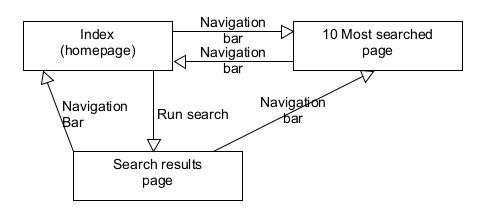
# Software Review Plan(S2)

## Current Architecture

The following navigation chart illustrates the current system structure of the project. The web application is comprised of three pages; Index, Search Results, and Top Ten. The chart demonstrates the relationships and pathways, including their direction.



The Index page is an HTML page which serves as a hub, providing text input fields and buttons. The buttons directly link to each of the two other pages, passing user input data using a “POST” method.

The Results page contains a connection script to the associated database. It generates SQL queries based on search terms posted from the index page, and constructs a table to display all matching entries in the database. A user-friendly message is displayed if no matching results are found.

The Ten Most-Searched Movies page is similar to the search results page. The page contains its own connection script and generates an SQL query to find and return the ten most-searched movies in the database. Movies are displayed in a table identical to that of the results page, and includes a bar chart displaying the titles and search count of each of the top ten movies.

## Software Quality Issues

Quality issues can be grouped in the following categories:

### Modularity

The two results pages (search results and top ten) contain **repeated code**; both of these pages have their own identical connection scripts.

### Portability

The database connection scripts contain **hard coded values** for host, username, password and database name. Given the scope of the project, this may be inconsequential but is worth noting.

### Efficiency

The code appears to contain some **redundant variables** and links, a few at most. Impact may be negligible. Further analysis required.

### Extensibility

In its current state, the codebase is quite small and simple. New additions would not be difficult to implement and integrate.

### Readability

Code is sufficiently commented in plain terms using minimal technical vocabulary. Appropriate indentation is utilized throughout. Code contains minimal nesting. Naming conventions are consistent and meaningful.

## Proposed Improvements

* Remove connection scripts from results pages. Create a dedicated database connection file to be referenced as needed, or place the connection script in a global function.
* Streamline code; check for redundant variables/links and remove if necessary
* Format code with auto-formatting tool

### Suggestions/Recommendations (Optional)

* Create optional form on homepage to override default values for database connection with user-defined values (host, user, password, etc.)
* Place code for results table in a global function

## Testing

Full program will be tested, including previously tested features to **monitor effect of changes** and ensure new additions have not broken existing modules. The testing methodology, strategy, and metrics are consistent with those detailed in the Sprint One testing plan.

### Scope

The new proposed features are detailed in the following revised testing scope:

#### In Scope

* User authentication
* Display database
* Search database
  + With any combination of parameters
* Top ten movies
* Operation on three different sized devices
  + PC
  + Tablet/Laptop
  + Phone

**New features to be tested:**

* User subscription
* Unsubscribe option
* Email user with:
  + Newsletter
  + Newsflash updates
* Admin portal
  + Receives email when user unsubscribes
  + Function to remove user details from database
* User experience (NON FUNCTIONAL)
  + Speed of operation to be reviewed in **Performance Report**

#### Out of Scope

* Login security
* User account recovery
* Thumbnails with theatrical posters/cover art
* User reviews

### Test Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Report S2 T1 | | | | | | | | |
| Project Name: RAD Sprint Two | | | Test Type: Black Box | | | | | |
| Description: Movie Search Application | | | Date: 03/06/2021 | | | | | |
| Developer(s): Blake Topham, Erol Anil, Nathan Tai | | | Tester: Blake Topham | | | | | |
| Test Case # | Test Case Name | Test Steps | | Test Data | | Expected Results | Evidence | Pass? |
| 1 | Search by Name | Enter search terms | | Search term | | Matching results displayed | Figure 15 |  |
| 2 | Search by Name, Genre | Enter search terms | | Search term | | Matching results displayed | Figure 16 |  |
| 3 | Search by Name, Genre, Rating | Enter search terms | | Search term | | Matching results displayed | Figure 17 |  |
| 4 | Search by Name, Genre, Rating, Year | Enter search terms | | Search term | | Matching results displayed | Figure 18 |  |
| 5 | Top Ten Movies | Click “10 Most Searched” button | | 10 movies with highest search count | | 10 movies with highest search count displayed | Figure 19 |  |
| 6 | Runs on: Desktop PC | Run page in desktop browser | | Page elements | | Page runs | Figure 20 |  |
| 7 | Runs on:  iPad | Run page in device simulation mode | | Page elements | | Page runs | Figure 21 |  |
| 8 | Runs on: Galaxy S5 | Run page in device simulation mode | | Page elements | | Page runs | Figure 22 |  |
| 9 | User subscribes | Enter email and click subscribe | | Email address | | User subscribed, receives newsletter/newsflash | Figure 23 |  |
| 10 | User unsubscribes | Enter email and click unsubscribe | | Email address | | Email sent to Admin account for database removal | Figure 24 |  |
| 11 | Admin removes user | Remove user entry from database | | User details(email, etc.) | | User entry removed from database | Figure 25 |  |
| Test Completeness Criteria Achieved:  1. All test cases carried out successfully  2. All bugs fixed | | | | | Suspension Criteria Met:  1. Vital functionality broken  2. Error directly prevents further testing  3. Errors too many or too often  4. Error makes further testing redundant | | | |
| Testing Complete | | | | | Testing Suspended | | | |

