

myAudiofiler Test Report

Version 1.0

05/01/2023



Course: Software Engineering 01:198:431

Group 12

Vasanth Banumurthy

|

Samantha Ames

|

Henry Chau

Table of Contents

1. Introduction.....	3
i. Purpose.....	3
2. Testing Environment.....	3
i. Hardware Configurations.....	3
ii. Software Configurations.....	3
3. Unit Testing.....	3
i. Test Objective.....	3
ii. Test Summary.....	4
iii. Test Cases.....	6
4. Integration Testing.....	24
i. Test Objective.....	24
ii. Test Summary.....	24
iii. Test Cases.....	26
5. System Testing.....	44
i. Test Objective.....	44
ii. Test Summary.....	44
iii. Test Cases.....	47

1. Introduction

1.1 Purpose

This myAudiofiler Test Report provides a summary of the results of tests performed as outlined within this document for the course: **SOFTWARE ENGINEERING**

01:198:431 with Instructor: Juan Zhai

2. Testing Environment

2.1 Hardware Configurations

The tests performed in this report were executed on a 2019 MacBook Pro with macOS Ventura 13.2.1.

2.2 Software Configurations

The software used to perform the tests includes the following: npm 6.14.13, Node.js 14.17.3, ejs 3.1.9, nodemon 2.0.22, express 4.18.2, express-ejs-layouts 2.5.1, mongoose 7.1.0, dotenv 16.0.3, and body-parser 1.20.2, and Jest.

3. Unit Testing

3.1 Test Objective

The unit testing objective is to execute assessments for each requirement as individual components as each function or procedure is designed. The requirements tested can be seen in figure 3.2.

3.2 Test Summary

Function	Description	%TCs	% TCs	TCs	Priority	Remarks
		Executed	Passed	Pending		
Add Album	Check album is added to user catalog	100%	100%	0	High	n/a
Delete Album	Check album is deleted from user catalog	100%	0%	0	High	n/a
Add Artist	Check artist is added to user list	100%	100%	0	High	n/a
Delete Artist	Check artist is deleted from user list	100%	0%	0	High	n/a
Display Catalog	Confirm display is grid-list of albums in alphabetical order, with precedence of artist then year	100%	100%	0	High	n/a
Display Artist List	Confirm display is list of artists corresponding to albums in catalog, in alphabetical order	100%	0%	0	High	n/a
Edit Album Info	Check that album info can be changed and that updates are saved	100%	0%	0	High	n/a

Edit Artist Info	Check that artist info can be changed and that updates are saved	100%	100%	0	High	n/a
View Album Info	Check that all album info is correctly displayed	100%	100%	0	Moderate	n/a
View Artist Info	Check that all artist info is correctly displayed	100%	0%	0	Moderate	n/a
Album Search	Confirm correct result for user search for album using parameters artist name, album title, and/or year.	100%	100%	0	Moderate	n/a
Artist Search	Confirm correct result for user search for artist using parameters artist name and/or genre	100%	100%	0	Moderate	n/a
View Recent Albums	Confirm user homepage display of recently added albums, in order most recent to least recent	100%	100%	0	Low	n/a

Notify Duplicates - Album	Check that system notifies user if duplicate album is attempted to be added	100%	0%	0	Low	n/a
Notify Duplicates - Artist	Check that system notifies user if duplicate artist is attempted to be added	100%	0%	0	Low	n/a

Figure 3.2 - Unit Testing Summary Table

Figure 3.2 exhibits each functional requirement tested during the unit testing phase, and its subsequent results.

3.3 Test Cases

General Information			
Test Stage:	Unit		
Test Date:	4/2/23		
Tester:	Group 12	Test Case Number:	1
Test Case Description:	Check album is added to user catalog		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Add Album (REQ-1)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop	Terminate the running program or user clicks 'Cancel' from 'Add Album' page		

Procedures:	
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Release date must be valid; required fields (Title, Artist, Release Date, Cover) must be populated
Input Specifications:	Title, Artist, Release Date, Cover
Procedural Steps:	From homepage, user must click 'Add Album' and then fill out required fields. User then must click 'Create' to add the album.
Expected Result of Case:	If required fields have been populated and are valid, user will be redirected to 'All Albums' page where the album cover will be listed in their catalog. If required fields are empty or invalid, user will receive error message.
Actual Results	
Output Specifications:	Test passed and the expected result matches the actual result.

