

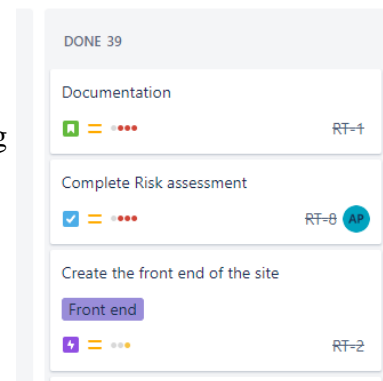
# Onur Hassan's Personal Project Portfolio

## Introduction

The aim of this portfolio is to document my personal participation towards the QA Cinema group project. The requirements of the project were to create a full stack application for "QA Cinema" while implementing agile team methods to achieve this, this included creating a SpringBoot application with back-end development, a user-friendly front-end with HTML, CSS and JavaScript and making them function with hosting services such as an AWS RDS.

## Day 1

The first day mainly consisted of deciding roles for the team in which I was given duties I both the back-end and front-end along with the creation of documentation and planning. We created a kanban board using Jira and all collaborated on deciding what user stories we needed to include to satisfy the MVP and other additions we felt we wanted. We then decided on when certain tasks needed to be completed by and then got to work with some basic tasks before ending off the day.



## Day 2

The main focus of the second day was to continue collaborating on creating documentation. However the majority of the day ended up consisting of back-end testing in which me and Danyel were in charge of testing the Java-SpringBoot based model features of the back-end in which we both collaborated in deciding what to include and make sure everything functioned correctly with regular tests. A screenshot example of some of the test functions:

```
@BeforeAll
public static void createBooking() {
    System.out.println("Creating Booking");
    booking = new Booking(1L, 1L, "31-08-22", 1L);
}

@Test
public void testEquals() {
    EqualsVerifier.simple().forClass(Booking.class)
        .withPrefabValues(Booking.class, new Booking(1L, "31-08-22", 1L), new Booking(2L, "30-08-22", 2L))
        .verify();
}

@Test
public void testToString() {
    String expecting = "Appointment [Id=" + booking.getId() + ", viewingID=" + booking.getViewingID()
        + ", bookingDate=" + booking.getBookingDate() + ", customerID=" + booking.getCustomerID() + "];";
    assertEquals(expecting, booking.toString());
}
```

## Day 3

This was mainly spent completing creating tests and then moving onto collaboration on back-end controller testing, an example being:

```
@Autowired
private CommentsController controller;

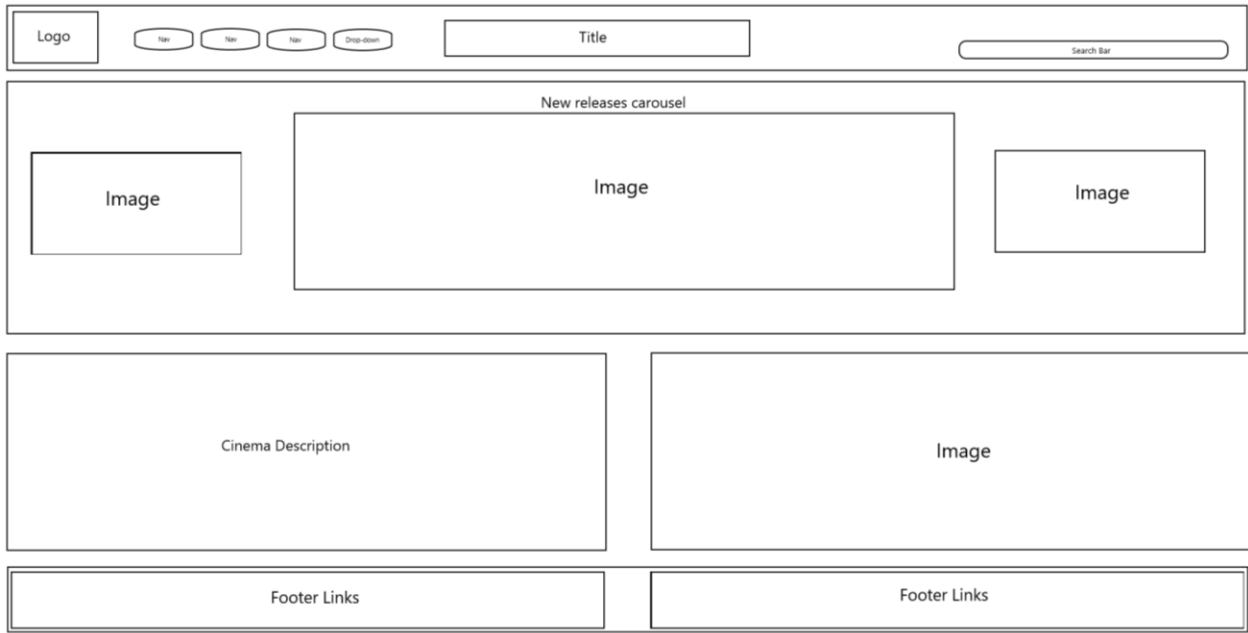
@MockBean
private CommentsService service;

@MockBean
private CommentsRepo repo;

@Test
public void createNewCommentTest() {
    Comments validComment = new Comments(1L, 1L, "content", "time", "author");
    Mockito.when(this.service.addComment(validComment)).thenReturn(validComment);
    ResponseEntity<Comments> response = new ResponseEntity<Comments>(validComment, HttpStatus.CREATED);
    assertThat(response).isEqualTo(controller.newComment(validComment));
    Mockito.verify(this.service, Mockito.times(1)).addComment(validComment);
}
```

