Birdry

Testing Report

Team 1
Kyle Fritz
Laras Istiqomah
David Leiberg
Julian Sniffen
Nicholay Topin

Client
MAPLE Lab (point of contact: John Winder)

5/3/17

Birdry
Testing Report

Table of Contents

1. Introduction

1.1 Purpose of This Document

The purpose of this document is to explain the various activities performed in testing of the Birdry application. Knowledge of the testing procedures may help future developers, maintainers, and testers in understanding the current functionality of Birdry and how these functionalities were tested.

1.2 References

Throughout this document references will be made to:

1. The Birdry System Requirements Document
2. The Birdry Code Inspection Report

2. Testing Process

2.1 Description

The team utilized the use cases outlined in the SRS document in order to confirm that all functional requirements of the software were met. In order to confirm results, multiple team members met to perform both code walkthroughs and use case tests. The author of the code was tasked with running the program and demonstrating the results to the other team members. This ensured that the code was error free and simultaneously allowed other team members to guarantee that the functional requirements had been met in their entirety.

Beyond meeting in person, the authors of each code base took personal time to review the remaining areas needed. In addition to basic use case testing, each other wrote unit tests at their own discretion.

2.2 Testing Sessions

Details of the dates and locations of specific tests are given in Table 1. Note that members other than the primary tester(s) were usually present to confirm that test results satisfied the functional requirements specified in the Birdry requirements document.

Table 1: Details of tests performed

Use Case #	Tester Name	Date	Time start	Time end	Location
1	Kyle Fritz	27 April 2017	07:00 pm	08:00 pm	Kyle/David's Apartment
2,3,4,5	Julian Sniffen and Nicholay Topin	28 April 2017	03:00 pm	03:15 pm	MAPLE Lab
6,7	Julian Sniffen	1 May 2017	08:00 pm	10:00 pm	UMBC Library
8,9,10, 11,12, 13	David Leiberg	3 May 2017	6:00 pm	8:00 pm	Kyle/David's Apartment

2.3 Impressions of the Process

Overall, the testing process was effective in finding errors as well as ensuring that all functional requirements had been met. Because we made an effort to perform testing in a group, we were able to perform higher quality testing at a faster rate. Additionally, testing in a group allowed different team members to obtain a higher understanding of the code base. We discovered several rudimentary bugs that were easily corrected, however, more importantly, we found areas of the app that were not designed in the way that the customer had originally intended. As a result, one part of the existing application had to be slightly re-engineered.

The quality of our application increased by a large amount after undergoing our testing process. Each member provided valuable input into how to increase the effectiveness of the code base. This resulted in code that was shorter, more readable, and less error-prone. Additionally, the numerous refactorings allowed the application to fully meet the functional requirements.

Our application is split into three separate code bases that all interact with each other over a network. The code base with the fewest flaws is easily the component responsible for running the neural network. This is caused by the simplicity and size of the code base relative to the other components. The code base with most flaws is the Android application. This is due to the size and complexity of the Java code base. Additionally, our team had the least amount of cumulative experience with writing Android applications. The specific module which is most vulnerable to errors is the camera portion of the Android application, as noted

in the Birdry Code Inspection Report. However, due to the thorough testing process, there is a low likelihood that there are any remaining significant errors in the code base.

3. **Test Results**

Details of the specific tests run are given in Section 3.1, a set of tables showing the testing suite. The results of these tests are specified in Section 3.2, given in the same order as the tests in Section 3.1.

3.1 Testing Suite

Use Case #1	**Take Picture**
Valid Situation	The user takes a picture while connected to the internet.
Invalid Situation	The user takes a picture while not connected to the internet.
Purpose	To test the ability to take a picture.
Expected Results for Valid Situation	The Android application saves the image to the filesystem. The image is sent to the server and returns the bird points the image has been awarded. The image is updated with the new bird points and a toast is shown to the user when the bird points are returned. The image is added to the map and gallery.
Expected Results for Invalid Situation	The Android application should capture an image and save it to the filesystem. The bird points of the image are set to -1 and a toast is shown to the user that the request could not be sent. The image is added to the map and gallery.

Use Case #2	**View Gallery**
Situation	The user swipes right to view the gallery.
Purpose	To test the ability to view the gallery.
Expected Results	The Android application smoothly slides the gallery into view. All images previously taken within the application are viewable as thumbnails in a scrollable list.

Boundary Conditions	If the user selects an image, they are redirected to the Picture Detail View for the corresponding image. If the user selects the sort button, they are prompted with a dialog for containing sorting selections

Use Case #3	**View Map**
Situation	The user swipes left to view the map.
Purpose	To test the ability to view the map.
Expected Results	The Android application smoothly slides the map into view. All images previously taken are within the application are viewable as markers on the map.
Boundary Conditions	If the user selects an image, they are redirected to the Picture Detail View for the corresponding image. If the user selects the sort button, they are prompted with a dialog for containing sorting selections

Use Case #4	**Viewing Picture Detail**
Situation	The user selects an image from the gallery or the map view.
Purpose	To test the ability to view an image in the Picture Detail View
Expected Results	The Android application displays a new view containing the image and its associated metadata. There are also buttons displayed for returning, deleting the image, and reviewing the image.

Use Case #5	**Filter Photos**
Situation	The user selects a method of sorting in the map or gallery view
Purpose	To test the ability to sort images.

Expected Results	The Android application dynamically displays the correct images in the map and gallery view depending on what the filtering option the user selects.

Use Case #6	**Delete Picture**
Situation	The user selects an image and deletes it from the Picture Detail View
Purpose	To test the ability to delete images.
Expected Results	The Android application removes the image from the Android filesystem, the gallery, and the map. The cumulative bird points score is lowered by the amount of bird points the image was awarded.
Boundary Conditions	If the image has not been sent to the server, its bird points is set to -1. In this case we do not subtract the amount of bird points from the applications cumulative score.

Use Case #7	**Ask For Review**
Valid Situation	The user selects an image and asks for review while connected to the internet.
Invalid Situation	The user selects an image and asks for review while not connected to the internet
Purpose	To test the ability to review images.
Expected Results for Valid Situation	The image's bird points is updated with the new score and the application's cumulative bird points are updated. The application sends the filename to the server which also updates the image's status.
Expected Results for Invalid Situation	The image's bird points is updated with the new status and the application's cumulative bird points are updated.

Use Case #8	**View Database**

Valid Situation	An administrator logs into the database with correct credentials.
Invalid Situation	An administrator attempts logging into the database with incorrect credentials.
Purpose	To test that an administrator can log into the database.
Expected Results for Valid Situation	The administrator logs in and views picture information on valid credential input.
Expected Result for Invalid Situation	The administrator is denied access on invalid credential input.

