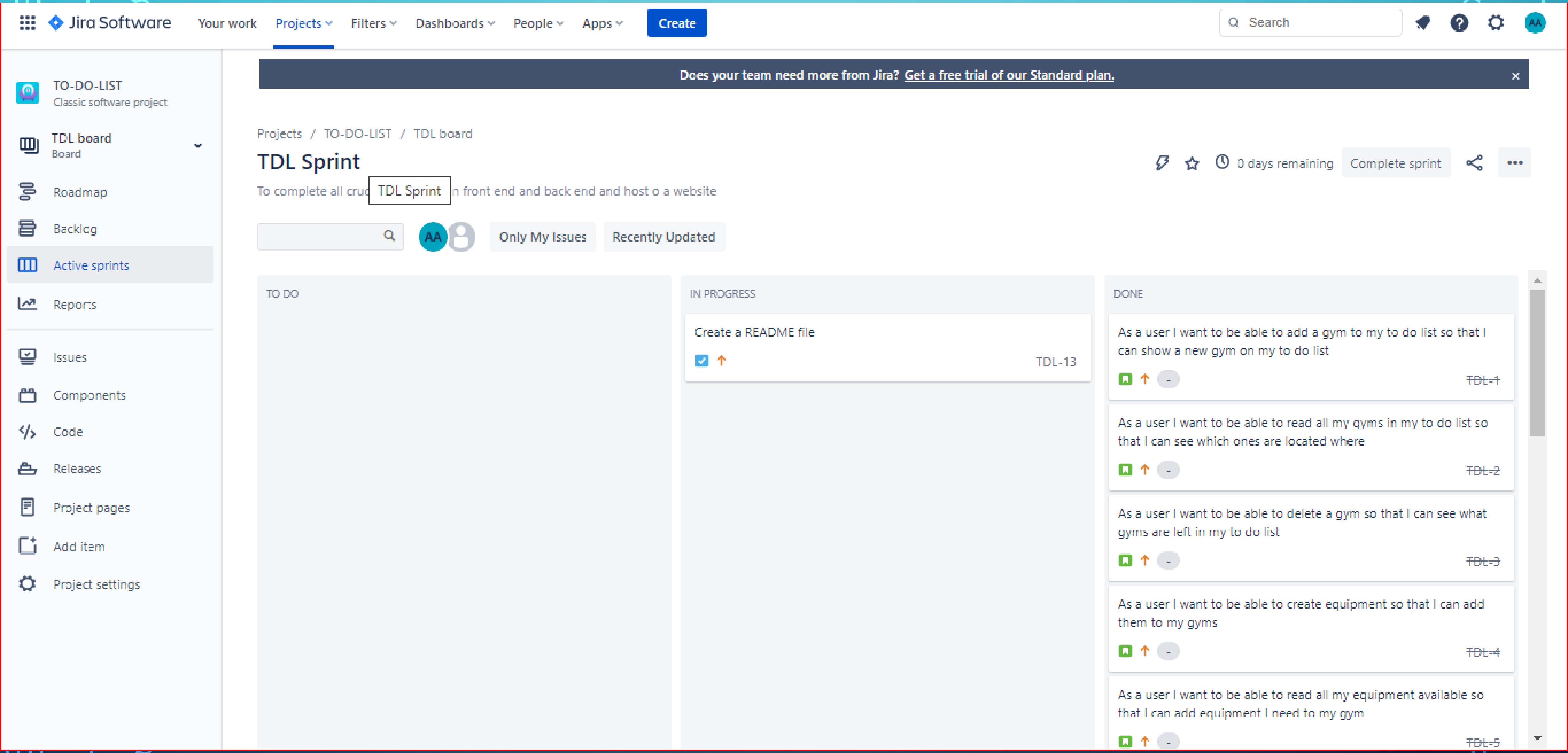
DONE

As a user I want to be able to add a gym to my to do list so that I can show a new gym on my to do list ~~TDL-1~~

As a user I want to be able to read all my gyms in my to do list so that I can see which ones are located where ~~TDL-2~~