### Screenshots

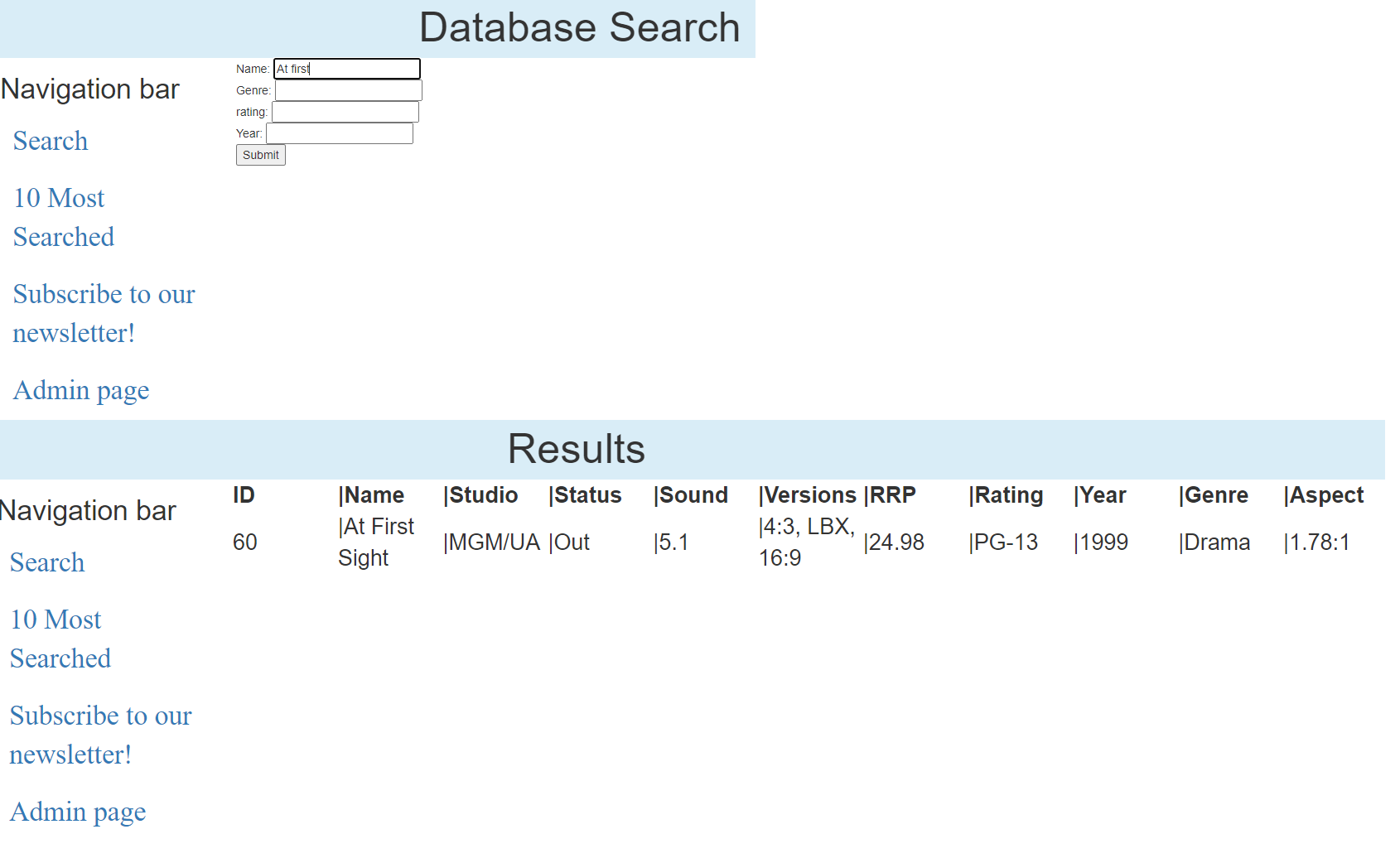


Figure 15- Search by name

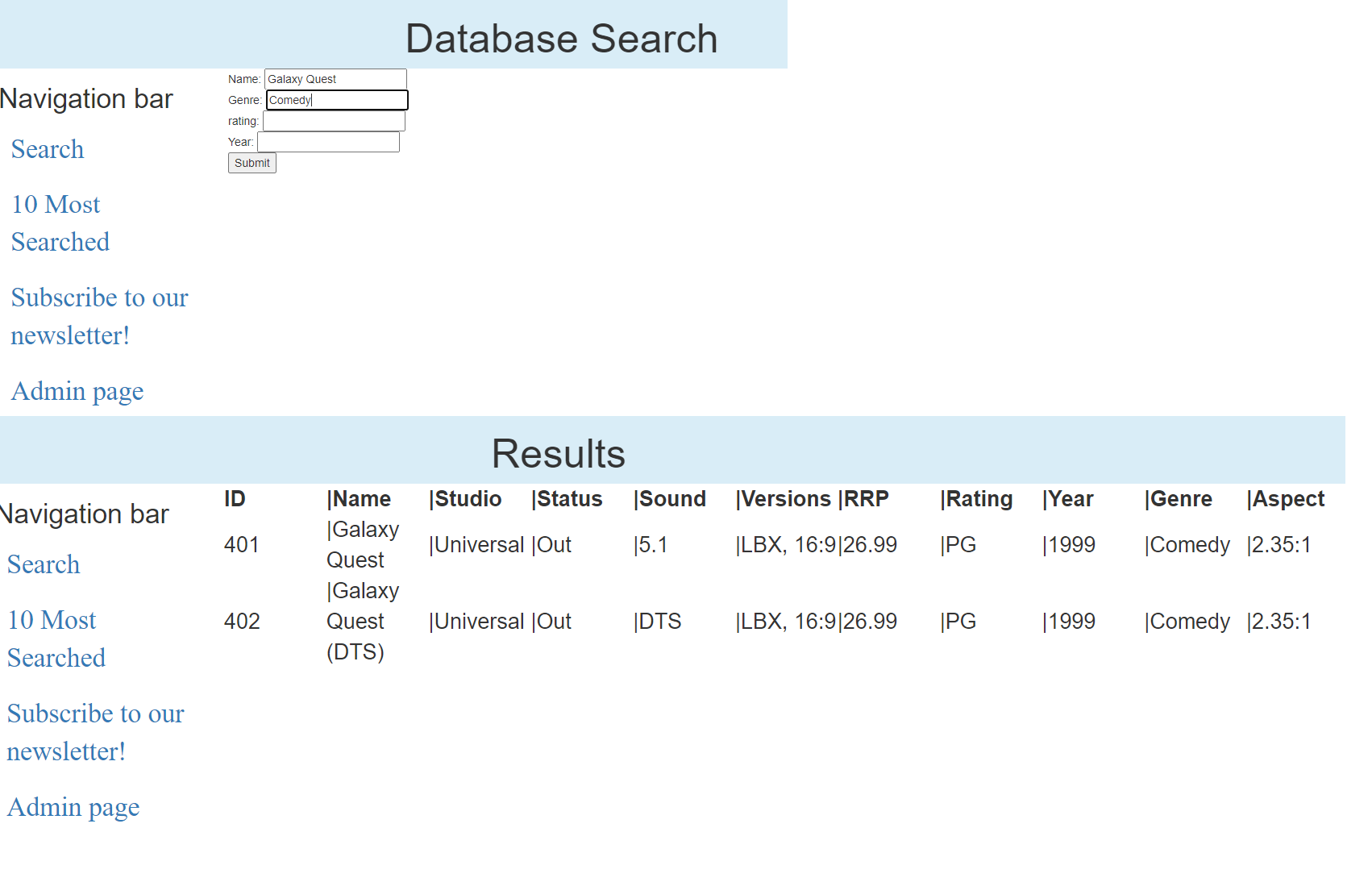


Figure 16- Name, Genre

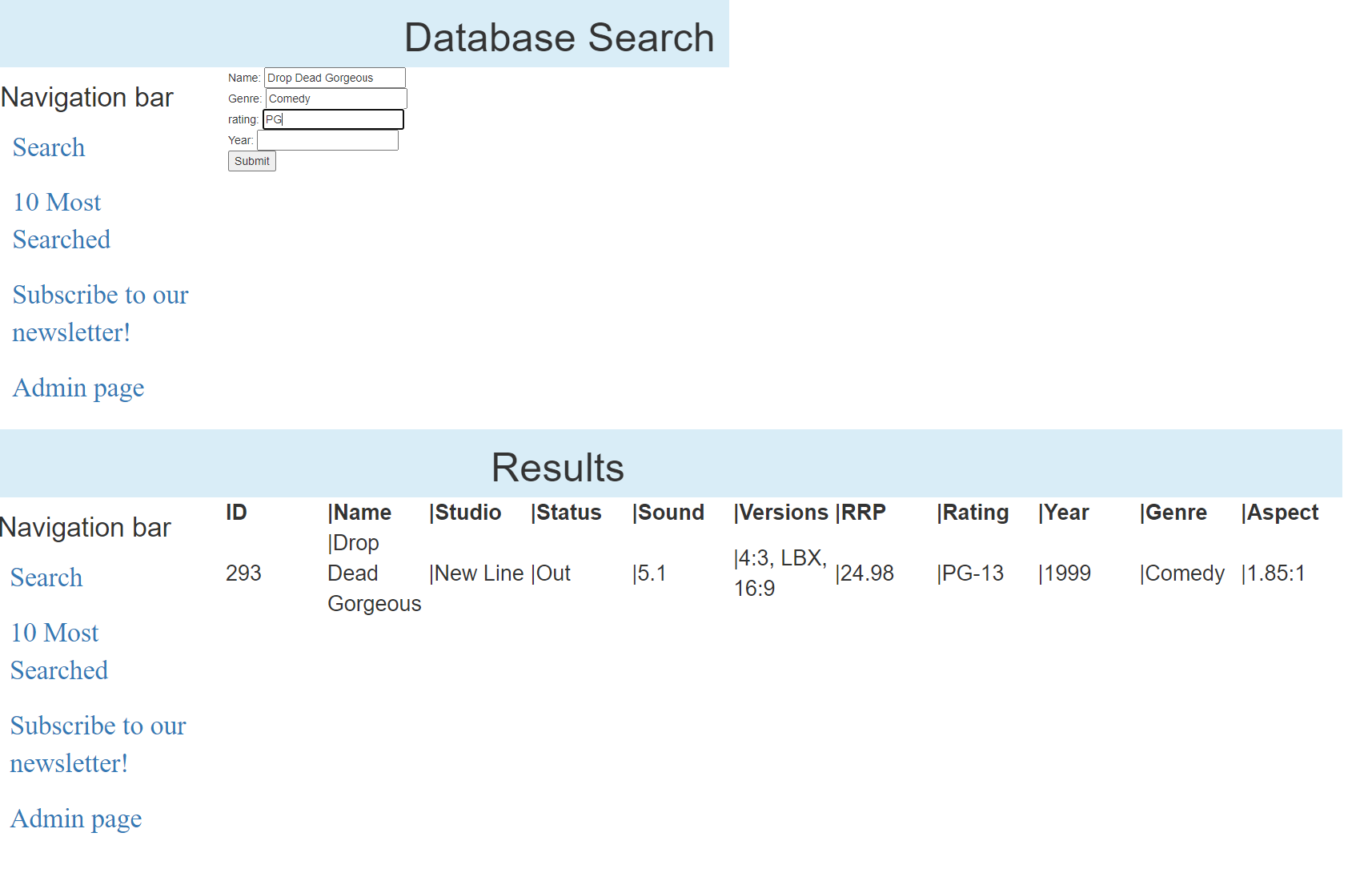


Figure 17-Name, Genre, Rating

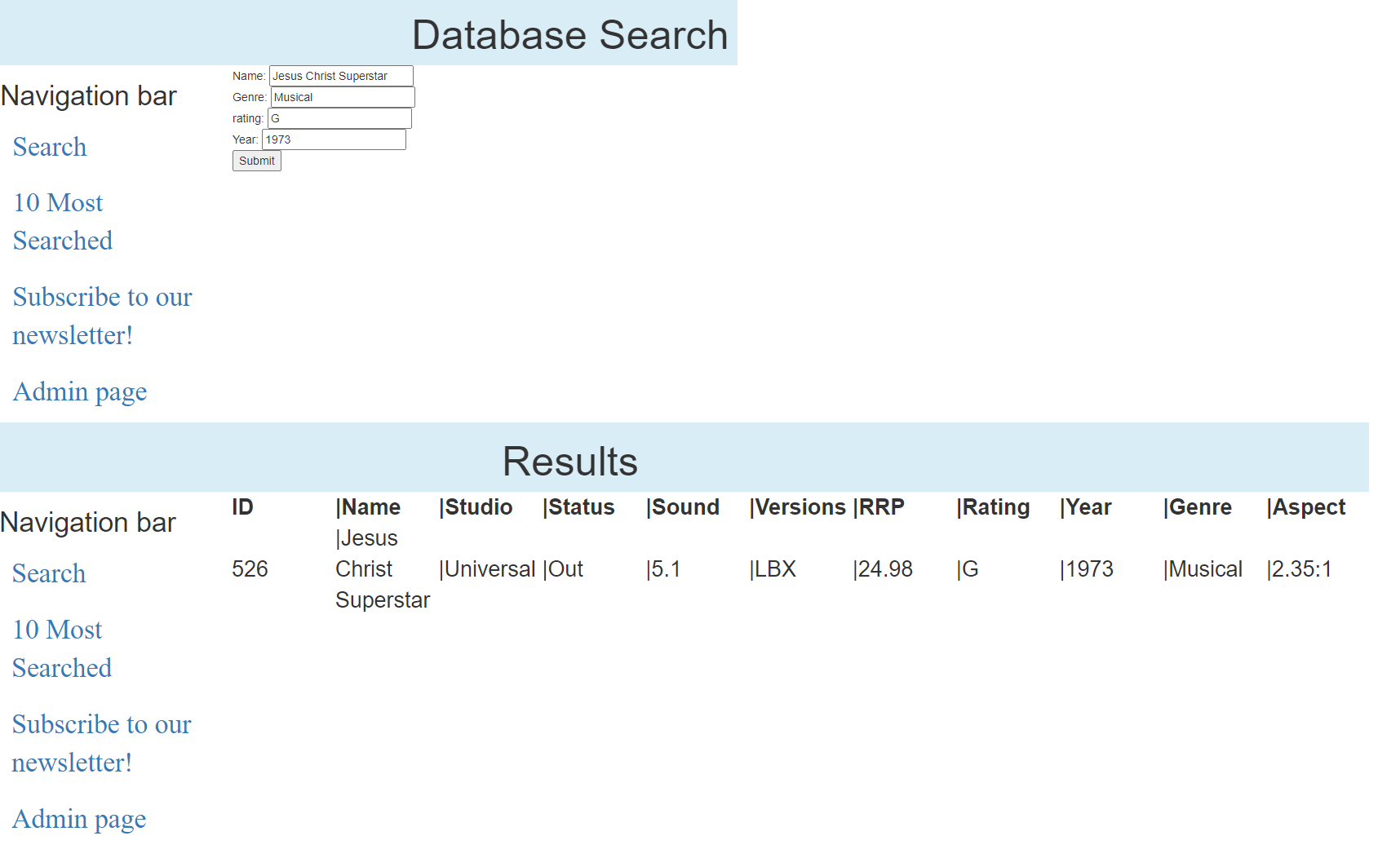


Figure 18- Name, Genre, Rating, Year

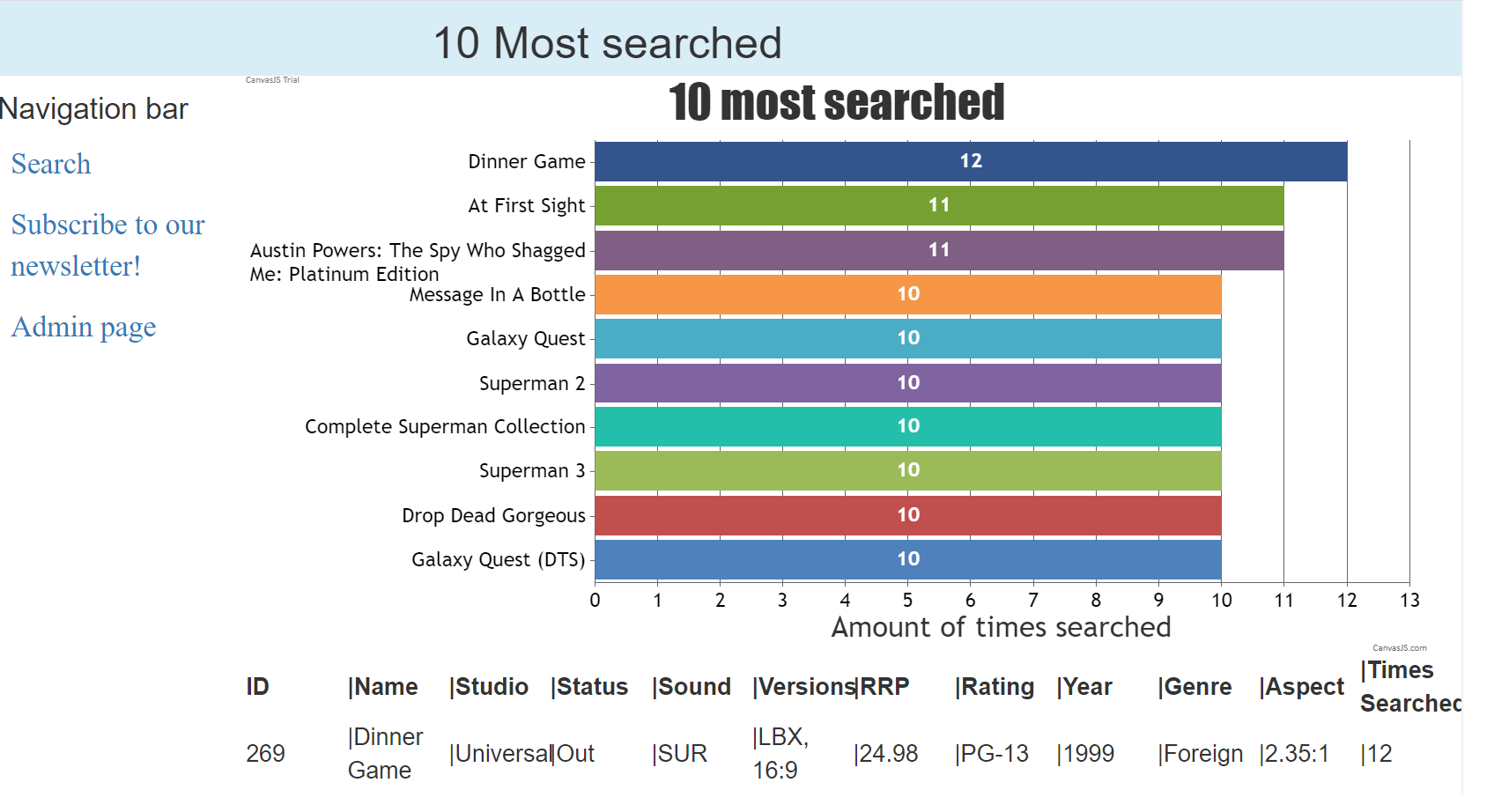


Figure 19- Top ten

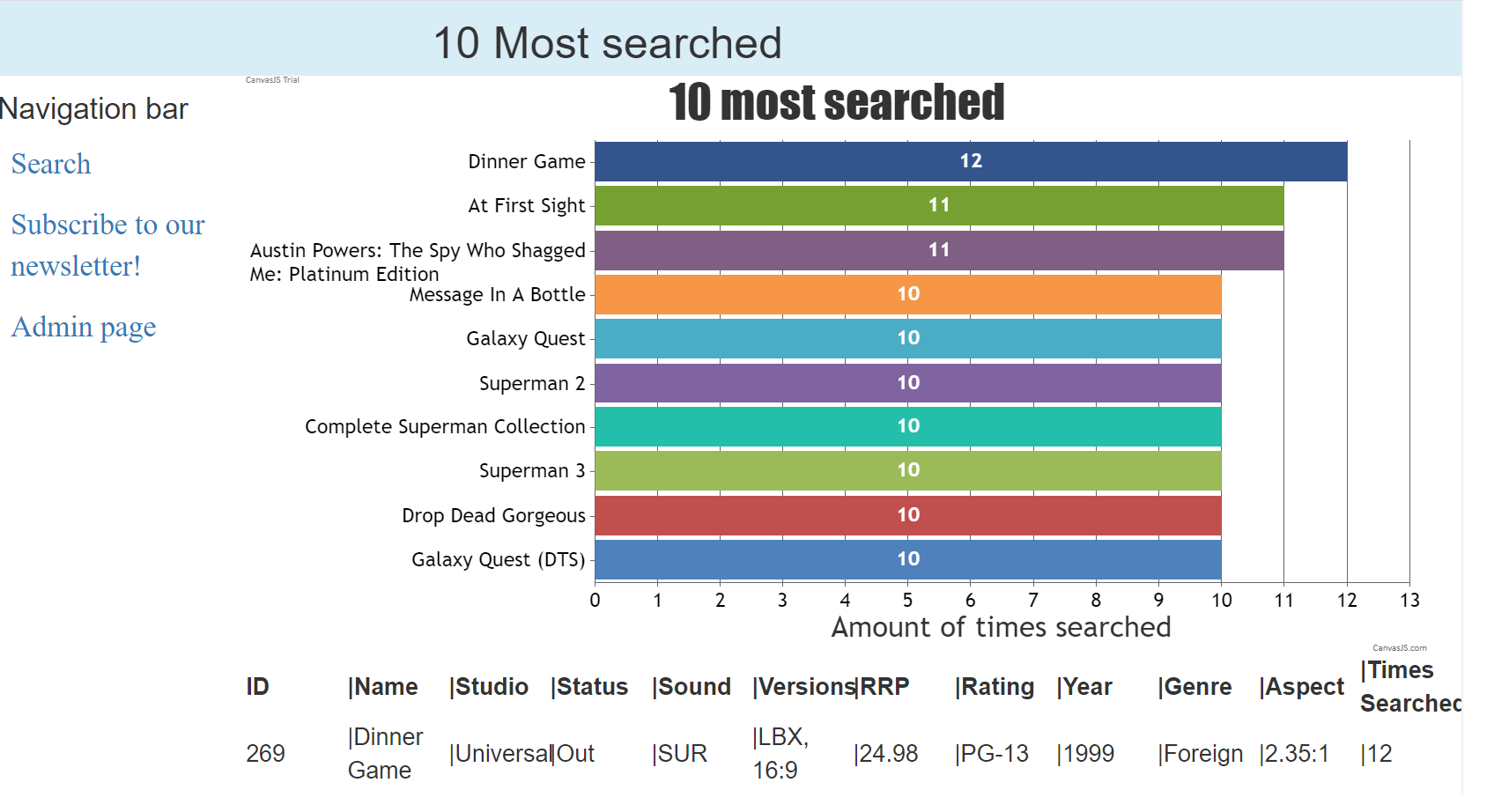


Figure 20- Runs on PC

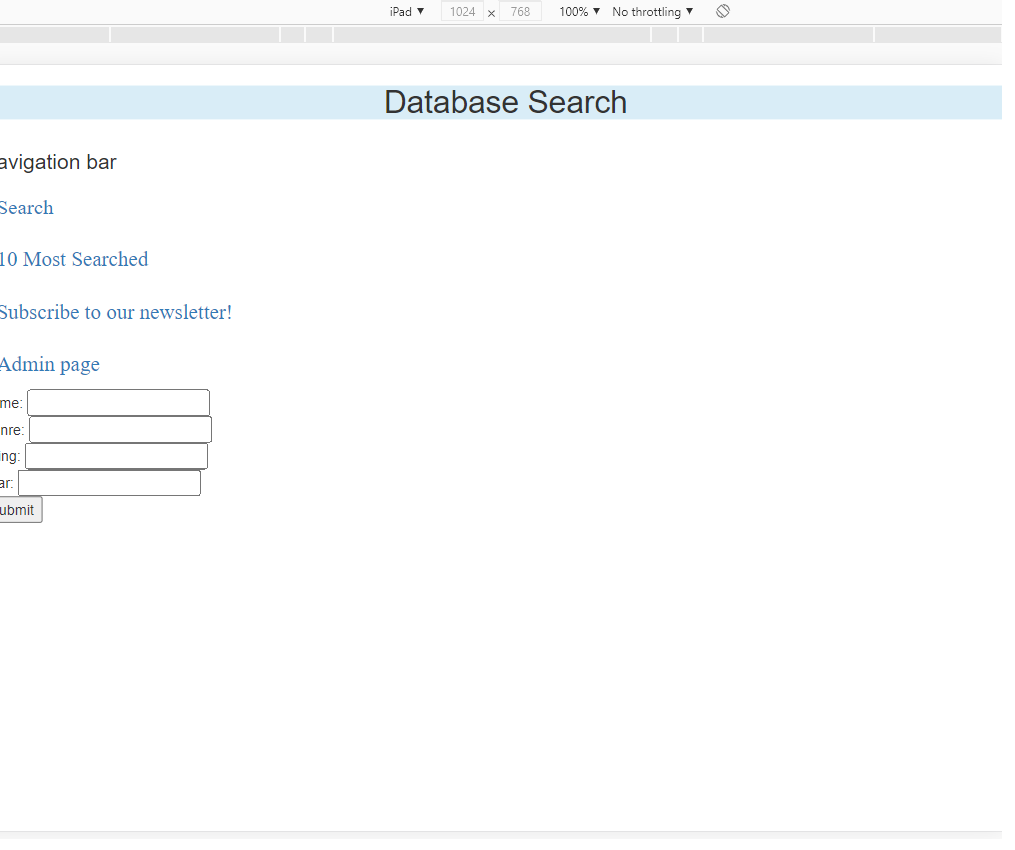