Use Case #9	**Admin Search Picture**
Valid Situation	An administrator searches for pictures by a user's ID.
Invalid Situation	An administrator searches for pictures by an invalid user ID.
Purpose	To test the administrator's ability to search the database.
Expected Result on Valid Input	The database returns the set of photos taken by the associated user,
Expected Result on Invalid Input	Iff user ID is invalid the database returns an empty set or table.

Use Case #10	**Admin Picture Detail**
Situation	An administrator selects an image to view its data.
Purpose	To test the administrator's ability to view the data of an image.
Expected Result	The database returns the data of the associated image.

Use Case #11	**Admin Add Picture Detail**

Situation	An administrator edits the data of an image.
Purpose	To test the administrator's ability to edit the data of an image
Expected Result	The database updates the data of an image.

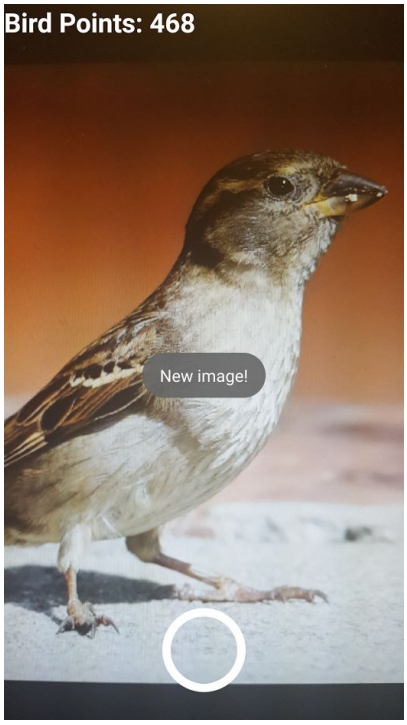
Use Case #12	**Admin Delete Picture**
Situation	The administrator deletes an image's information from the database.
Purpose	To test the administrator's ability to delete an image's information from the database.
Expected Result	The database removes the associated image's database record.

Use Case #13	**Admin Filter**
Situation	The administrator filters the images in the database.
Purpose	To test the administrator's ability to filter images in the database.
Expected Result	The database displays filtered photo information.