As a user I want to be able to delete a gym so that I can see what gyms are left in my to do list ~~TDL-3~~

As a user I want to be able to update a gym so that I can correct any possible errors ~~TDL-15~~



TDL Sprint

To complete all cruc TDL Sprint n front end and back end and host o a website

TO DO

IN PROGRESS

Create a README file
☑ ↑ TDL-13

DONE

As a user I want to be able to add a gym to my to do list so that I can show a new gym on my to do list
↑ - ~~TDL-1~~

As a user I want to be able to read all my gyms in my to do list so that I can see which ones are located where
↑ - ~~TDL-2~~

As a user I want to be able to delete a gym so that I can see what gyms are left in my to do list
↑ - ~~TDL-3~~

As a user I want to be able to create equipment so that I can add them to my gyms
↑ - ~~TDL-4~~

As a user I want to be able to read all my equipment available so that I can add equipment I need to my gym
↑ - ~~TDL-5~~

Sprint Retrospective

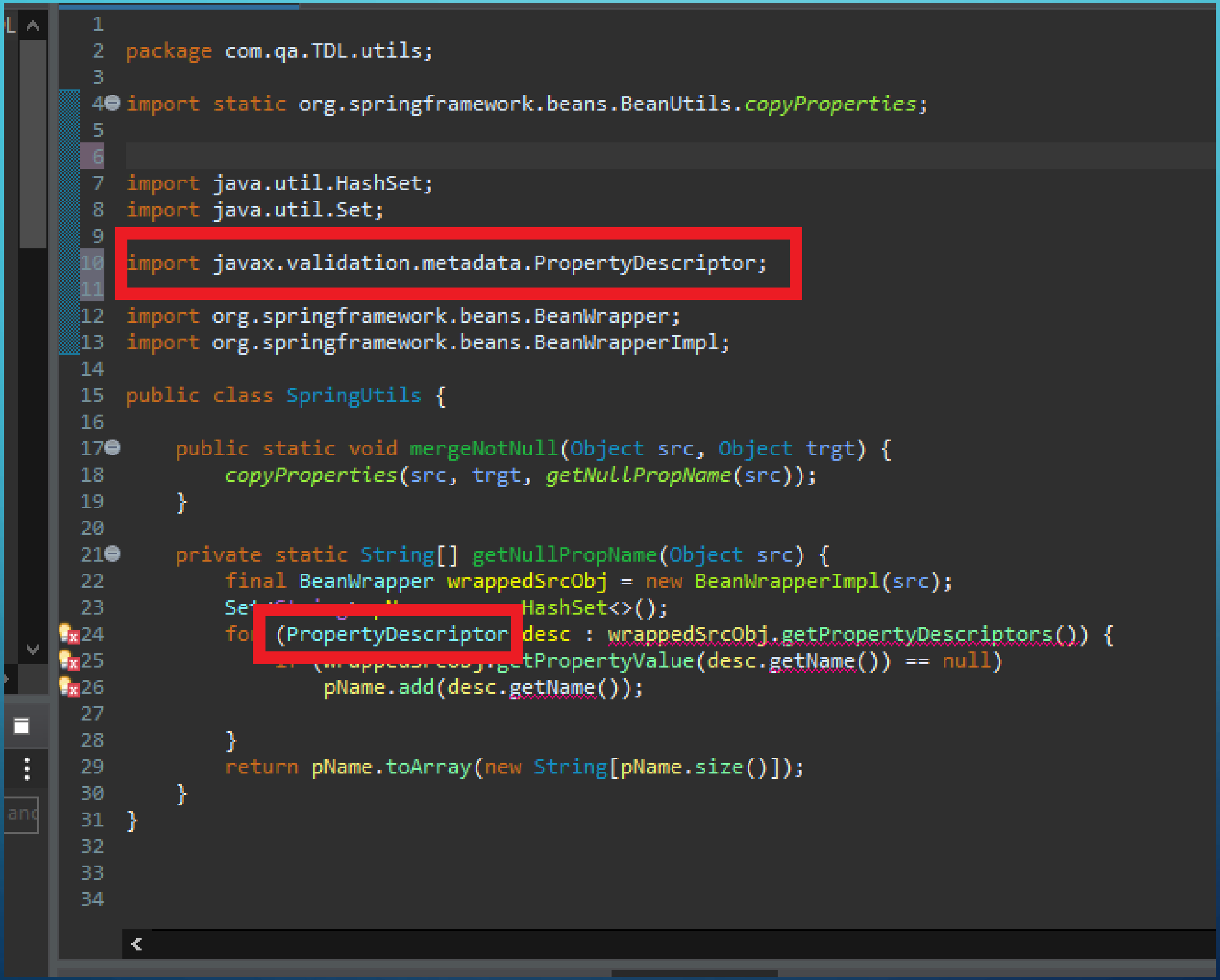
What went well:

- Managed to complete my entire sprint.
- Have a successful back-end and front-end working together.
- Testing coverage is over 80%.
- Managed to do selenium testing for my website and get extent reports.
- All feature branches on github mostly pushed correctly.

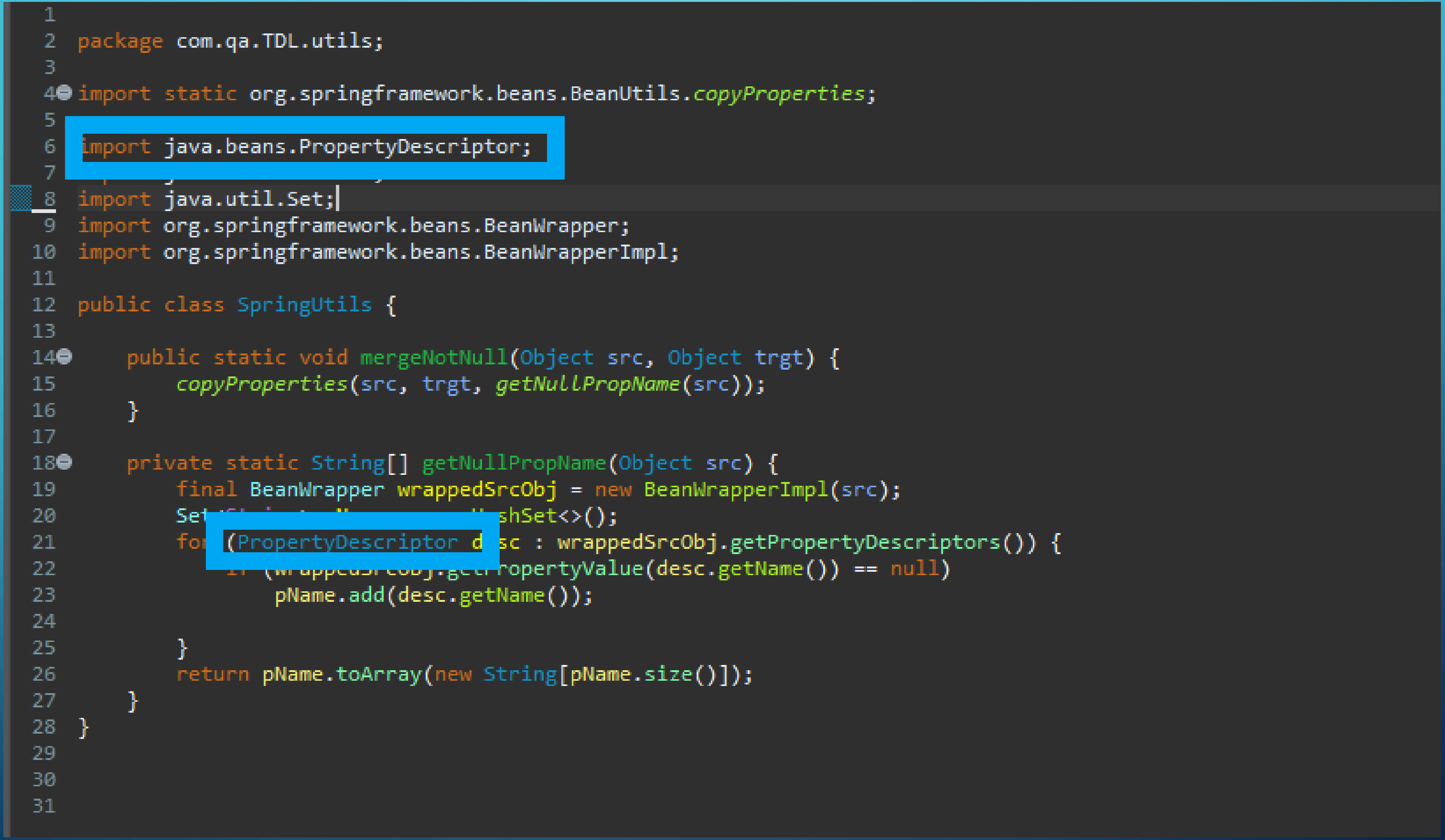
What could be improved:

- Use of CSS to design website better
- Thorough understanding of HTML to design a more suitable looking website.
- Testing for complete integration controllers and figuring out how to test for toString and hashCode.

More Error Handling

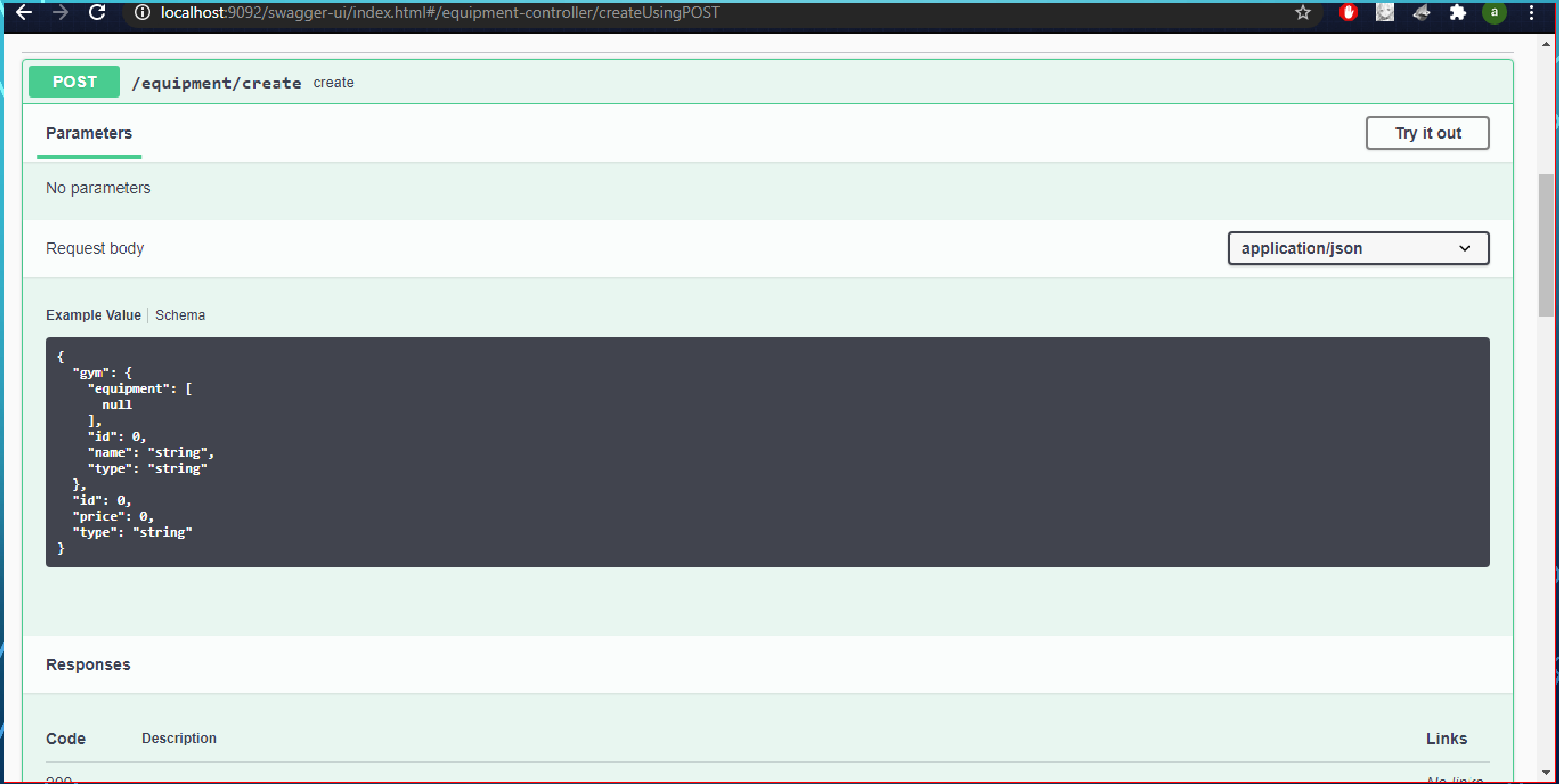


```java

package com.qa.TDL.utils;

import static org.springframework.beans.BeanUtils.copyProperties;


import java.util.HashSet;
import java.util.Set;

import javax.validation.metadata.PropertyDescriptor;

import org.springframework.beans.BeanWrapper;
import org.springframework.beans.BeanWrapperImpl;

public class SpringUtils {

    public static void mergeNotNull(Object src, Object trgt) {
        copyProperties(src, trgt, getNullPropName(src));
    }

    private static String[] getNullPropName(Object src) {
        final BeanWrapper wrappedSrcObj = new BeanWrapperImpl(src);
        Set<String> pName = new HashSet<>();
        for (PropertyDescriptor desc : wrappedSrcObj.getPropertyDescriptors()) {
            if (wrappedSrcObj.getPropertyValue(desc.getName()) == null)
                pName.add(desc.getName());

        }
        return pName.toArray(new String[pName.size()]);
    }
}
```



```java

package com.qa.TDL.utils;

import static org.springframework.beans.BeanUtils.copyProperties;

import java.beans.PropertyDescriptor;
import java.util.HashSet;
import java.util.Set;
import org.springframework.beans.BeanWrapper;
import org.springframework.beans.BeanWrapperImpl;

public class SpringUtils {

    public static void mergeNotNull(Object src, Object trgt) {
        copyProperties(src, trgt, getNullPropName(src));
    }

    private static String[] getNullPropName(Object src) {
