

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light green. They are positioned diagonally, with the blue one partially covering the green one.

CRUD-Based Web Application

Ryan Sharp

Introduction





Initial notes

Columns:

- Album
- Artist
- Year
- Genre
- Artwork*
- Spotify link*
- Tracklist*

GET requests:

- All
- By Id
- By Artist*
- By Year*
- By Genre*

Concept



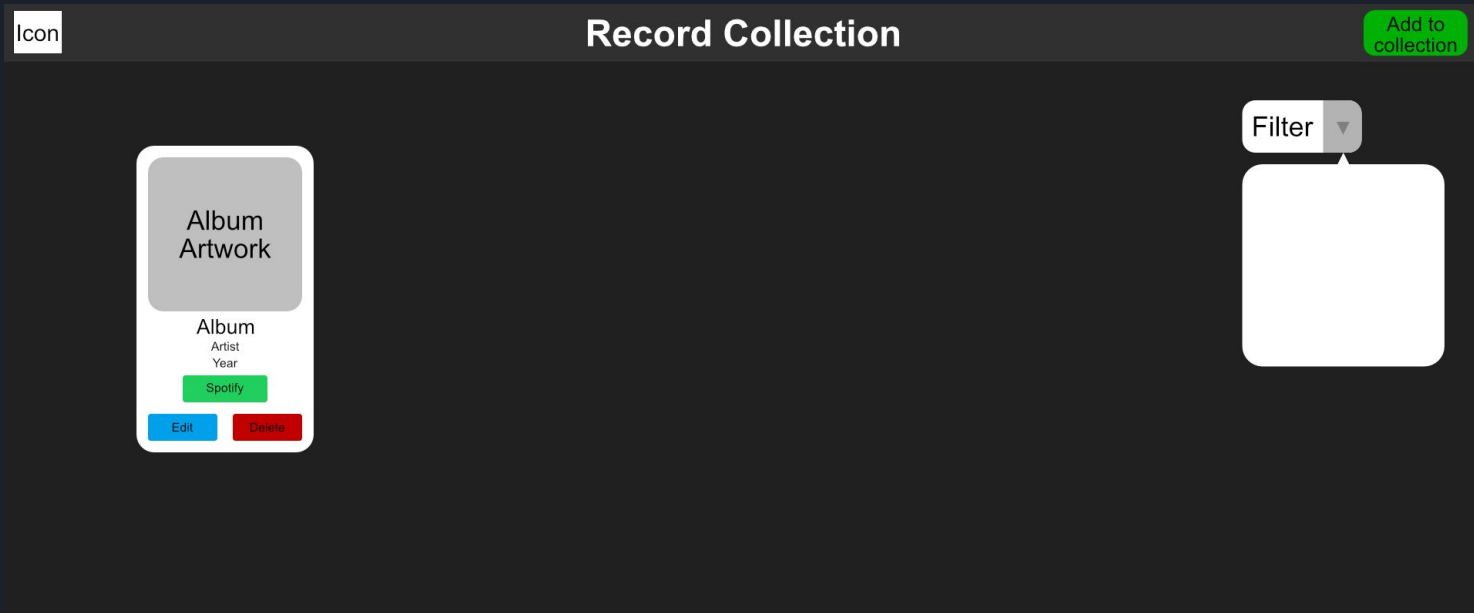
The issue with Discogs

Versions (325)

Find Your Version

Title (Format)	Label	Cat#	Country	Year	
Born To Run (LP, Album, M/Print, Pit)	Columbia	PC 33795	US	1975	▼
Born To Run = 明日なき暴走 (LP, Album, Promo)	CBS/Sony	SOPO - 96	Japan	1975	▼
Born To Run = 明日なき暴走 (Cass, Album)	CBS/Sony	25KP 869	Japan	1975	▼
Born To Run = 明日なき暴走 (LP, Album)	CBS/Sony, CBS/Sony	SOPO 96, SOPO-96	Japan	1975	▼
Born To Run (8-Trk, Album)	Columbia	CA 33795	Canada	1975	▼
Born To Run (8-Trk, Album)	CBS	42-69170	UK	1975	▼