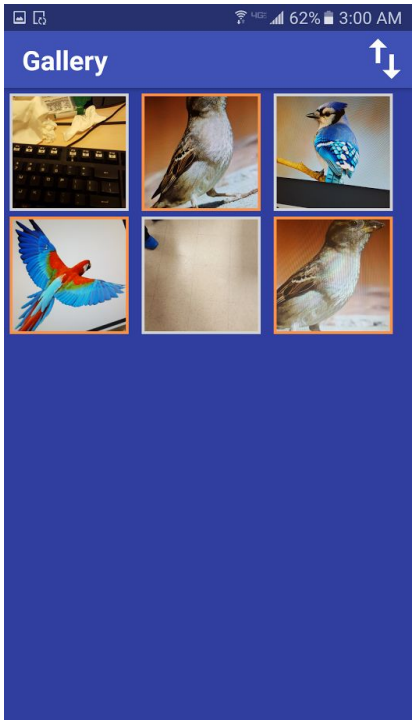
3.2 Test Results

Use Case #1 - Conducted by: Kyle Fritz

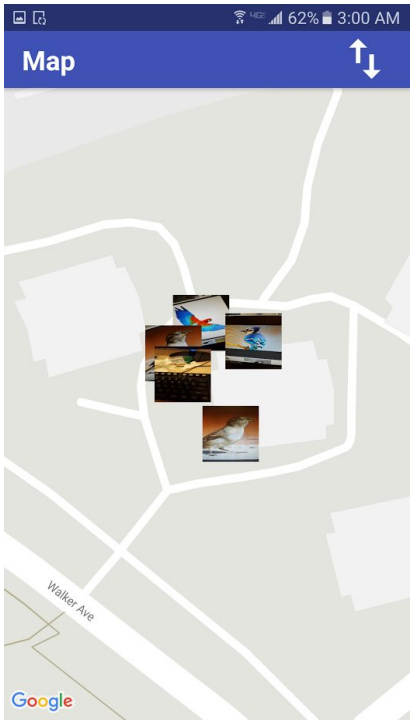
Valid Results



Toast Displayed

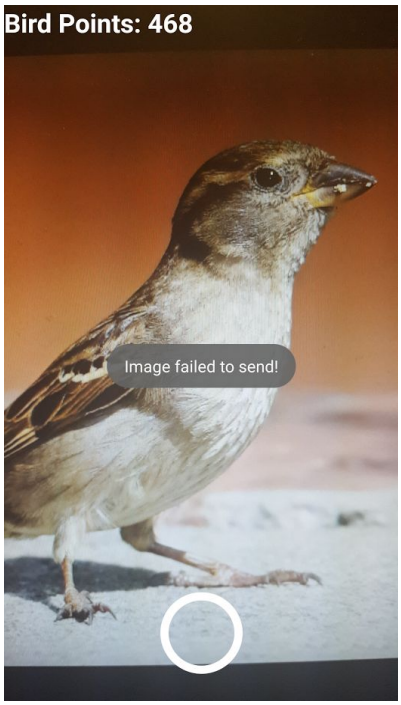


Gallery Update

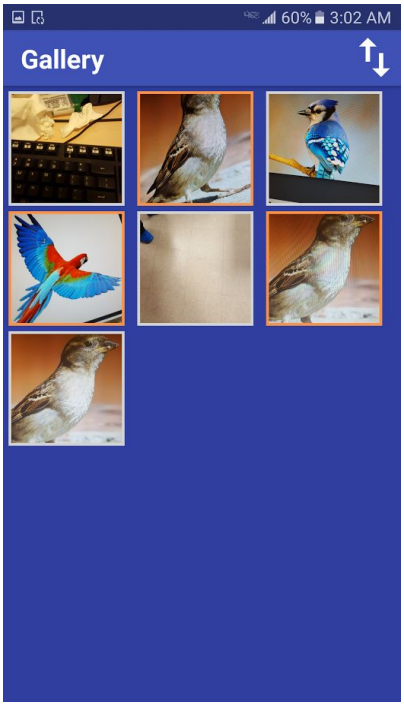


Map Update

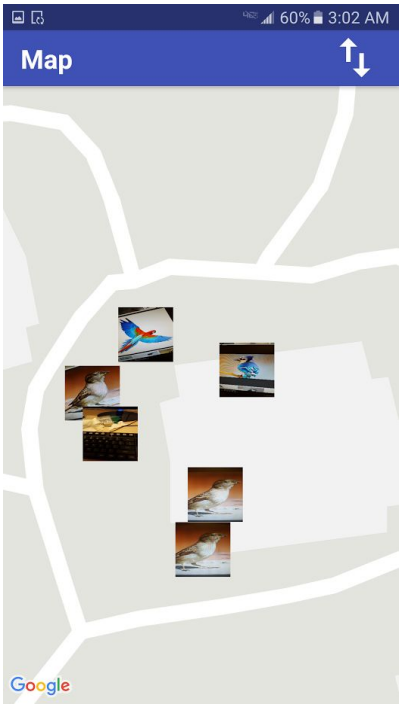
Invalid Results



Toast Displayed



Gallery Updated

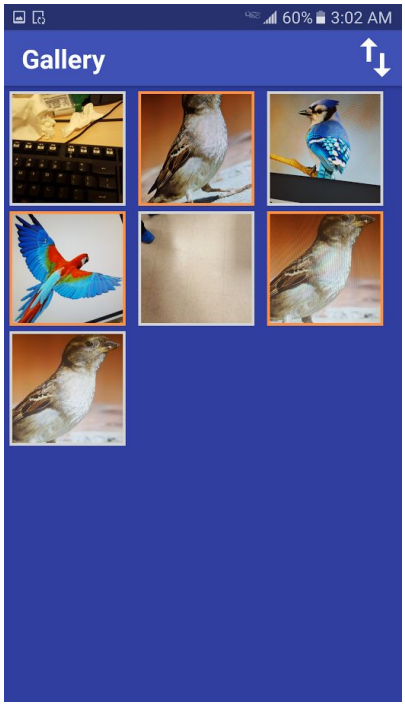


Map Updated

Test Output Summary:
Defect when picture taken, then exited out of app where accept and cancel buttons stay on the screen and let you save the old picture. Fixed by adding boolean of whether the buttons are on the screen or not. Everything else works well.

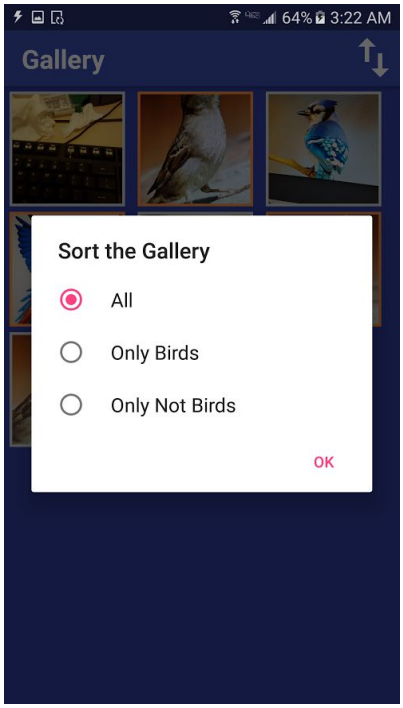
Use Case #2 - Conducted By: Julian Sniffen and Nicholay Topin

Results



The Gallery View

Boundary Condition

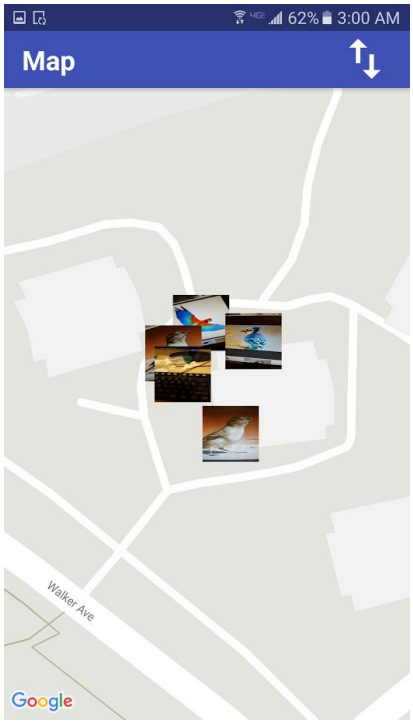


The Gallery Sort Dialog

Test Output Summary:
Everything works as expected. The transitions are smooth and the sorting selections function correctly.

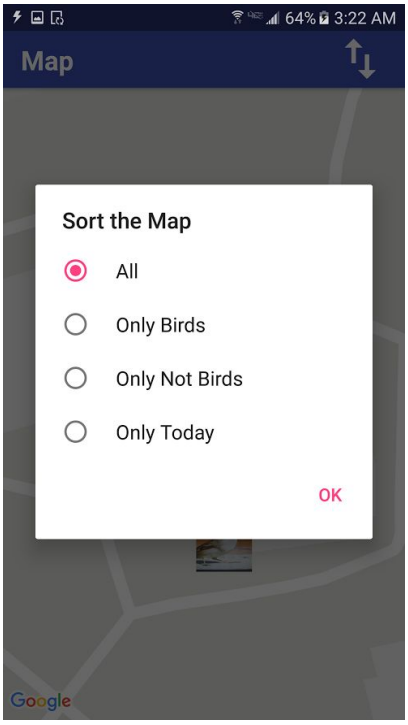
Use Case #3 - Conducted By: Julian Sniffen and Nicholay Topin

Results



The Map View

Boundary Condition



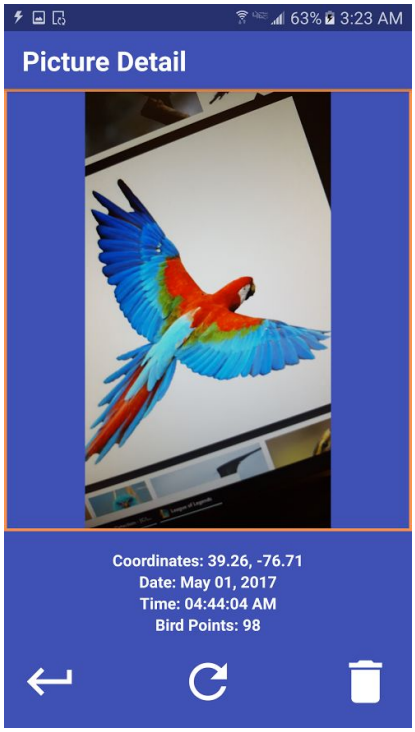
The Map Sort Dialog

Test Output Summary:

In some instances, the filename title was displaying on the map which was undesired behaviour. Additionally, some pictures were covering each other if the location where the image was taken was the same. To fix the title problem, a simple option was changed in the Google Maps API. To fix the problem of pictures covering each other, small changes were made to the location of each image if it was determined they were too close together. Other than that, animations were smooth and the sorting dialog works as expected.