General Information			
Test Stage:	Unit		
Test Date:	4/5/23		
Tester:	Group 12	Test Case Number:	2
Test Case Description:	Check album is deleted from user catalog		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			

Requirement(s) to be tested:	Delete Album (REQ-2)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Album must exist in the user's catalog
Input Specifications:	n/a
Procedural Steps:	User can access albums by either clicking 'Albums' link from homepage, or user can click on a recently added album from their homepage. Once the user is viewing the album, they can click the 'Delete' button to delete the album.
Expected Result of Case:	The user should be redirected to their homepage and the album should be deleted from the database/user catalog.
Actual Results	
Output Specifications:	Album does not properly remove from database, "Error Deleting Album" output

General Information			
Test Stage:	Unit		
Test Date:	4/2/23		
Tester:	Group 12	Test Case Number:	3

Test Case Description:	Check artist is added to user list
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail
Introduction	
Requirement(s) to be tested:	Add Artist (REQ-3)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program or user clicks 'Cancel'
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	All required fields must be populated (Name)
Input Specifications:	Name field must be populated
Procedural Steps:	From homepage, user clicks 'Add Artist', and then populates the information fields. User either clicks 'Create' to add new artist, or 'Cancel' to return
Expected Result of Case:	If required fields have been populated and are valid, user will be redirected to 'All Albums' page where the album cover will be listed in their catalog. If required fields are empty or invalid, user will receive error message.
Actual Results	
Output Specifications:	Test passed and the expected result matches the actual result.

General Information			
Test Stage:	Unit		
Test Date:	4/2/23		
Tester:	Group 12	Test Case Number:	4
Test Case Description:	Check artist is deleted from user list		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Delete Artist (REQ-4)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Artist must exist in user's list.		
Input Specifications:	n/a		
Procedural Steps:	From the homepage, user clicks 'Artists' link which redirects them to their artist list page. User then clicks 'Delete' button beneath the artist name that they wish to delete.		
Expected Result of Case:	User should be redirected to homepage and the artist should be removed from database/artist list.		
Actual Results			

Output Specifications:	Artist is not removed from user list or database, "Error deleting artist" output
-------------------------------	--

General Information			
Test Stage:	Unit		
Test Date:	4/2/23		
Tester:	Group 12	Test Case Number:	5
Test Case Description:	Confirm display is of albums in alphabetical order, with precedence of artist then year		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Display Catalog (REQ-5)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Displayed catalog must be accessed from 'Albums' link and not the recently added albums on homepage		
Input Specifications:	n/a		
Procedural Steps:	From homepage, user must click 'Albums' link, which redirects them to their catalog.		

Expected Result of Case:	Grid display is of albums in alphabetical order, with precedence of artist then year
Actual Results	
Output Specifications:	Albums are properly displayed in grid format, but order is from most recently added to least recently added.

General Information			
Test Stage:	Unit		
Test Date:	4/4/23		
Tester:	Group 12	Test Case Number:	6
Test Case Description:	Confirm display is list of artists corresponding to albums in catalog, in alphabetical order		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Display Artist List (REQ-6)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	n/a		

Input Specifications:	n/a
Procedural Steps:	From homepage, user clicks 'Artists'
Expected Result of Case:	Displayed will be a list of artists in alphabetical order
Actual Results	
Output Specifications:	Artists displayed in order of addition by user, from least to most recent.

General Information			
Test Stage:	Unit		
Test Date:	4/4/23		
Tester:	Group 12	Test Case Number:	7
Test Case Description:	Check that album info can be changed and that updates are saved		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Edit Album Info (REQ-7)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		

Test	
Test Items and Features:	Release date must be a valid date, artist must have been previously added. Album must exist in user's catalog.
Input Specifications:	Required input fields: artist name, release date, album cover, album title.
Procedural Steps:	From homepage, user can access catalog by clicking the 'Album' link or can access a recently added album from the homepage. Once user is redirected to album page, user clicks 'Edit' button. The user is then redirected to the edit album page where they are prompted to change the existing fields. To update the information, user clicks 'Save' button.
Expected Result of Case:	User should be redirected to homepage and any changes made to the album should be saved in the database/catalog.
Actual Results	
Output Specifications:	Album information does not update, "Error updating album" output

General Information			
Test Stage:	Unit		
Test Date:	4/5/23		
Tester:	Group 12	Test Case Number:	8
Test Case Description:	Check that artist info can be changed and that updates are saved		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Edit Artist Info (REQ-8)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop	Terminate the running program		

Procedures:	
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Artist must exist in user's artist list.
Input Specifications:	Artist name must be populated.
Procedural Steps:	From the homepage, user must click 'Artists' button to be redirected to the artist list page. From there, the user clicks 'Edit' button beneath the name of the artist that they wish to update. User is then redirected to the edit artist page and is prompted to input the name field. User clicks 'Save' to update information.
Expected Result of Case:	User will be redirected to the artist page where the new name will populate the page. The updated name is saved in the database/artist list.
Actual Results	
Output Specifications:	The actual outcome is that of the expected outcome.