A simpler solution



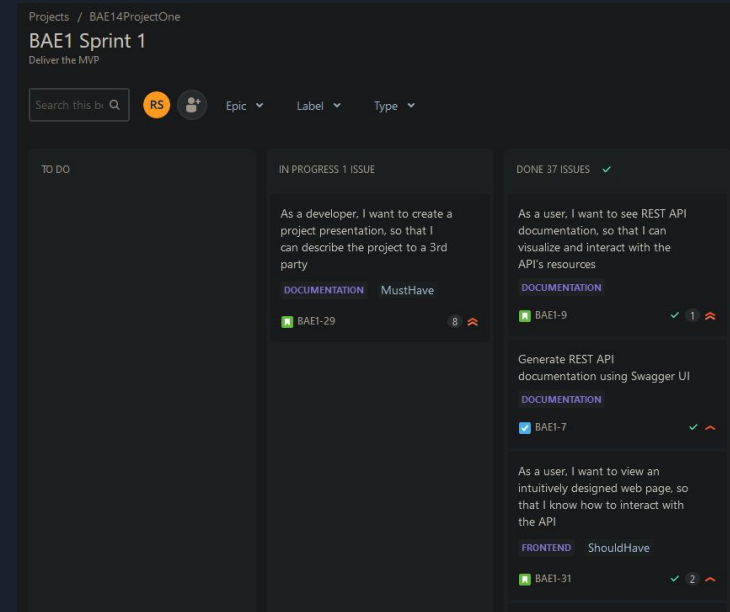
Sprint Plan



The Plan

4 Epics:

- Front end
- Back end
- Testing
- Documentation





Front End

A functional front-end website which connects to the back-end API and can be used to perform CRUD operations on the database



Back End

- A functional application written using the Java framework, Spring Boot
- A relational database, used to persist data for the project containing at least one entity



Testing

Unit and integration tests for the application with the aim of reaching the industry-standard 80% coverage



Documentation

- A risk assessment outlining the issues and risks during the timeframe
- An image of the API endpoints
- A completed README
- A copy of this presentation