Use Case #4 - Conducted By: Julian Sniffen and Nicholay Topin

Results

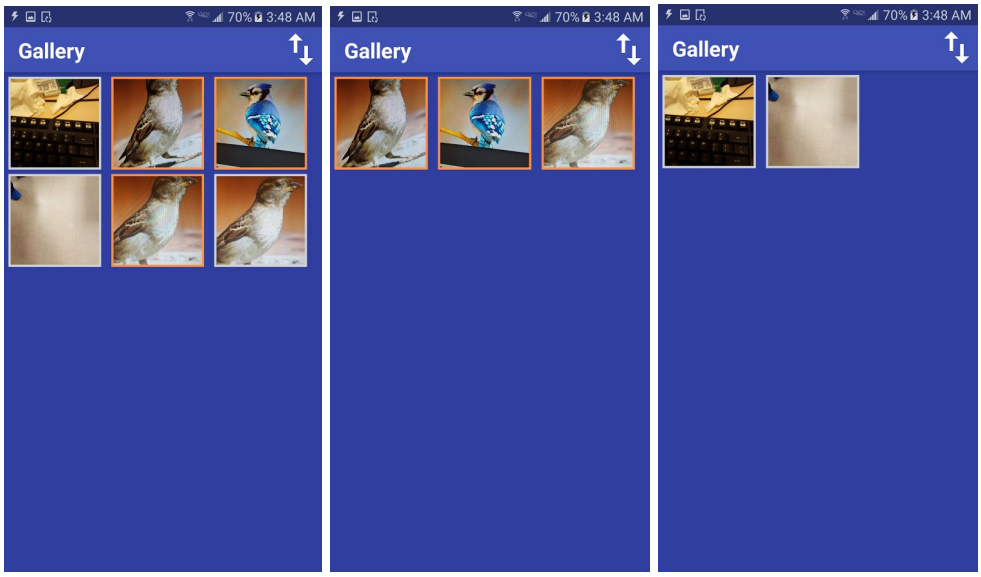


The Picture Detail View

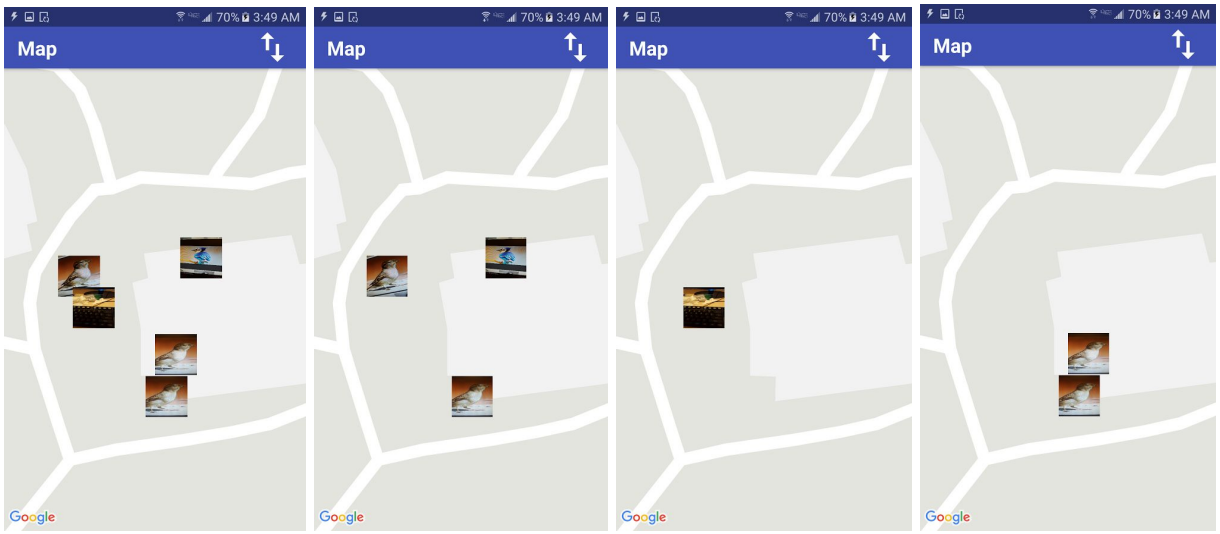
Test Output Summary:
The animations were smooth and the picture information was accurate. The buttons and information are displayed correctly.

Use Case #5 - Conducted By: Julian Sniffen and Nicholay Topin

Results



All *Only Birds* *Only Not Birds*

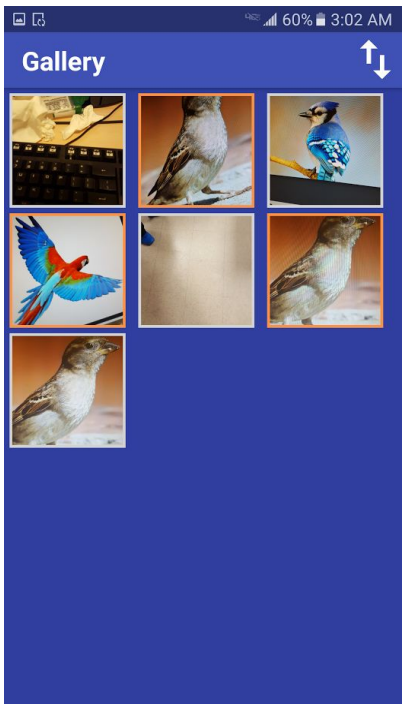


All *Only Birds* *Only Not Birds* *Only Today*

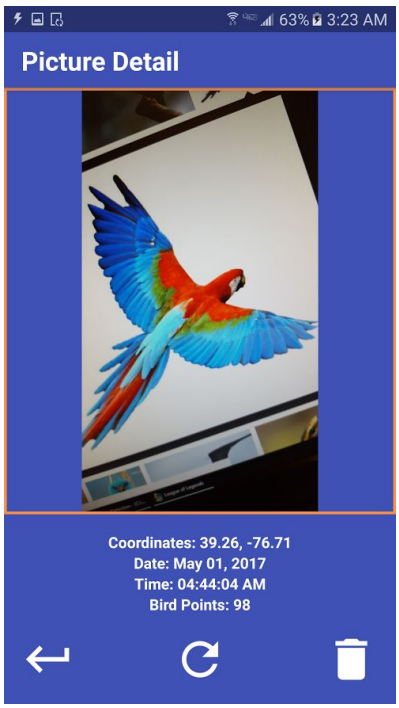
Test Output Summary:
All sorting and filtering functionality worked as expected. Animations were smooth and updates to the UI were instant