Figure 21- Runs on iPad

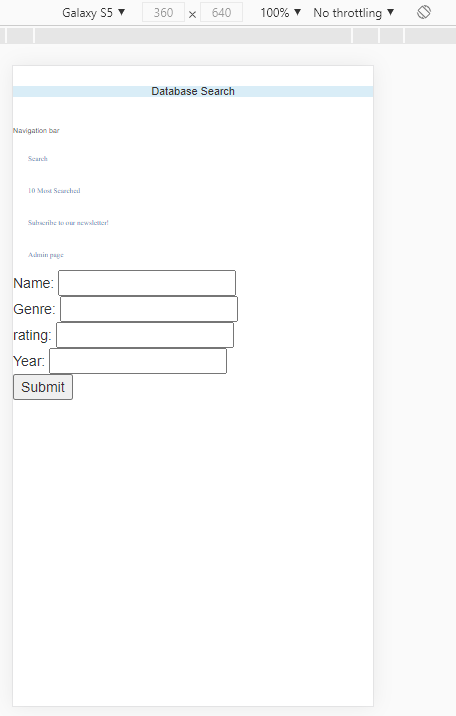


Figure 22- Runs on S5

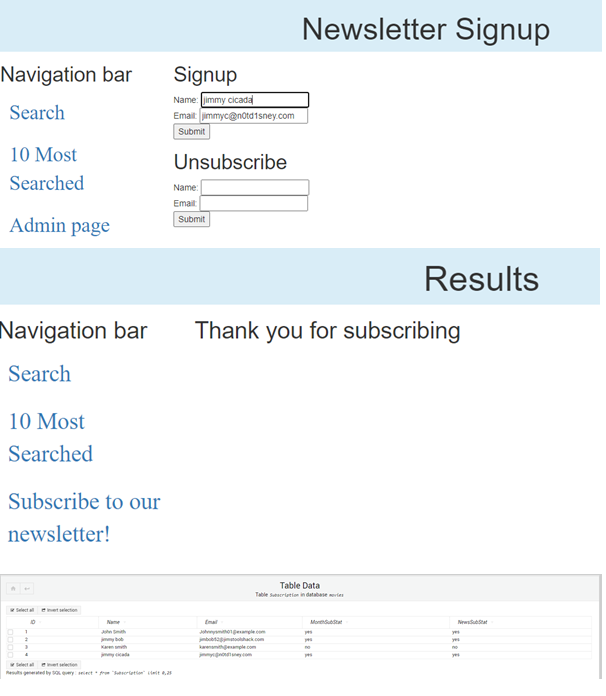


Figure 23- User unsubscribes, database not yet reflected

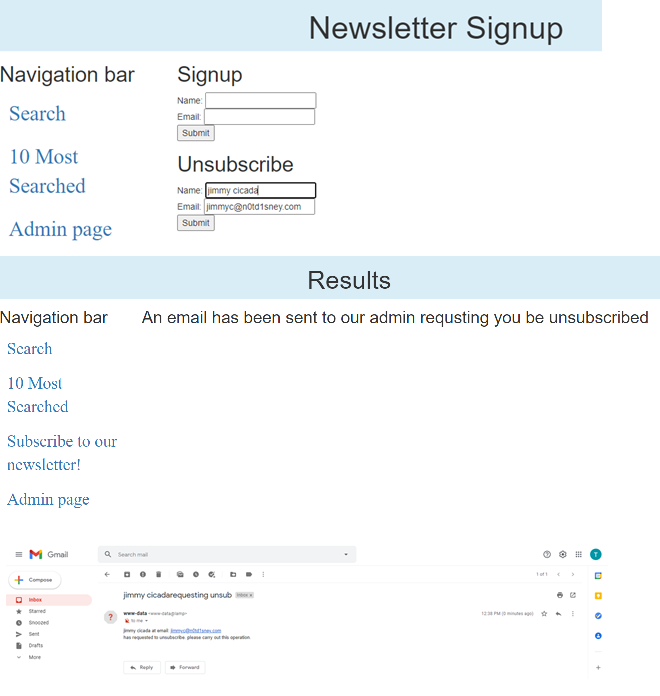


Figure 24- User unsubscribes, admin email

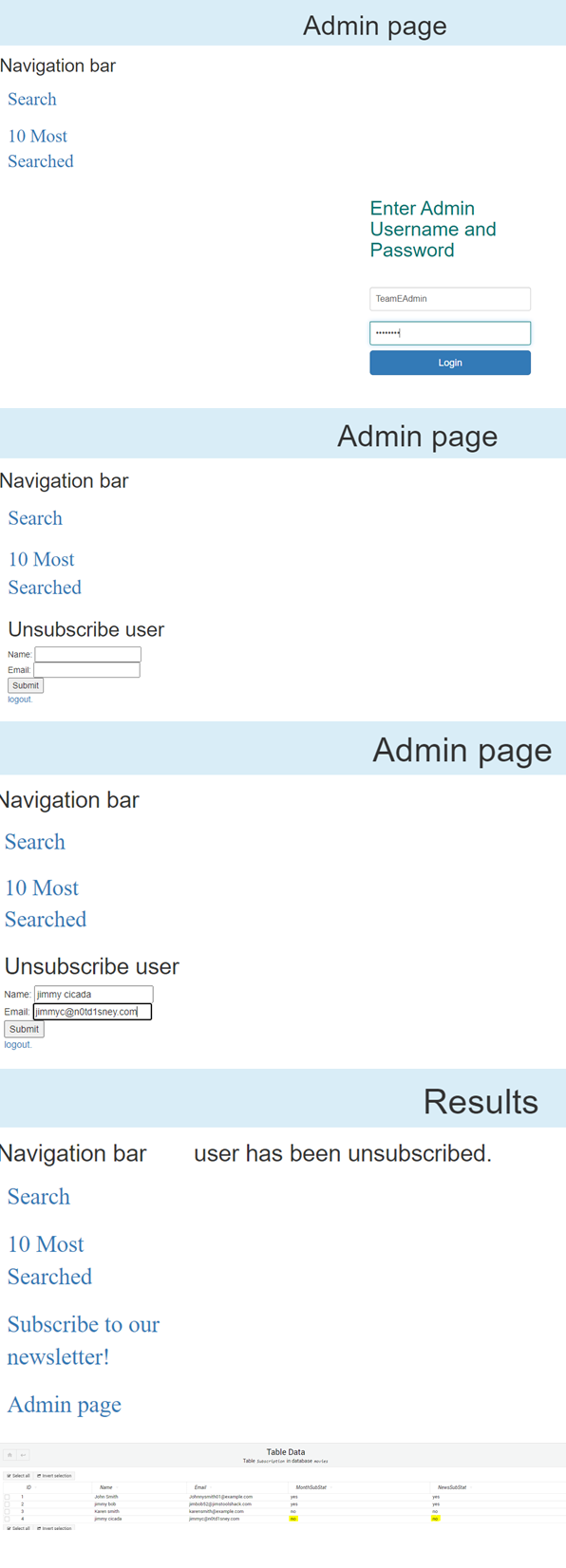


Figure 25- Admin unsubscribes user

### Bug Triage 2

#### Meeting Minutes

08/06/2021

Present: Blake Topham, Erol Anil, Nathan Tai