I then worked with Sha in designing the initial

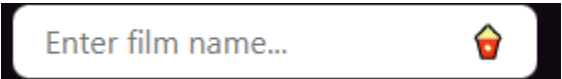
design of the front-end; the look, the HTML, the css and the JS. We then moved onto designing some wire frames for the front-end in which I was the main designer. It involved creating a user-friendly and attractive design with implementation and functionality in mind for each and every page. An example of one of the wire frames:



focusing on which page, I was assigned with implementing the search functionality for the front-end.

**Day 4-7**

The reason for grouping these days is due to the fact that I spent all of these days working on the search functionality. I began by designing the search bar itself, in which I created a very professional looking one with a button which was of popcorn to fit in with the Cinema theme as shown below:



Initially the idea was to implement a search system into the nav-bar, but this proved to be too difficult for me and even with others in my team when asking for any advice they could come up with. So instead of wasting time bashing my head, I confided with my team and decided to settle on a dedicated page specifically for searching. I started off by writing the HTML code that would fit the search bar into my page along with the other features that the page would have. An example shown below:

```
</header>
<main>
  <div class="searchBox">
    <input type="text" id="filmSearch" placeholder="Enter film name...">
    <button id="search">🔍</button>
  </div>
  <div class="films">
  </div>
  <script src="/js/scriptforsearch.js"></script>
</body>
</main>
```

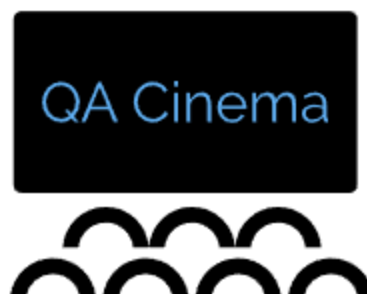
I then wrote the script that would first implement the search functionality by the ID parameter to check functionality of not only the search bar, but of the database and other connections. This worked flawlessly and showed up all the desired information and images. The script I wrote looked like this:

```
document.querySelector('#search').addEventListener("click", getFilmByTitle);

function getFilmByTitle(e){
  const id = document.querySelector("#filmSearch").value;
  fetch(`http://localhost:8090/movie/${title}`)
  .then(res => res.json())
  .then((data) => {
    document.querySelector(".films").innerHTML = `
<div class="filmInfo">
  <h1>${data.title}</h1>
  <p>Runtime: ${data.runtime}</p>
  <p>Cast: ${data.cast}</p>
  <p>Release Date: ${data.releaseDate}</p>
  <p>Age Rating : ${data.ageRating}</p>
  <p>Description: ${data.description}</p>
  <p><img src=${data.posterUrl}></p>
</div>
`;
  }).catch((err) => {
    console.log('Film not found', err);
  });
  e.preventDefault();
}
```

## Day 8

After noticing my proficiency with design, my team were more than content with me nominating myself to design the main logo for “QA Cinema”. I decided to go with a clean minimalist look which



seamlessly fit in a long with the website and it's theme. Here's what it looked like:

I then moved onto making the script I had written in the previous days work with the desired parameters which would be by film title. I spent the vast majority of this day trying to get this to work with no success. I conferred with my team and they also were at a loss for advice, so I made the decision to roll-back the code to the last functional state, so that we had something functional for users to use.