Use Case #6 - Conducted By: Julian Sniffen

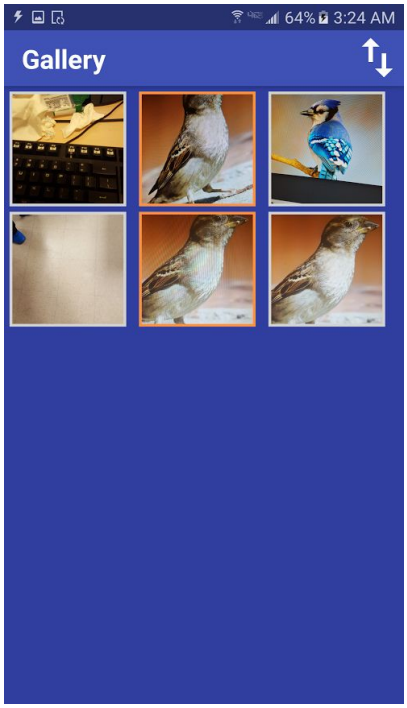
Results



Prior to deleting



Selecting Delete



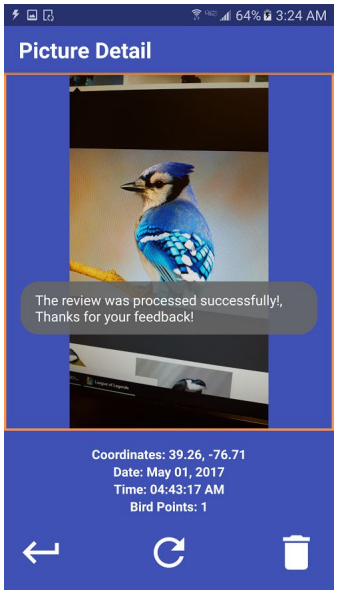
After Deleting

Boundary Condition
If the image was not sent to the server, the cumulative bird points is not affected as intended.

Test Output Summary:
All boundary cases were handled correctly. Animations are smooth, and the map and gallery update as intended.

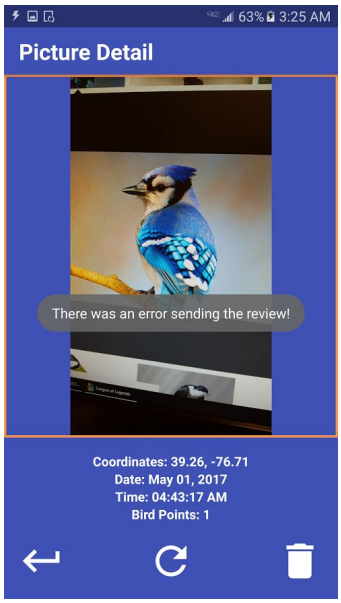
Use Case #7 - Conducted By: Julian Sniffen

Valid Results



Toast Displayed

Invalid Results



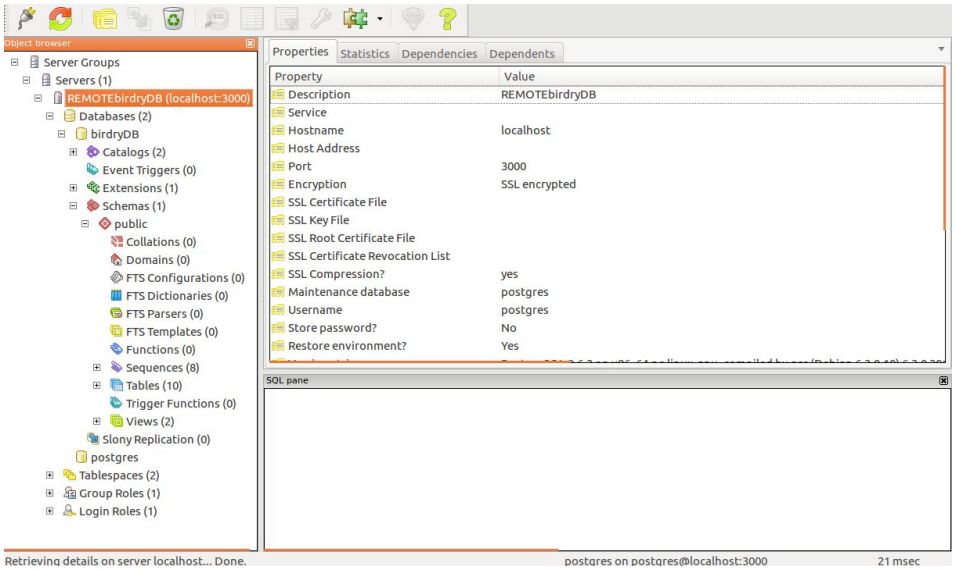
Toast Displayed

Test Output Summary:

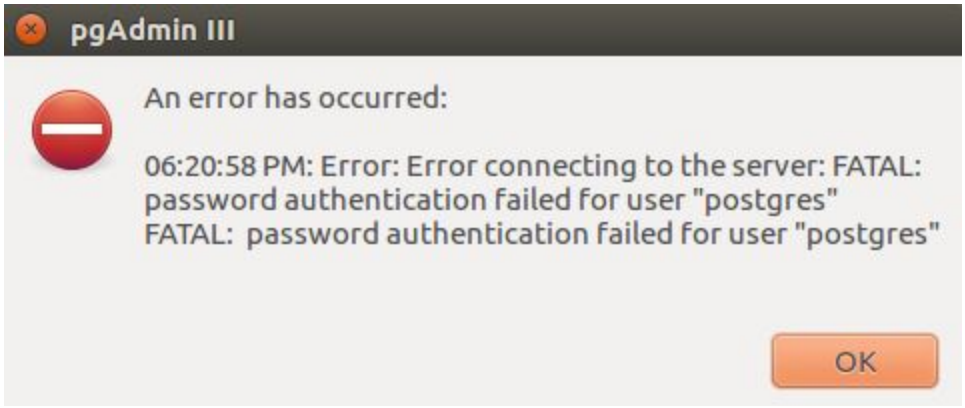
There was an error where the updated amount of bird points would incorrectly affect the cumulative amount of points for the application. This was fixed by adding conditionals in the code to handle specific edge cases. Additionally, there was an error communicating how the review should be implemented. After understanding the functional requirements, the code was slightly re-engineered. Everything else worked as expected.