General Information			
Test Stage:	Unit		
Test Date:	4/5/23		
Tester:	Group 12	Test Case Number:	9
Test Case	Check that all album info is correctly displayed		

Description:	
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail
Introduction	
Requirement(s) to be tested:	View Album Info (REQ-9)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	User must access an album to view its information
Input Specifications:	n/a
Procedural Steps:	From the homepage, user can access an album by clicking 'Albums' link which redirects them to their catalog, where they can click on an album, or directly from the homepage by clicking on a recently added album. Once redirected to the album page, the information will display.
Expected Result of Case:	Album cover, Artist Name, Release Date, and Notes (if applicable) will display
Actual Results	
Output Specifications:	All album information is correctly displayed in the album page.

General Information			
Test Stage:	Unit		
Test Date:	4/8/23		
Tester:	Group 12	Test Case Number:	10
Test Case Description:	Check that all artist info is correctly displayed		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	View Artist Info (REQ-10)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	User must access artist page to view information		
Input Specifications:	n/a		
Procedural Steps:	From homepage, user clicks on 'Artists' link. Once redirected to the artist list page, user clicks 'View' button next to their selected artist. Once redirected to the artist page, information is displayed.		
Expected Result of Case:	Artist name and all albums in the user's catalog by that artist will display.		
Actual Results			

Output Specifications:	Artist page correctly displays information and album covers.
-------------------------------	--

General Information			
Test Stage:	Unit		
Test Date:	4/8/23		
Tester:	Group 12	Test Case Number:	11
Test Case Description:	Confirm correct result for user search for album using parameters artist name, album title, and/or year.		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Album Search (REQ-11)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Album must exist in user's catalog.		
Input Specifications:	Search query must match at least part of the album title.		
Procedural Steps:	User clicks 'Albums' link and is then redirected to their catalog. At the top of the page, they can either search by populating 'Album Title' or		

	by selected a date range for the release date. The user then clicks 'Search' and is redirected to the results page.
Expected Result of Case:	If an existing album title matches any part of the search query string, or if the release date falls within the range entered, it will appear on the results page. If no album exists matching the search criteria, results page will be empty.
Actual Results	
Output Specifications:	Search functionality works for album title, but not release date.

General Information			
Test Stage:	Unit		
Test Date:	4/8/23		
Tester:	Group 12	Test Case Number:	12
Test Case Description:	Confirm correct result for user search for artist using parameters artist name and/or genre		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Artist Search (REQ-12)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		

Test	
Test Items and Features:	Artist must exist in artist list
Input Specifications:	Must search using artist name.
Procedural Steps:	From the homepage, user clicks 'Artists' link and is redirected to the artist list page. At the top of the page, user populates the 'Artist Name' field and clicks 'Search.' User is redirected to the results page.
Expected Result of Case:	If an existing artist name matches the input string, the artist name will be listed on the results page. If no artist matches search criteria, results page is empty.
Actual Results	
Output Specifications:	Artist search functionality executes correctly.

General Information			
Test Stage:	Unit		
Test Date:	4/9/23		
Tester:	Group 12	Test Case Number:	13
Test Case Description:	Confirm user homepage display of recently added albums, in order most recent to least recent		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	View Recent Albums (REQ-13)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		

Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	20 most recently added albums will be displayed
Input Specifications:	n/a
Procedural Steps:	Grid list of album covers is displayed on the user's homepage
Expected Result of Case:	The top 20 most recently added albums by the user will populate the homepage in the form of a grid of album covers.
Actual Results	
Output Specifications:	The recently added albums are correctly displayed.

General Information			
Test Stage:	Unit		
Test Date:	4/9/23		
Tester:	Group 12	Test Case Number:	14
Test Case Description:	Check that system notifies user if duplicate album is attempted to be added		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Notify Duplicates - Album (REQ-14)		

Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest
Test	
Test Items and Features:	Album attempting to be added must match the artist name, release date, and album title of an existing album in the user's catalog.
Input Specifications:	Required input: Album title, artist name, release date, album cover
Procedural Steps:	From the homepage, user clicks 'Add Album' link and is redirected to the add album page. From there, the user populates the album title, artist name, release date, album cover, and notes (if applicable). If the album title, release date and artist name match an existing album, a popup window should appear to notify the user that this is a duplicate addition.
Expected Result of Case:	If the album title, release date and artist name match an existing album, a popup window should appear to notify the user that this is a duplicate addition. This should prevent the duplicate from being added to the database/catalog.
Actual Results	
Output Specifications:	Functionality not implemented.

General Information	
Test Stage:	Unit
Test Date:	4/9/23

Tester:	Group 12	Test Case Number:	15
Test Case Description:	Check that system notifies user if duplicate artist is attempted to be added		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Notify Duplicates - Artist (REQ-15)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Artist attempting to be added must match the artist name of an existing artist		
Input Specifications:	Required input: artist name		
Procedural Steps:	From the homepage, user clicks 'Add Artist' link and is redirected to the add artist page. After populating the artist name field and clicking 'Save', the user will be notified via popup window if this artist name already exists.		
Expected Result of Case:	If artist name exists in artist list, a popup window should notify the user of this duplicate artist, preventing the duplicate from being saved in the database/catalog.		
Actual Results			
Output Specifications:	Functionality not implemented.		