Next meeting: 10/06/2021, 9:00AM, Murdoch Campus

**1. Bug Report**

One issue identified and discussed. Not necessarily a bug. Unsubscribe emails going straight to spam folder.

**2. Triage (Bug Priority)**

Sole issue takes top priority by default.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#ID** | **DESCRIPTION** | **SEVERITY** | **FREQUENCY** | **RISK** | **PRIORITY** |
| 2 | Unsubscribe emails going straight to spam folder | **LOW** | **HIGH** | **LOW** | **HIGH** |

### Team Meeting 2

#### Meeting Minutes

03/06/2021

Present: Blake Topham, Erol Anil, Nathan Tai

Next meeting: 10/06/2021, 9:00AM, Murdoch Campus

**1. Team Role Assignment**

**ANIL- Configuration Manager**

Source Control Snapshot

* Update GitHub
* Create screenshots, add to master document

General Version Control

* Manage repositories

Performance Report

GANTT Chart for SPRINT TWO (week seven)

* Timeline
* Include who does what

**BLAKE- Developer**

Development

* Admin page
* Database tables for:
  + Email addresses
  + Login details if required
* Unsubscribe feature
  + Sends email to admin account
  + Do not allow user to directly manipulate database
  + Admin account required to remove database entries
* Separate options for user to select:
  + Monthly newsletter
  + Newsflash notification email

**NATHAN- Scrum Master**

Project Management Plan

* Discuss roles
* Record team meeting minutes
* Allocate tasks
* Produce documentation detailing task allocations

Software Review Plan

* Update requirements based on client meeting
* Record minutes of all meetings (client, task allocations, bug triage)
* Review application against new requirements
* Update testing plan to include new features

Demonstrate the Movie Database Application

* Read and understand all documentation
* Read and understand all code
* Demonstrate application in 3 different sized device environments

### Client Meeting 2

#### Meeting Minutes

03/06/2021

Present: Client, Blake Topham, Erol Anil, Nathan Tai

Next meeting: 10/06/2021, 9:30AM, Murdoch Campus

**1. Review Requirements**

* Project requirements discussed and documented.
* Project planning commenced.
* Additions requested.
  + Subscription feature, choice of:
    - Monthly newsletter
    - Newsflash notifications
* Requirements noted and agreed.

**2. User Experience**

* Performance review requested
  + Speed
  + Usability

Sprint Three