Use Case #8 - Conducted By: David Leiberg

Valid Results



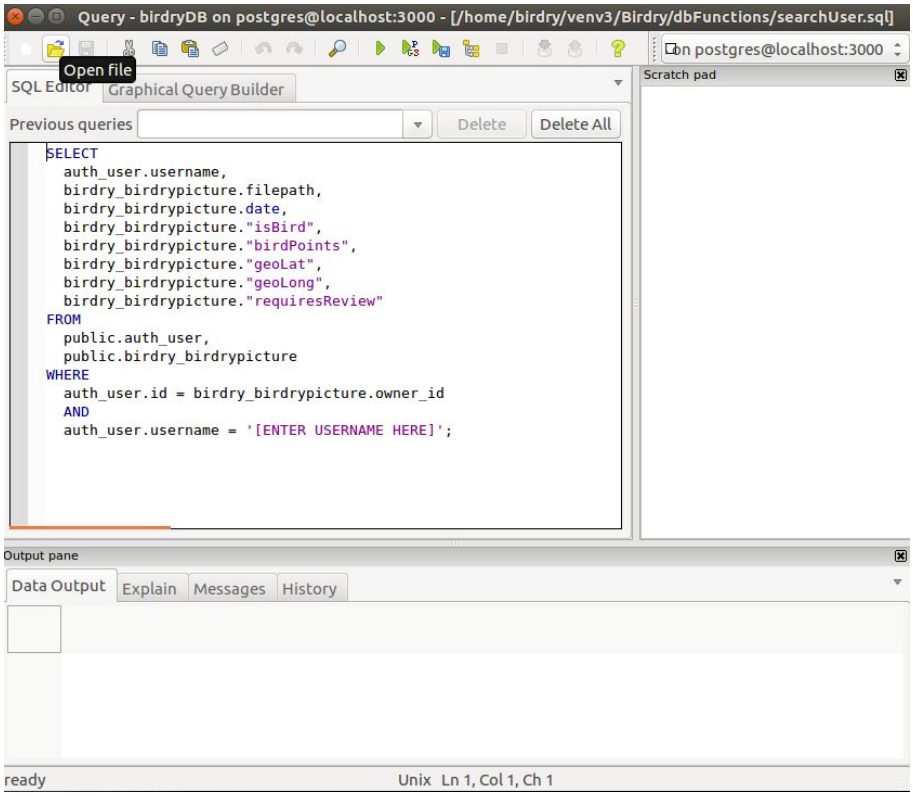
Invalid Results



Test Output Summary:

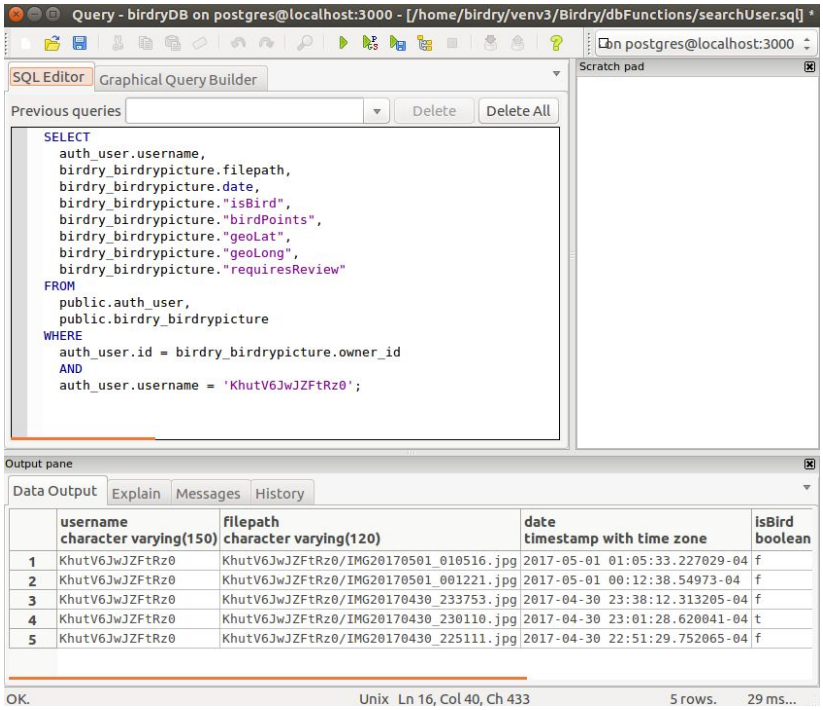
Login procedure experienced expected results. Correct login with PGAdminIII allowed access to the database's contents, while an incorrect credential provided to the login prevented access.

Use Case #9 - Conducted By: David Leiberg



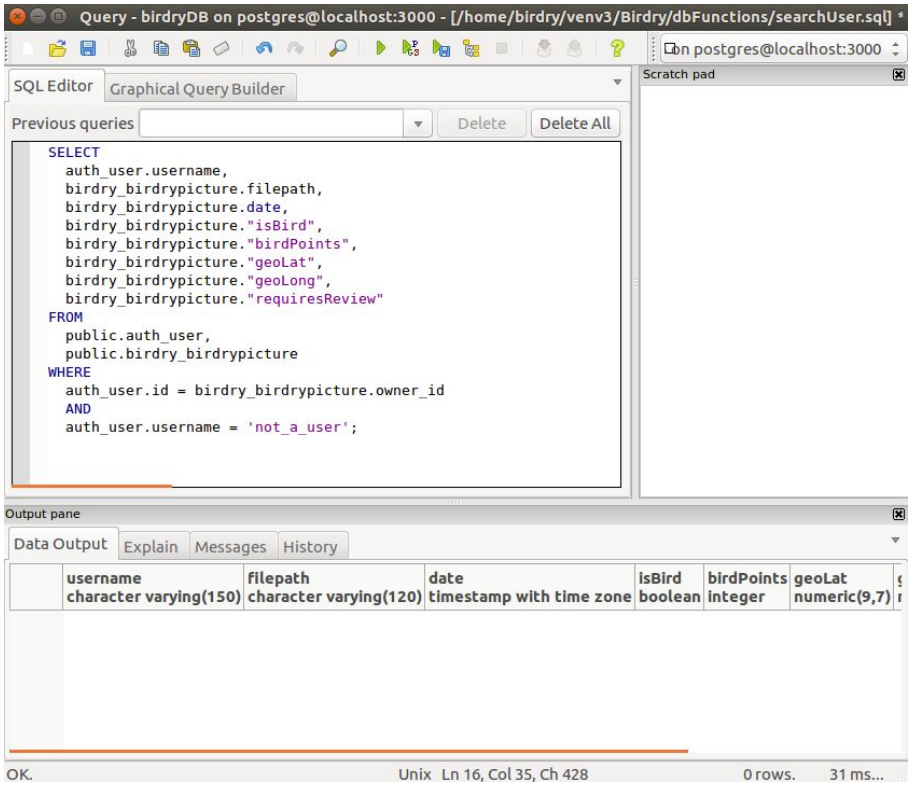
Using pre-fabricated SQL query from file searchUser.sql