4. Integration Testing

4.1 Test Objective

The integration testing objective is to confirm that the requirements are still met as each function or procedure's module is designed and integrated. The type of integration testing used for this report is Bottom-Up Integration Testing. The requirements tested can be seen in figure 4.2.

4.2 Test Summary

Function	Description	%TCs Executed	% TCs Passed	TCs Pending	Priority	Remarks
Add Album	Check album is added to user catalog	100%	100%	0	High	n/a
Delete Album	Check album is deleted from user catalog	100%	0%	0	High	n/a
Add Artist	Check artist is added to user list	100%	100%	0	High	n/a
Delete Artist	Check artist is deleted from user list	100%	0%	0	High	n/a
Display Catalog	Confirm display is grid-list of albums in alphabetical order, with	100%	100%	0	High	n/a

	precedence of artist then year					
Display Artist List	Confirm display is list of artists corresponding to albums in catalog, in alphabetical order	100%	0%	0	High	n/a
Edit Album Info	Check that album info can be changed and that updates are saved	100%	0%	0	High	n/a
Edit Artist Info	Check that artist info can be changed and that updates are saved	100%	100%	0	High	n/a
View Album Info	Check that all album info is correctly displayed	100%	100%	0	Moderate	n/a
View Artist Info	Check that all artist info is correctly displayed	100%	0%	0	Moderate	n/a
Album Search	Confirm correct result for user search for album using parameters artist name, album title, and/or year.	100%	100%	0	Moderate	n/a
Artist Search	Confirm correct result	100%	100%	0	Moderate	n/a

	for user search for artist using parameters artist name and/or genre					
View Recent Albums	Confirm user homepage display of recently added albums, in order most recent to least recent	100%	100%	0	Low	n/a
Notify Duplicates - Album	Check that system notifies user if duplicate album is attempted to be added	100%	0%	0	Low	n/a
Notify Duplicates - Artist	Check that system notifies user if duplicate artist is attempted to be added	100%	0%	0	Low	n/a

Figure 4.2 - Bottom-Up Integration Testing Summary Table

Figure 4.2 exhibits each functional requirement tested during the bottom-up integration testing phase, and its subsequent results.

4.3 Test Cases

General Information	
Test Stage:	Integration

Test Date:	4/26/23		
Tester:	Group 12	Test Case Number:	16
Test Case Description:	Check album is added to user catalog		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Add Album (REQ-1)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Release date must be valid; required fields (Title, Artist, Release Date, Cover) must be populated		
Input Specifications:	Title, Artist, Release Date, Cover		
Procedural Steps:	From homepage, user must click 'Add Album' and then fill out required fields. User then must click 'Create' to add the album.		
Expected Result of Case:	If required fields have been populated and are valid, user will be redirected to 'All Albums' page where the album cover will be listed in their catalog. If required fields are empty or invalid, user will receive error message.		
Actual Results			
Output Specifications:	Actual results match the expected results.		

General Information			
Test Stage:	Integration		
Test Date:	4/26/23		
Tester:	Group 12	Test Case Number:	17
Test Case Description:	Check album is deleted from user catalog		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Delete Album (REQ-2)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Album must exist in the user's catalog		
Input Specifications:	n/a		
Procedural Steps:	User can access albums by either clicking 'Albums' link from homepage, or user can click on a recently added album from their homepage. Once the user is viewing the album, they can click the 'Delete' button to delete the album.		

Expected Result of Case:	The user should be redirected to their homepage and the album should be deleted from the database/user catalog.
Actual Results	
Output Specifications:	Album does not properly remove from database, "Error Deleting Album" output

General Information			
Test Stage:	Integration		
Test Date:	4/26/23		
Tester:	Group 12	Test Case Number:	18
Test Case Description:	Check artist is added to user list		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Add Artist (REQ-3)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	All required fields must be populated (Name)		

Input Specifications:	Name field must be populated
Procedural Steps:	From homepage, user clicks 'Add Artist', and then populates the information fields. User either clicks 'Create' to add new artist, or 'Cancel' to return
Expected Result of Case:	If required fields have been populated and are valid, user will be redirected to 'All Albums' page where the album cover will be listed in their catalog. If required fields are empty or invalid, user will receive error message.
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	Integration		
Test Date:	4/26/23		
Tester:	Group 12	Test Case Number:	19
Test Case Description:	Check artist is deleted from user list		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Delete Artist (REQ-4)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		

Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Artist must exist in user's list.
Input Specifications:	n/a
Procedural Steps:	From the homepage, user clicks 'Artists' link which redirects them to their artist list page. User then clicks 'Delete' button beneath the artist name that they wish to delete.
Expected Result of Case:	User should be redirected to homepage and the artist should be removed from database/artist list.
Actual Results	
Output Specifications:	Artist is not removed from user list or database, "Error deleting artist" output