Consultant Journey

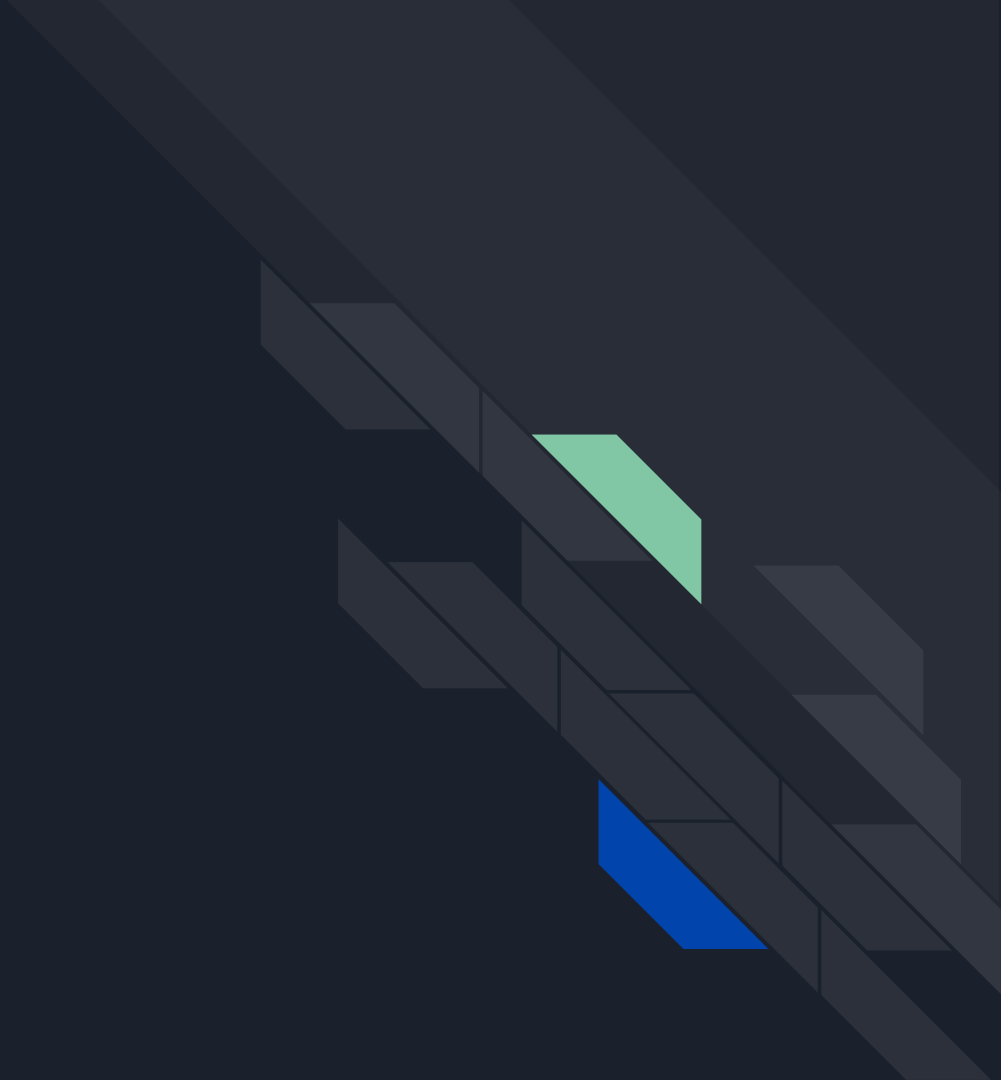




Learned Technologies

- Git
- HTML
- CSS
- JavaScript
- Java
- Spring Boot
- Maven
- JUnit

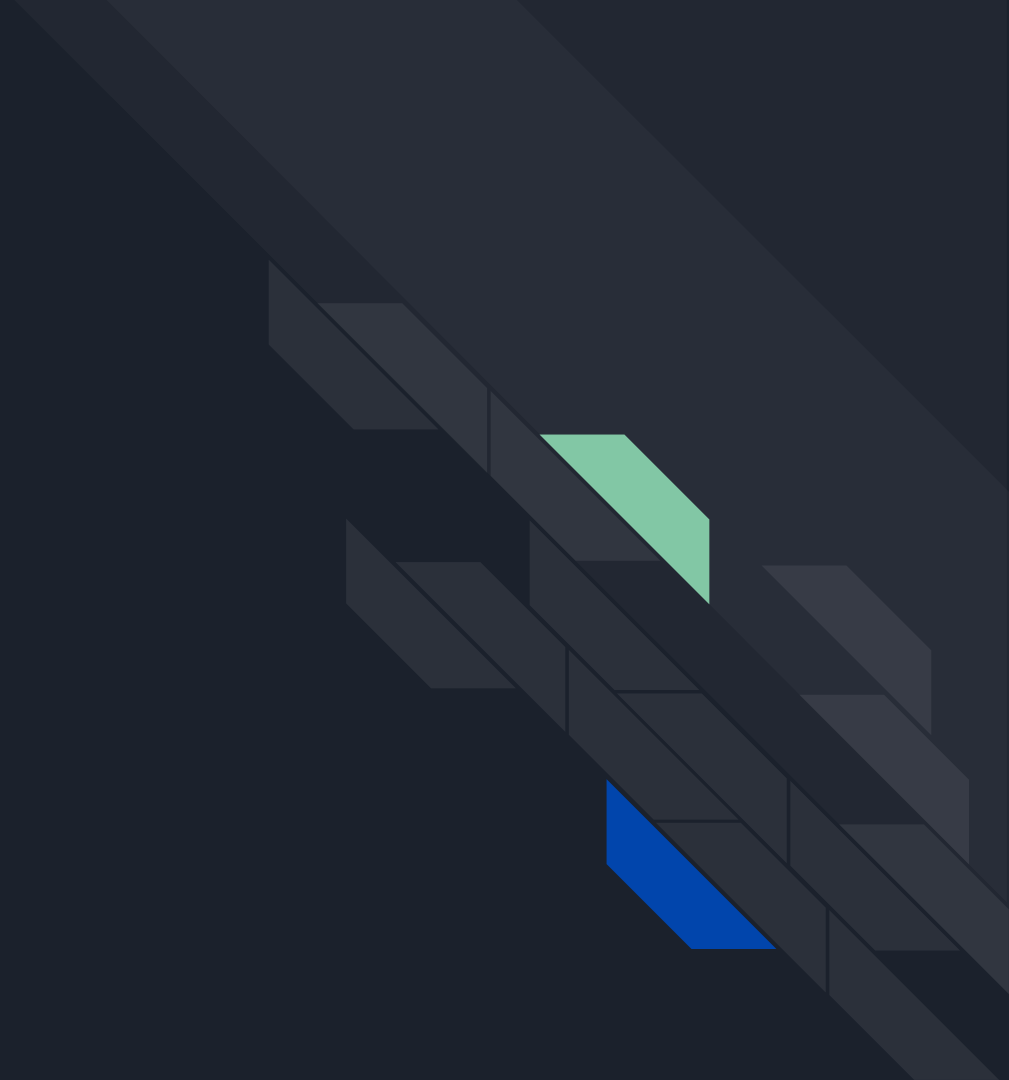
CI



Version Control



Testing





Integration Testing

```
106 // Create test
107 @Test
108 public void createTest() throws Exception {
109     Records entry = new Records("Born to Run", "Bruce Springsteen", "Rock & Roll", 1975);
110     String entryAsJSON = mapper.writeValueAsString(entry);
111
112     Records result = new Records(2L, "Born to Run", "Bruce Springsteen", "Rock & Roll", 1975);
113     String resultAsJSON = mapper.writeValueAsString(result);
114
115     mvc.perform(post("/records/create")
116         .contentType(MediaType.APPLICATION_JSON)
117         .content(entryAsJSON)
118         .andExpect(status().isCreated())
119         .andExpect(content().json(resultAsJSON)));
120 }
```













Unit Testing

```
116 @Test
117 public void createTest() throws Exception {
118     Records entry = new Records("Born to Run", "Bruce Springsteen", "Rock & Roll", 1975);
119     String entryAsJSON = mapper.writeValueAsString(entry);
120
121     Records result = new Records(2L, "Born to Run", "Bruce Springsteen", "Rock & Roll", 1975);
122     String resultAsJSON = mapper.writeValueAsString(result);
123
124     Mockito.when(service.create(entry)).thenReturn(result);
125
126     mvc.perform(post("/records/create")
127         .contentType(MediaType.APPLICATION_JSON)
128         .content(entryAsJSON)
129         .andExpect(status().isCreated())
130         .andExpect(content().json(resultAsJSON)));
131 }
```

77%

Test Coverage

src/main/java

Element	Missed Instructions	Cov.
 com.qa.baespring.domain		63%
 com.qa.baespring		37%
 com.qa.baespring.controller		95%
 com.qa.baespring.exceptions		0%
 com.qa.baespring.service		97%
Total	82 of 366	77%

RecordCollection (28 Apr 2022 12:56:17)

Demonstration



Sprint Review





What did I complete?

- Everything I set out to achieve with this sprint has been completed

The Second Sprint

'Could Have' endpoints exist in code

- Get by Artist Name
- Get by Release Year
- Get by Genre

Placeholders

- Album Artwork
- Spotify Button



Sprint Retrospective





What Went Well?