Valid Results



List of associated pictures in data output pane at bottom

Invalid Results



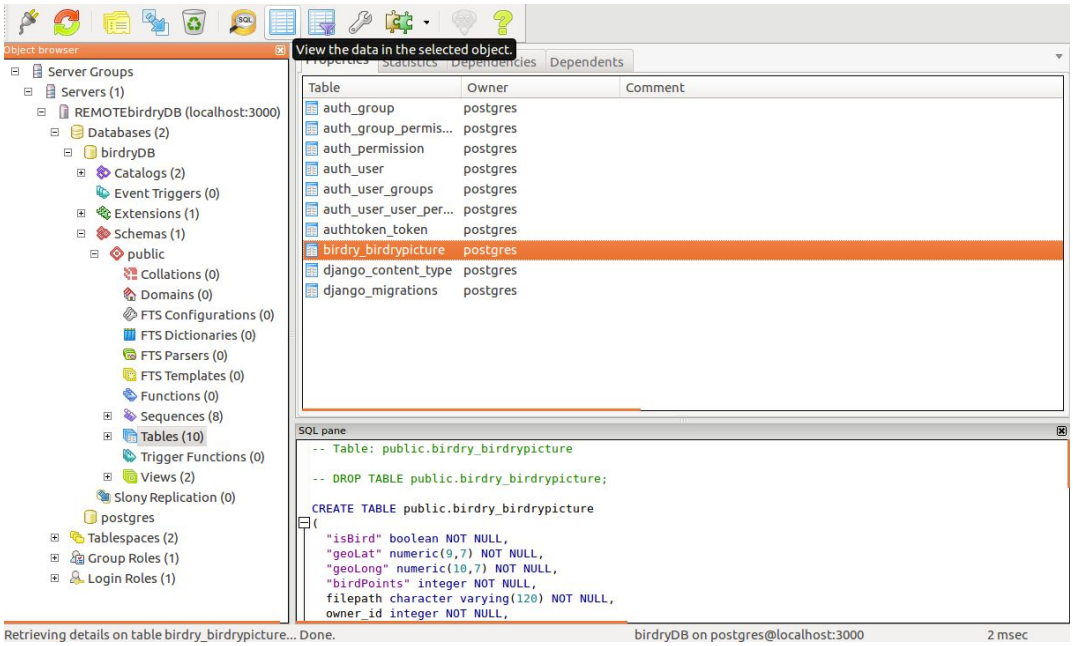
Empty set of pictures associated with invalid user in bottom frame

Test Output Summary:

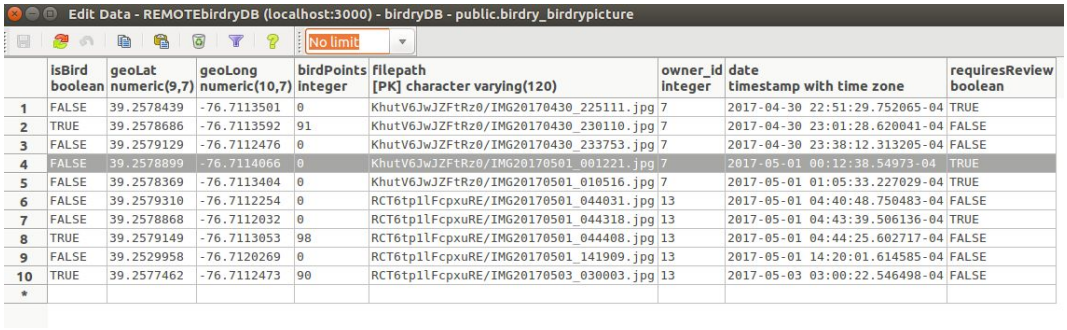
Results for this test were as expected. Using PGAdminIII's SQL query tool and the pre-constructed user-search query from the file searchUser.sql (included with the overall server code), a search for a valid user ID (for a user who has sent pictures) generates a list of associated pictures, while an invalid ID simply generates an empty table or set. A method baked into the database in some form like a view, such as the filtering options (see the test below for Use Case #13), would be preferable to remove extra steps of opening an external file.

Use Case #10 - Conducted By: David Leiberg

Valid Results



Selecting the table of all picture information to view



	isBird boolean	geoLat numeric(9,7)	geoLong numeric(10,7)	birdPoints integer	filepath [PK] character varying(120)	owner_id integer	date timestamp with time zone	requiresReview boolean
1	FALSE	39.2578439	-76.7113501	0	KhutV6JwJZFtRz0/IMG20170430_225111.jpg	7	2017-04-30 22:51:29.752065-04	TRUE
2	TRUE	39.2578686	-76.7113592	91	KhutV6JwJZFtRz0/IMG20170430_230110.jpg	7	2017-04-30 23:01:28.620041-04	FALSE
3	FALSE	39.2579129	-76.7112476	0	KhutV6JwJZFtRz0/IMG20170430_233753.jpg	7	2017-04-30 23:38:12.313205-04	FALSE
4	FALSE	39.2578899	-76.7114066	0	KhutV6JwJZFtRz0/IMG20170501_001221.jpg	7	2017-05-01 00:12:38.54973-04	TRUE
5	FALSE	39.2578369	-76.7113404	0	KhutV6JwJZFtRz0/IMG20170501_010516.jpg	7	2017-05-01 01:05:33.227029-04	TRUE
6	FALSE	39.2579310	-76.7112254	0	RCT6tp1lFcpxuRE/IMG20170501_044031.jpg	13	2017-05-01 04:40:48.750483-04	FALSE
7	FALSE	39.2578868	-76.7112032	0	RCT6tp1lFcpxuRE/IMG20170501_044318.jpg	13	2017-05-01 04:43:39.506136-04	TRUE
8	TRUE	39.2579149	-76.7113053	98	RCT6tp1lFcpxuRE/IMG20170501_044408.jpg	13	2017-05-01 04:44:25.602717-04	FALSE
9	FALSE	39.2529958	-76.7120269	0	RCT6tp1lFcpxuRE/IMG20170501_141909.jpg	13	2017-05-01 14:20:01.614585-04	FALSE
10	TRUE	39.2577462	-76.7112473	90	RCT6tp1lFcpxuRE/IMG20170503_030003.jpg	13	2017-05-03 03:00:22.546498-04	FALSE
*								

Selecting and highlighting a single picture to better view information

Test Output Summary:
Results for this test were as expected. Using PGAdminIII's ability to graphically view all information in a table, an administrator is able to select and highlight which image they would like to see more information about

Use Case #11 - Conducted By: David Leiberg

Valid Results

geoLong numeric(10,7)	birdPoints integer	filepath [PK] character
-76.7113501	0	KhutV6JwJZFtRz
-76.7113592	91	KhutV6JwJZFtRz

Selecting field to alter

geoLong numeric(10,7)	birdPoints integer	filepath [PK] character va
-76.7113501	0	KhutV6JwJZFtRz0/
-76.7113592	81	KhutV6JwJZFtRz0/
-76.7112476	0	KhutV6JwJZFtRz0/

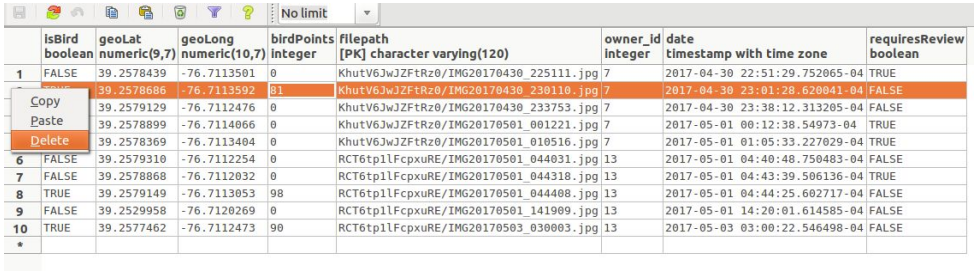
Altered field

Test Output Summary:

Test results were as expected. Using PGAdminIII's ability to view and edit data in tables graphically, the information fields of a selected image can be changed within the confines of their stated data type (as indicated at the top row of each image; the altered field shown takes integers).