General Information			
Test Stage:	Integration		
Test Date:	4/26/23		
Tester:	Group 12	Test Case Number:	20
Test Case Description:	Confirm display is of albums in alphabetical order, with presence of artist then year		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Display Catalog (REQ-5)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		

Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Displayed catalog must be accessed from 'Albums' link and not the recently added albums on homepage
Input Specifications:	n/a
Procedural Steps:	From homepage, user must click 'Albums' link, which redirects them to their catalog.
Expected Result of Case:	Grid display is of albums in alphabetical order, with precedence of artist then year
Actual Results	
Output Specifications:	Grid format of album covers properly displayed, but still in most to least recent order.

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	21
Test Case Description:	Confirm display is list of artists corresponding to albums in catalog, in alphabetical order		
Results:	☑Pass ☐Fail		
Introduction			

Requirement(s) to be tested:	Display Artist List (REQ-6)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	n/a
Input Specifications:	n/a
Procedural Steps:	From homepage, user clicks 'Artists'
Expected Result of Case:	Displayed will be a list of artists in alphabetical order
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	22
Test Case	Check that album info can be changed and that updates are saved		

Description:	
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail
Introduction	
Requirement(s) to be tested:	Edit Album Info (REQ-7)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Release date must be a valid date, artist must have been previously added. Album must exist in user's catalog.
Input Specifications:	Required input fields: artist name, release date, album cover, album title.
Procedural Steps:	From homepage, user can access catalog by clicking the 'Album' link or can access a recently added album from the homepage. Once user is redirected to album page, user clicks 'Edit' button. The user is then redirected to the edit album page where they are prompted to change the existing fields. To update the information, user clicks 'Save' button.
Expected Result of Case:	User should be redirected to homepage and any changes made to the album should be saved in the database/catalog.
Actual Results	
Output Specifications:	Album information does not update, "Error updating album" output

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	23
Test Case Description:	Check that artist info can be changed and that updates are saved		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Edit Artist Info (REQ-8)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Artist must exist in user’s artist list.		
Input Specifications:	Artist name must be populated.		
Procedural Steps:	From the homepage, user must click ‘Artists’ button to be redirected to the artist list page. From there, the user clicks ‘Edit’ button beneath the name of the artist that they wish to update. User is then redirected to the edit artist page and is prompted to input the name field. User clicks ‘Save’ to update information.		
Expected Result of Case:	User will be redirected to the artist page where the new name will populate the page. The updated name is saved in the database/artist		

	list.
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	24
Test Case Description:	Check that all album info is correctly displayed		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	View Album Info (REQ-9)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	User must access an album to view its information		
Input	n/a		

Specifications:	
Procedural Steps:	From the homepage, user can access an album by clicking 'Albums' link which redirects them to their catalog, where they can click on an album, or directly from the homepage by clicking on a recently added album. Once redirected to the album page, the information will display.
Expected Result of Case:	Album cover, Artist Name, Release Date, and Notes (if applicable) will display
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	25
Test Case Description:	Check that all artist info is correctly displayed		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	View Artist Info (REQ-10)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js		

	14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, jest.
Test	
Test Items and Features:	User must access artist page to view information
Input Specifications:	n/a
Procedural Steps:	From homepage, user clicks on 'Artists' link. Once redirected to the artist list page, user clicks 'View' button next to their selected artist. Once redirected to the artist page, information is displayed.
Expected Result of Case:	Artist name and all albums in the user's catalog by that artist will display.
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	26
Test Case Description:	Confirm correct result for user search for album using parameters artist name, album title, and/or year.		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Album Search (REQ-11)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop	Terminate the running program		

Procedures:	
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Album must exist in user's catalog.
Input Specifications:	Search query must match at least part of the album title.
Procedural Steps:	User clicks 'Albums' link and is then redirected to their catalog. At the top of the page, they can either search by populating 'Album Title' or by selected a date range for the release date. The user then clicks 'Search' and is redirected to the results page.
Expected Result of Case:	If an existing album title matches any part of the search query string, or if the release date falls within the range entered, it will appear on the results page. If no album exists matching the search criteria, results page will be empty.
Actual Results	
Output Specifications:	Search functions properly for album title, but not 100% functional for release date.

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	27
Test Case Description:	Confirm correct result for user search for artist using parameters artist name and/or genre		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		

Introduction	
Requirement(s) to be tested:	Artist Search (REQ-12)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Artist must exist in artist list
Input Specifications:	Must search using artist name.
Procedural Steps:	From the homepage, user clicks 'Artists' link and is redirected to the artist list page. At the top of the page, user populates the 'Artist Name' field and clicks 'Search.' User is redirected to the results page.
Expected Result of Case:	If an existing artist name matches the input string, the artist name will be listed on the results page. If no artist matches search criteria, results page is empty.
Actual Results	
Output Specifications:	Actual results match expected results.

