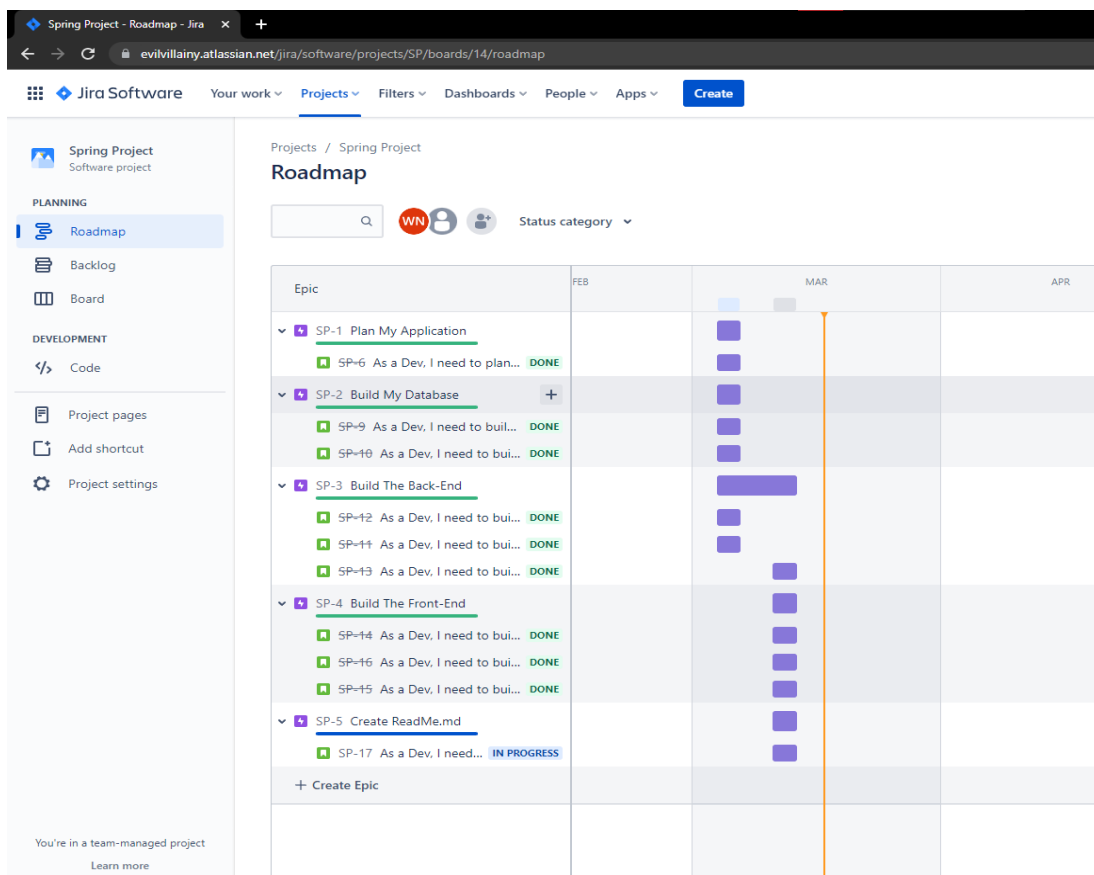


Spring Project - The Trainer Database.

This project brief was to create a CRUD application that used all of the modules covered in training. I decided to create a system that allowed a user to add trainers to a database. This could be used by a collector to show his collection of trainers.

Planning Resources;

I used Jira, to create a Kanban board, I then created Epics and user stories. From the user stories, I created child issues that need to be worked on. I also added back end and front end epics so they were easy to break down.



Jira Board Link;