## Day 9

After rolling back my code, testing everything and making sure it was fully functional, I moved onto finishing off the HTML of the main page and creating a CSS based style-sheet that would make the search page and bar fit perfectly with the rest of the website and display normally despite the resolution/window size. An example of the CSS:

```
.searchBox {
  display: flex;
  justify-content: center;
  align-items: center;
  margin-top: 3%;
}

.searchBox input {
  border-radius: 8px 0 0 8px;
  border: 0;
  outline: none;
  height: 36px;
  padding: 4px 16px;
  background-color: white;
  color: #000;
}

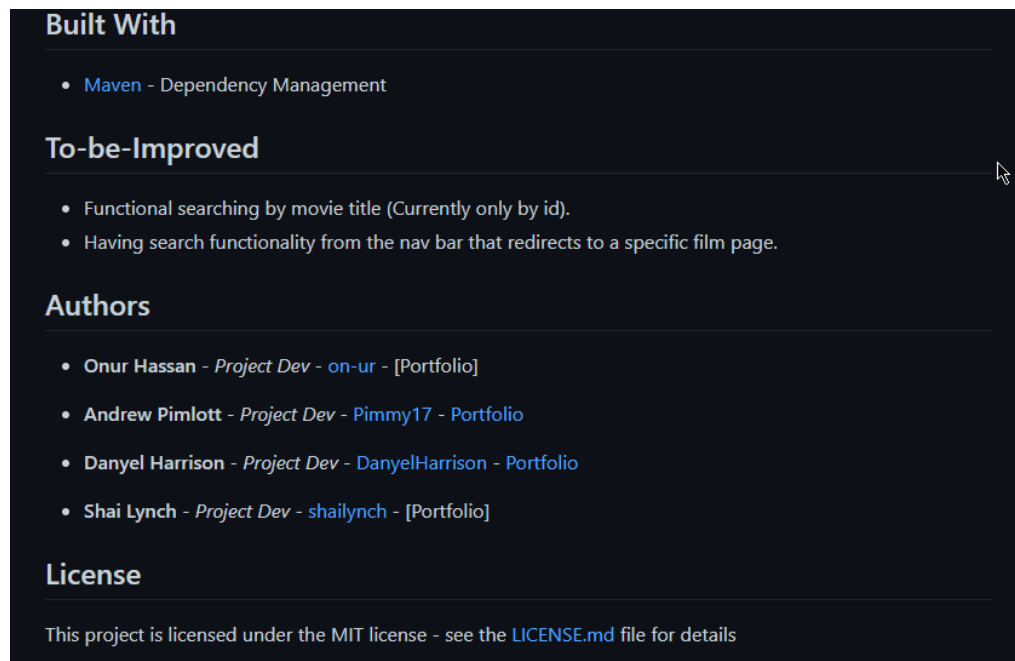
.searchBox button {
  border-radius: 0 8px 8px 0;
  border: 0;
  height: 36px;
  padding: 4px 16px;
  background-color: white;
}

.films {
  display: flex;
  margin-top: 2rem;
  justify-content: center;
}

.films p {
  margin-top: 1rem;
}

.films .filmInfo {
  padding: 2rem;
  color: rgb(216, 212, 255);
}
```

After lunch I helped team with any issues they may have had such as observing code or the Jenkins file to point out any errors/corrections that would need to be made. We then as a team we decided to move onto making a project presentation and finishing off the day with allocating roles of who was going to read and explain what from the presentation and I then added some formatting, info and embedded links to the readme to make it look professional. A snippet shown below:



## Conclusion

At the end of the project I was left feeling conflicted, on one hand I was extremely happy with the amount of progress that I have made not only during the project but also the course of the training especially considering that I had no prior experience with any development. On the other hand however, I was frustrated at myself for not being able to fully achieve the desired functionality of the search bar. On a final note, I am very proud of myself and my team (Myself, Danyel, Shai, Andrew) on how well we worked together and did with our final product which I would deem to have been a very professional outcome. I am very thankful for having the team that I had for this project and the tutors I had along the way, it will only be up from here.