General Information	
Test Stage:	Integration
Test Date:	4/27/23

Tester:	Group 12	Test Case Number:	28
Test Case Description:	Confirm user homepage display of recently added albums, in order most recent to least recent		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	View Recent Albums (REQ-13)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	20 most recently added albums will be displayed		
Input Specifications:	n/a		
Procedural Steps:	Grid list of album covers is displayed on the user's homepage		
Expected Result of Case:	The top 20 most recently added albums by the user will populate the homepage in the form of a grid of album covers.		
Actual Results			
Output Specifications:	Actual results match expected results		

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	29
Test Case Description:	Check that system notifies user if duplicate album is attempted to be added		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Notify Duplicates - Album (REQ-14)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Album attempting to be added must match the artist name, release date, and album title of an existing album in the user's catalog.		
Input Specifications:	Required input: Album title, artist name, release date, album cover		
Procedural Steps:	From the homepage, user clicks 'Add Album' link and is redirected to the add album page. From there, the user populates the album title, artist name, release date, album cover, and notes (if applicable). If the album title, release date and artist name match an existing album, a popup window should appear to notify the user that this is a duplicate addition.		
Expected Result	If the album title, release date and artist name match an existing		

of Case:	album, a popup window should appear to notify the user that this is a duplicate addition. This should prevent the duplicate from being added to the database/catalog.
Actual Results	
Output Specifications:	Functionality not implemented.

General Information			
Test Stage:	Integration		
Test Date:	4/27/23		
Tester:	Group 12	Test Case Number:	30
Test Case Description:	Check that system notifies user if duplicate artist is attempted to be added		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Notify Duplicates - Artist (REQ-15)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Artist attempting to be added must match the artist name of an existing artist		

Input Specifications:	Required input: artist name
Procedural Steps:	From the homepage, user clicks 'Add Artist' link and is redirected to the add artist page. After populating the artist name field and clicking 'Save', the user will be notified via popup window if this artist name already exists.
Expected Result of Case:	If artist name exists in artist list, a popup window should notify the user of this duplicate artist, preventing the duplicate from being saved in the database/catalog.
Actual Results	
Output Specifications:	Functionality not implemented.

5. System Testing

5.1 Test Objective

The system testing objective is to confirm that all requirements are met when myAudiofiler is tested as a whole in its design and complete integration. The requirements tested can be seen in figure 5.2.

5.2 Test Summary

Function	Description	%TCs Executed	% TCs Passed	TCs Pending	Priority	Remarks
Add Album	Check album is added to user catalog	100%	100%	0	High	n/a
Delete Album	Check album is deleted	100%	0%	0	High	n/a

	from user catalog					
Add Artist	Check artist is added to user list	100%	100%	0	High	n/a
Delete Artist	Check artist is deleted from user list	100%	0%	0	High	n/a
Display Catalog	Confirm display is grid-list of albums in alphabetical order, with precedence of artist then year	100%	100%	0	High	n/a
Display Artist List	Confirm display is list of artists corresponding to albums in catalog, in alphabetical order	100%	0%	0	High	n/a
Edit Album Info	Check that album info can be changed and that updates are saved	100%	0%	0	High	n/a
Edit Artist Info	Check that artist info can be changed and that updates are saved	100%	100%	0	High	n/a
View Album Info	Check that all album info is correctly displayed	100%	100%	0	Moderate	n/a

View Artist Info	Check that all artist info is correctly displayed	100%	0%	0	Moderate	n/a
Album Search	Confirm correct result for user search for album using parameters artist name, album title, and/or year.	100%	100%	0	Moderate	n/a
Artist Search	Confirm correct result for user search for artist using parameters artist name and/or genre	100%	100%	0	Moderate	n/a
View Recent Albums	Confirm user homepage display of recently added albums, in order most recent to least recent	100%	100%	0	Low	n/a
Notify Duplicates - Album	Check that system notifies user if duplicate album is attempted to be added	100%	0%	0	Low	n/a
Notify Duplicates - Artist	Check that system notifies user if duplicate artist is	100%	0%	0	Low	n/a

	attempted to be added					
--	-----------------------	--	--	--	--	--

Figure 5.2 - System Testing Summary Table

Figure 5.2 exhibits each functional requirement tested during the system testing phase, and its subsequent results.

5.3 Test Cases

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	31
Test Case Description:	Check album is added to user catalog		
Results:	⊖Pass □Fail		
Introduction			
Requirement(s) to be tested:	Add Album (REQ-1)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			

Test Items and Features:	Release date must be valid; required fields (Title, Artist, Release Date, Cover) must be populated
Input Specifications:	Title, Artist, Release Date, Cover
Procedural Steps:	From homepage, user must click 'Add Album' and then fill out required fields. User then must click 'Create' to add the album.
Expected Result of Case:	If required fields have been populated and are valid, user will be redirected to 'All Albums' page where the album cover will be listed in their catalog. If required fields are empty or invalid, user will receive error message.
Actual Results	
Output Specifications:	Actual results match the expected results.