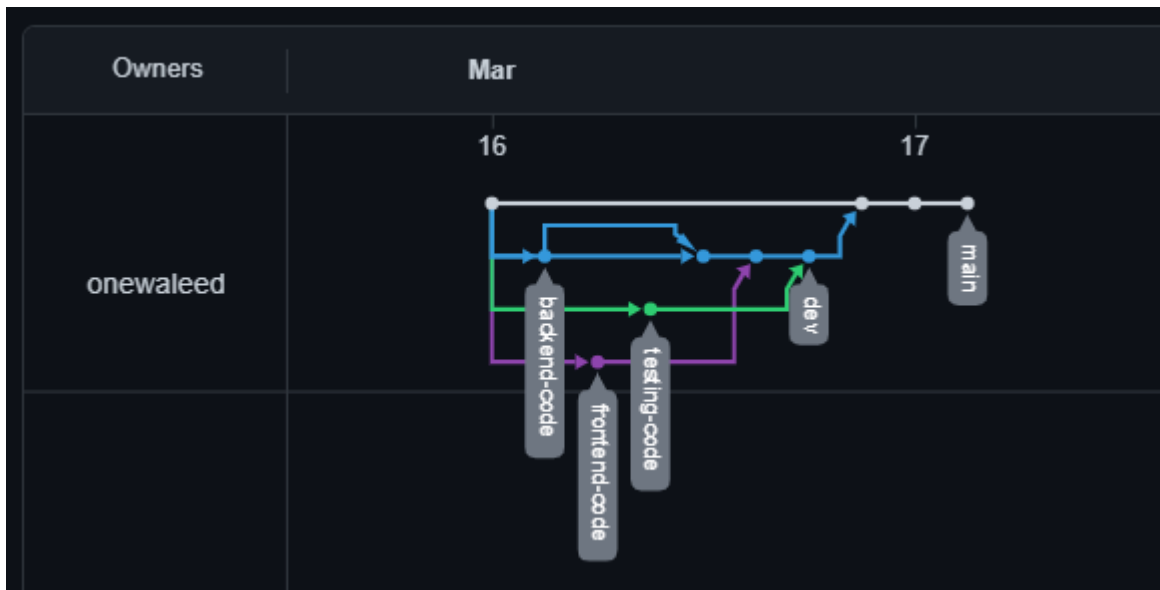
<https://evilvillainy.atlassian.net/jira/software/projects/SP/boards/14/roadmap?shared=&atlOrigin=eyJpIjoiYzY0NWQyYmYxZGEzNDBiODg1YWZjYWEwNTE0NTZmMzciLCJwIjoiIj9>

Git Hub Link;

<https://github.com/onewaleed/springprojectwal>

These are the demonstrated skills used in this project, in no particular order.

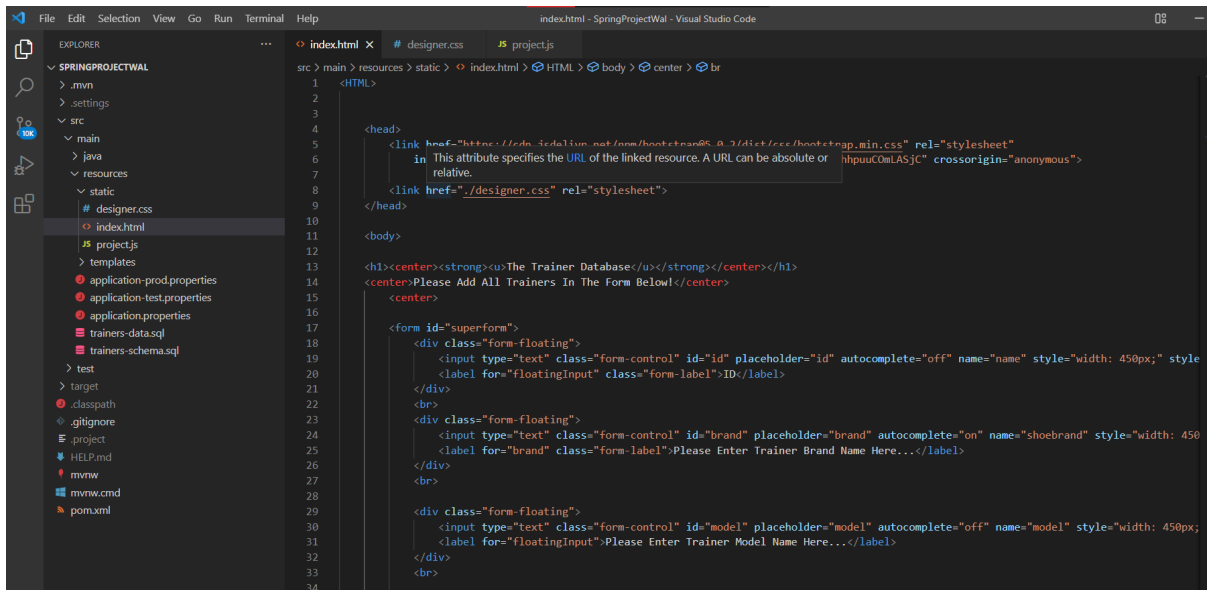
- Git Hub was used, I set up and used three branches. Including a dev branch.



- Live Studio, to see changes in real-time.