- Progressed at a decent pace
- Achieved everything I intended



Improvements?

- Story point estimates
 - Spent more time on unexpected areas

```
155 - // MODALS
156 -
157 - // Get the modal
158 - const createModal = document.getElementById("create-modal");
159 - const editModal = document.getElementById("edit-modal");
160 -
161 - // Get the button that opens the modal
162 - const addBtn = document.querySelector("#add-btn");
163 -
164 - // Get the <span> element that closes the modal
165 - const span = document.getElementsByClassName("close")[0];
166 -
167 - // When the user clicks on the button, open the modal
168 - addBtn.onclick = function() {
169 -   createModal.style.display = "block";
170 - }
171 -
172 - // When the user clicks on <span> (x), close the modal
173 - span.onclick = function() {
174 -   createModal.style.display = "none";
175 -   editModal.style.display = "none";
176 - }
177 -
178 - // When the user clicks anywhere outside of the modal, close it
179 - window.onclick = function(event) {
180 -   if (event.target == createModal) {
181 -     createModal.style.display = "none";
182 -     editModal.style.display = "none";
183 -   }
184 - }
```

```

68 // PUT request
69 const openEdit = (id) => {
70   // Gets the modal and opens it
71   const editModal = document.getElementById("edit-modal");
72   editModal.style.display = "block";
73
74   axios.get(`${address}/getId/${id}`)
75     .then((res) => {
76       const entry = res.data;
77       // Populates the edit form with current values
78       const edit_form = document.forms["edit-form"];
79       edit_form["album"].value = entry.albumName;
80       edit_form["artist"].value = entry.artistName;
81       edit_form["genre"].value = entry.genre;
82       edit_form["year"].value = entry.releaseYear;
83
84       // Gets the <span> that closes this modal
85       const span = document.getElementsByClassName("close")[1];
86
87       // When the user clicks on <span>, close the modal
88       span.onclick = function() {
89         editModal.style.display = "none";
90       }
91
92       // When the user clicks outside of the modal, close the modal
93       window.onclick = function(event) {
94         if (event.target == editModal) {
95           editModal.style.display = "none";
96         }
97       }
98
99       document.querySelector("#edit-form").addEventListener('submit', (e) => {
100         e.preventDefault(); // stops the form submitting in the default way
101
102         // Takes new values from the form
103         const form = e.target;
104         const data = {
105           albumName: form.album.value,
106           artistName: form.artist.value,
107           genre: form.genre.value,
108           releaseYear: form.year.value
109         }
110
111         console.log("DATA: ", data);
112
113         // PUT request
114         axios.put(`${address}/update/${id}`, data)
115           .then((res) => {
116             console.log(res);
117             getAll();
118             editModal.style.display = "none";
119           }).catch((err) => console.error(err))
120       }).catch((err) => console.error(err))
121     })
122   }
123 }









```

Conclusion



What's next?

▼ Backlog (8 issues) 0 0 0 Create sprint

<input checked="" type="checkbox"/> BAE1-46 Add album artwork column to database	=	TO DO ▼	
<input checked="" type="checkbox"/> BAE1-47 Pull through album artwork on to the placeholder ...	=	TO DO ▼	
<input checked="" type="checkbox"/> BAE1-48 Add spotify link column to database	=	TO DO ▼	
<input checked="" type="checkbox"/> BAE1-49 Pull through spotify link on to the placeholder spotify but...	=	TO DO ▼	
<input checked="" type="checkbox"/> BAE1-50 Add search bar for album / artist	=	TO DO ▼	
<input checked="" type="checkbox"/> BAE1-51 Add ability to filter results of GET request	=	TO DO ▼	
<input checked="" type="checkbox"/> BAE1-52 Add ability to sort results of GET request	=	TO DO ▼	
<input checked="" type="checkbox"/> BAE1-53 Add method to sort a filtered GET request	=	TO DO ▼	

+ Create issue

Questions