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	32
Test Case Description:	Check album is deleted from user catalog		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Delete Album (REQ-2)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			

Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Album must exist in the user's catalog
Input Specifications:	n/a
Procedural Steps:	User can access albums by either clicking 'Albums' link from homepage, or user can click on a recently added album from their homepage. Once the user is viewing the album, they can click the 'Delete' button to delete the album.
Expected Result of Case:	The user should be redirected to their homepage and the album should be deleted from the database/user catalog.
Actual Results	
Output Specifications:	Album does not properly remove from database, "Error Deleting Album" output

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	33
Test Case Description:	Check artist is added to user list		
Results:	☑Pass ☐Fail		
Introduction			
Requirement(s) to be tested:	Add Artist (REQ-3)		

Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	All required fields must be populated (Name)
Input Specifications:	Name field must be populated
Procedural Steps:	From homepage, user clicks 'Add Artist', and then populates the information fields. User either clicks 'Create' to add new artist, or 'Cancel' to return
Expected Result of Case:	If required fields have been populated and are valid, user will be redirected to 'All Albums' page where the album cover will be listed in their catalog. If required fields are empty or invalid, user will receive error message.
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	34
Test Case	Check artist is deleted from user list		

Description:	
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail
Introduction	
Requirement(s) to be tested:	Delete Artist (REQ-4)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Artist must exist in user's list.
Input Specifications:	n/a
Procedural Steps:	From the homepage, user clicks 'Artists' link which redirects them to their artist list page. User then clicks 'Delete' button beneath the artist name that they wish to delete.
Expected Result of Case:	User should be redirected to homepage and the artist should be removed from database/artist list.
Actual Results	
Output Specifications:	Artist is not removed from user list or database, "Error deleting artist" output

General Information

Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	35
Test Case Description:	Confirm display is of albums in alphabetical order, with presence of artist then year		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Display Catalog (REQ-5)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Displayed catalog must be accessed from 'Albums' link and not the recently added albums on homepage		
Input Specifications:	n/a		
Procedural Steps:	From homepage, user must click 'Albums' link, which redirects them to their catalog.		
Expected Result of Case:	Grid display is of albums in alphabetical order, with precedence of artist then year		
Actual Results			
Output Specifications:	Actual results match expected results.		

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	36
Test Case Description:	Confirm display is list of artists corresponding to albums in catalog, in alphabetical order		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Display Artist List (REQ-6)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	n/a		
Input Specifications:	n/a		
Procedural Steps:	From homepage, user clicks 'Artists'		
Expected Result of Case:	Displayed will be a list of artists in alphabetical order		

Actual Results	
Output Specifications:	Actual results match expected results

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	37
Test Case Description:	Check that album info can be changed and that updates are saved		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Edit Album Info (REQ-7)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Release date must be a valid date, artist must have been previously added. Album must exist in user's catalog.		
Input Specifications:	Required input fields: artist name, release date, album cover, album title.		

Procedural Steps:	From homepage, user can access catalog by clicking the 'Album' link or can access a recently added album from the homepage. Once user is redirected to album page, user clicks 'Edit' button. The user is then redirected to the edit album page where they are prompted to change the existing fields. To update the information, user clicks 'Save' button.
Expected Result of Case:	User should be redirected to homepage and any changes made to the album should be saved in the database/catalog.
Actual Results	
Output Specifications:	Album information does not update, "Error updating album" output

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	38
Test Case Description:	Check that artist info can be changed and that updates are saved		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Edit Artist Info (REQ-8)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon		

	2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Artist must exist in user's artist list.
Input Specifications:	Artist name must be populated.
Procedural Steps:	From the homepage, user must click 'Artists' button to be redirected to the artist list page. From there, the user clicks 'Edit' button beneath the name of the artist that they wish to update. User is then redirected to the edit artist page and is prompted to input the name field. User clicks 'Save' to update information.
Expected Result of Case:	User will be redirected to the artist page where the new name will populate the page. The updated name is saved in the database/artist list.
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	39
Test Case Description:	Check that all album info is correctly displayed		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	View Album Info (REQ-9)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		

Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	User must access an album to view its information
Input Specifications:	n/a
Procedural Steps:	From the homepage, user can access an album by clicking 'Albums' link which redirects them to their catalog, where they can click on an album, or directly from the homepage by clicking on a recently added album. Once redirected to the album page, the information will display.
Expected Result of Case:	Album cover, Artist Name, Release Date, and Notes (if applicable) will display
Actual Results	
Output Specifications:	Actual results match expected results.