The screenshot shows a web application titled 'The Trainer Database' running on a local server at 127.0.0.1:5500. The application has a dark blue background and white text. It contains two main sections: 'Please Add All Trainers In The Form Below!' and 'Update Your Trainers Here'. The first section has four input fields for 'ID', 'Please Enter Trainer Brand Name Here...', 'Please Enter Trainer Model Name Here...', and 'Please Enter Trainer Colour Here...'. Below these fields are four buttons: 'Add Trainers', 'Delete Trainers', 'Read By Trainer ID', and 'See All Items'. The second section, 'Update Your Trainers Here', has a subtitle 'Use This Form To Update Any Vairables For A Trainer' and four input fields for 'ID', 'Brand', 'Model', and 'Colour'. An 'Update Trainer' button is at the bottom of this section.

- Visual Studio Code - HTML was used for the front end.



```
1 <HTML>
2
3
4
5 <head>
6   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
7     in This attribute specifies the URL of the linked resource. A URL can be absolute or
8     http://localhost:3000/ http://localhost:3000/ http://localhost:3000/ http://localhost:3000/ http://localhost:3000/
9   <link href="designer.css" rel="stylesheet">
10 </head>
11
12 <body>
13
14 <h1><center><strong><u>The Trainer Database</u></strong></center></h1>
15 <center>Please Add All Trainers In The Form Below!</center>
16 <center>
17   <form id="superform">
18     <div class="form-floating">
19       <input type="text" class="form-control" id="id" placeholder="id" autocomplete="off" name="name" style="width: 450px;" style
20       <label for="floatingInput" class="form-label">ID</label>
21     </div>
22     <br>
23     <div class="form-floating">
24       <input type="text" class="form-control" id="brand" placeholder="brand" autocomplete="on" name="shoebrand" style="width: 450
25       <label for="brand" class="form-label">Please Enter Trainer Brand Name Here...</label>
26     </div>
27     <br>
28     <div class="form-floating">
29       <input type="text" class="form-control" id="model" placeholder="model" autocomplete="off" name="model" style="width: 450px;
30       <label for="floatingInput">Please Enter Trainer Model Name Here...</label>
31     </div>
32   </form>
33 </body>
34
```

- JUnit was used for testing.