        final BeanWrapper wrappedSrcObj = new BeanWrapperImpl(src);
        Set<String> pName = new HashSet<>();
        for (PropertyDescriptor desc : wrappedSrcObj.getPropertyDescriptors()) {
            if (wrappedSrcObj.getPropertyValue(desc.getName()) == null)
                pName.add(desc.getName());

        }
        return pName.toArray(new String[pName.size()]);
    }
}



```

The Use of Swagger-UI



localhost:9092/swagger-ui/index.html#/equipment-controller/createUsingPOST

POST `/equipment/create` create

Parameters

Try it out

No parameters

Request body

application/json

Example Value | Schema

```
{
  "gym": {
    "equipment": [
      null
    ],
    "id": 0,
    "name": "string",
    "type": "string"
  },
  "id": 0,
  "price": 0,
  "type": "string"
}
```

Responses

Code	Description	Links
200		No links

Conclusion

To conclude, this project has been able to demonstrate my growth from the very beginning of my QA journey to where I am now. My knowledge of testing and Java has greatly increased and I have been able to show that with this project. I have also been able to take upon new ideas such as HTML, selenium and javascript to create a functioning website. However, I would like to spend more time on researching HTML and javascript to improve upon designing of a website and its methods.
For the future, I need to make sure that I set more time for tasks I find more difficult. As well as this, I need to ensure that I am able to test using selenium by the pom method taught and remove the use of Thread.sleep.

Overall, I believe that my development throughout this course has sharpened my skills and knowledge required to become an excellent consultant.

Thank You