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	40
Test Case Description:	Check that all artist info is correctly displayed		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		

Introduction	
Requirement(s) to be tested:	View Artist Info (REQ-10)
Set Up Procedures:	Must meet all hardware/software requirements, as stated below
Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	User must access artist page to view information
Input Specifications:	n/a
Procedural Steps:	From homepage, user clicks on 'Artists' link. Once redirected to the artist list page, user clicks 'View' button next to their selected artist. Once redirected to the artist page, information is displayed.
Expected Result of Case:	Artist name and all albums in the user's catalog by that artist will display.
Actual Results	
Output Specifications:	Actual results match expected results.

General Information	
Test Stage:	System
Test Date:	5/1/23

Tester:	Group 12	Test Case Number:	41
Test Case Description:	Confirm correct result for user search for album using parameters artist name, album title, and/or year.		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Album Search (REQ-11)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Album must exist in user's catalog.		
Input Specifications:	Search query must match at least part of the album title.		
Procedural Steps:	User clicks 'Albums' link and is then redirected to their catalog. At the top of the page, they can either search by populating 'Album Title' or by selected a date range for the release date. The user then clicks 'Search' and is redirected to the results page.		
Expected Result of Case:	If an existing album title matches any part of the search query string, or if the release date falls within the range entered, it will appear on the results page. If no album exists matching the search criteria, results page will be empty.		
Actual Results			
Output Specifications:	Search functions properly for album title, but not 100% functional for release date.		

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	42
Test Case Description:	Confirm correct result for user search for artist using parameters artist name and/or genre		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Artist Search (REQ-12)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	Artist must exist in artist list		
Input Specifications:	Must search using artist name.		
Procedural Steps:	From the homepage, user clicks 'Artists' link and is redirected to the artist list page. At the top of the page, user populates the 'Artist Name' field and clicks 'Search.' User is redirected to the results page.		
Expected Result	If an existing artist name matches the input string, the artist name		

of Case:	will be listed on the results page. If no artist matches search criteria, results page is empty.
Actual Results	
Output Specifications:	Actual results match expected results

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	43
Test Case Description:	Confirm user homepage display of recently added albums, in order most recent to least recent		
Results:	<input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	View Recent Albums (REQ-13)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		
Test			
Test Items and Features:	20 most recently added albums will be displayed		

Input Specifications:	n/a
Procedural Steps:	Grid list of album covers is displayed on the user's homepage
Expected Result of Case:	The top 20 most recently added albums by the user will populate the homepage in the form of a grid of album covers.
Actual Results	
Output Specifications:	Actual results match expected results

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	44
Test Case Description:	Check that system notifies user if duplicate album is attempted to be added		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Notify Duplicates - Album (REQ-14)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		
Stop Procedures:	Terminate the running program		
Environmental Needs			
Hardware	Computer that has access to valid web browser.		
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.		

Test	
Test Items and Features:	Album attempting to be added must match the artist name, release date, and album title of an existing album in the user's catalog.
Input Specifications:	Required input: Album title, artist name, release date, album cover
Procedural Steps:	From the homepage, user clicks 'Add Album' link and is redirected to the add album page. From there, the user populates the album title, artist name, release date, album cover, and notes (if applicable). If the album title, release date and artist name match an existing album, a popup window should appear to notify the user that this is a duplicate addition.
Expected Result of Case:	If the album title, release date and artist name match an existing album, a popup window should appear to notify the user that this is a duplicate addition. This should prevent the duplicate from being added to the database/catalog.
Actual Results	
Output Specifications:	Functionality not implemented.

General Information			
Test Stage:	System		
Test Date:	5/1/23		
Tester:	Group 12	Test Case Number:	45
Test Case Description:	Check that system notifies user if duplicate artist is attempted to be added		
Results:	<input type="checkbox"/> Pass <input checked="" type="checkbox"/> Fail		
Introduction			
Requirement(s) to be tested:	Notify Duplicates - Artist (REQ-15)		
Set Up Procedures:	Must meet all hardware/software requirements, as stated below		

Stop Procedures:	Terminate the running program
Environmental Needs	
Hardware	Computer that has access to valid web browser.
Software	The software used includes the following: npm 6.14.13, node.js 14.17.3, ejs 3.1.9, express 4.18.2, express-ejs-layouts 2.5.1, nodemon 2.0.22, mongoose 7.1.0, dotenv 16.0.3, body-parser 1.20.2, Jest.
Test	
Test Items and Features:	Artist attempting to be added must match the artist name of an existing artist
Input Specifications:	Required input: artist name
Procedural Steps:	From the homepage, user clicks 'Add Artist' link and is redirected to the add artist page. After populating the artist name field and clicking 'Save', the user will be notified via popup window if this artist name already exists.
Expected Result of Case:	If artist name exists in artist list, a popup window should notify the user of this duplicate artist, preventing the duplicate from being saved in the database/catalog.
Actual Results	
Output Specifications:	Functionality not implemented.