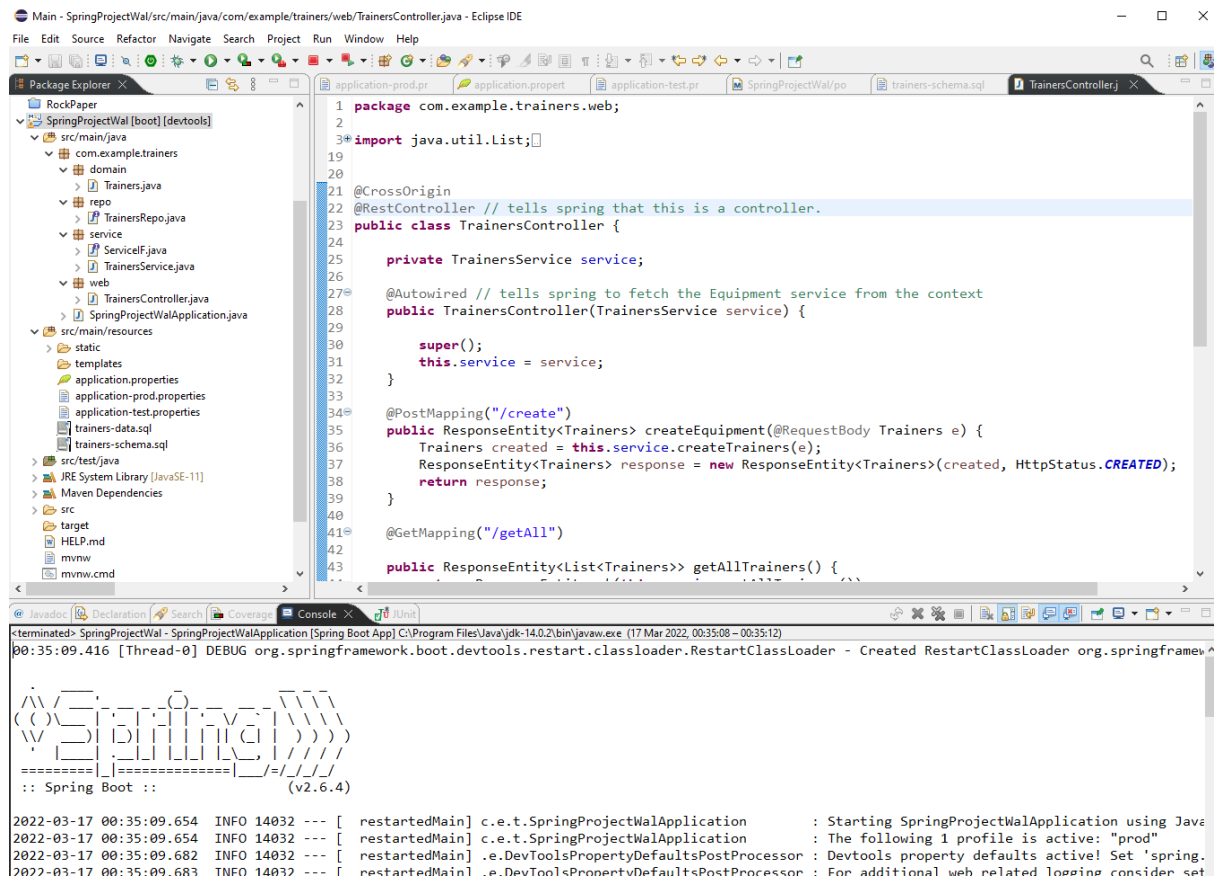


```
@ Javadoc Declaration Search Coverage Console JUnit X
Finished after 5.157 seconds

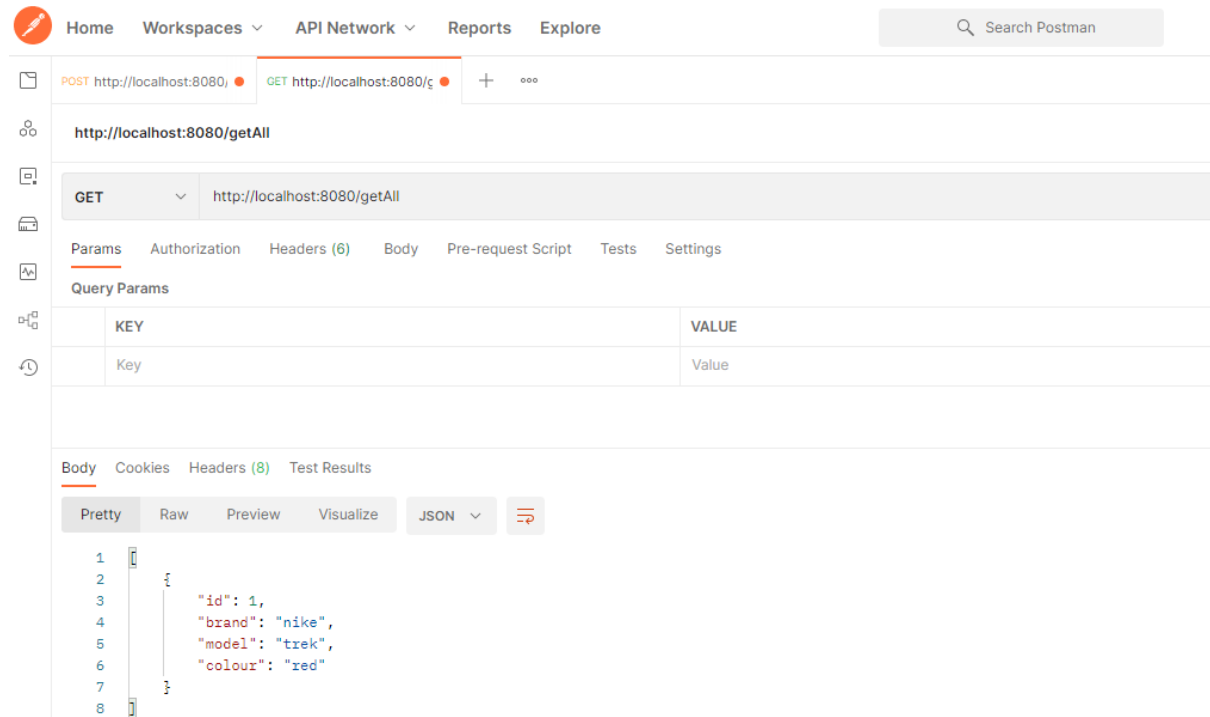
Runs: 6/6 Errors: 0 Failures: 0

SpringProjectWalApplicationTests [Runner: JUnit 5] (0.147 s)
  contextLoads() (0.147 s)
TrainersControllerIntegrationTest [Runner: JUnit 5] (0.272 s)
  getTest() (0.112 s)
  getAllTest() (0.082 s)
  testCreate() (0.046 s)
  testRemove() (0.017 s)
  testUpdate() (0.012 s)
```

- Eclipse IDE and SpringBoot was used to code the backend.



- Postman was used to test, the Post and Get functions on the local host.



- My SQL was used to store the data from the website and also used to create the tables.