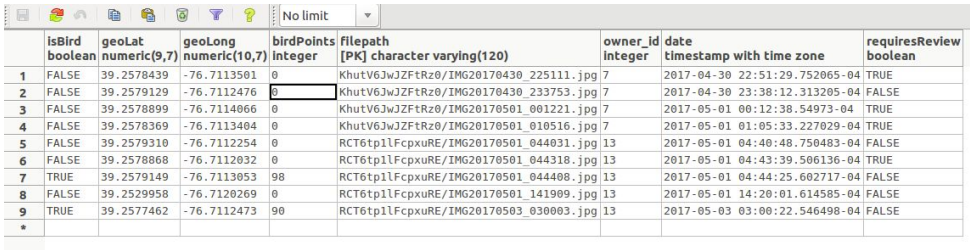
Use Case #12 - Conducted By: David Leiberg

Valid Results



	isBird boolean	geoLat numeric(9,7)	geoLong numeric(10,7)	birdPoints integer	filepath [PK] character varying(120)	owner_id integer	date timestamp with time zone	requiresReview boolean
1	FALSE	39.2578439	-76.7113501	0	KhutV6JwJZFtRz0/IMG20170430_225111.jpg	7	2017-04-30 22:51:29.752065-04	TRUE
	TRUE	39.2578686	-76.7113592	81	KhutV6JwJZFtRz0/IMG20170430_230110.jpg	7	2017-04-30 23:01:28.620041-04	FALSE
		39.2579129	-76.7112476	0	KhutV6JwJZFtRz0/IMG20170430_233753.jpg	7	2017-04-30 23:38:12.313205-04	FALSE
		39.2578899	-76.7114066	0	KhutV6JwJZFtRz0/IMG20170501_001221.jpg	7	2017-05-01 00:12:38.54973-04	TRUE
		39.2578369	-76.7113404	0	KhutV6JwJZFtRz0/IMG20170501_010516.jpg	7	2017-05-01 01:05:33.227029-04	TRUE
6	FALSE	39.2579310	-76.7112254	0	RCT6tp1lFcpxuRE/IMG20170501_044031.jpg	13	2017-05-01 04:40:48.750483-04	FALSE
7	FALSE	39.2578868	-76.7112032	0	RCT6tp1lFcpxuRE/IMG20170501_044318.jpg	13	2017-05-01 04:43:39.506136-04	TRUE
8	TRUE	39.2579149	-76.7113053	98	RCT6tp1lFcpxuRE/IMG20170501_044408.jpg	13	2017-05-01 04:44:25.602717-04	FALSE
9	FALSE	39.2529958	-76.7120269	0	RCT6tp1lFcpxuRE/IMG20170501_141909.jpg	13	2017-05-01 14:20:01.614585-04	FALSE
10	TRUE	39.2577462	-76.7112473	90	RCT6tp1lFcpxuRE/IMG20170503_030003.jpg	13	2017-05-03 03:00:22.546498-04	FALSE
*								

Selecting an image's information to delete



	isBird boolean	geoLat numeric(9,7)	geoLong numeric(10,7)	birdPoints integer	filepath [PK] character varying(120)	owner_id integer	date timestamp with time zone	requiresReview boolean
1	FALSE	39.2578439	-76.7113501	0	KhutV6JwJZFtRz0/IMG20170430_225111.jpg	7	2017-04-30 22:51:29.752065-04	TRUE
2	FALSE	39.2579129	-76.7112476	0	KhutV6JwJZFtRz0/IMG20170430_233753.jpg	7	2017-04-30 23:38:12.313205-04	FALSE
3	FALSE	39.2578899	-76.7114066	0	KhutV6JwJZFtRz0/IMG20170501_001221.jpg	7	2017-05-01 00:12:38.54973-04	TRUE
4	FALSE	39.2578369	-76.7113404	0	KhutV6JwJZFtRz0/IMG20170501_010516.jpg	7	2017-05-01 01:05:33.227029-04	TRUE
5	FALSE	39.2579310	-76.7112254	0	RCT6tp1lFcpxuRE/IMG20170501_044031.jpg	13	2017-05-01 04:40:48.750483-04	FALSE
6	FALSE	39.2578868	-76.7112032	0	RCT6tp1lFcpxuRE/IMG20170501_044318.jpg	13	2017-05-01 04:43:39.506136-04	TRUE
7	TRUE	39.2579149	-76.7113053	98	RCT6tp1lFcpxuRE/IMG20170501_044408.jpg	13	2017-05-01 04:44:25.602717-04	FALSE
8	FALSE	39.2529958	-76.7120269	0	RCT6tp1lFcpxuRE/IMG20170501_141909.jpg	13	2017-05-01 14:20:01.614585-04	FALSE
9	TRUE	39.2577462	-76.7112473	90	RCT6tp1lFcpxuRE/IMG20170503_030003.jpg	13	2017-05-03 03:00:22.546498-04	FALSE
*								

After deleting the selected information. Note there is one less row.

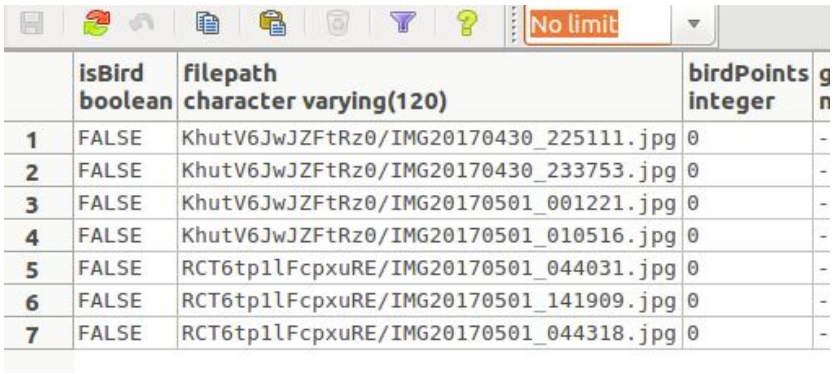
Test Output Summary:

The results of this test were as expected. PGAdminIII's graphical table view allows for alterations of a viewable table, including the dropping or deleting of records they contain. In this case, a chosen row (corresponding to a user-sent image) is deleted, leaving the table one row smaller. It should be noted this only removes information about the image from the database; it does not alter the image that is stored separately in the filesystem on disk in anyway (as intended).

Use Case #13 - Conducted By: David Leiberg

Valid Results

Available views filtering by depiction of birds or "not birds"



	isBird boolean	filepath character varying(120)	birdPoints integer	g n
1	FALSE	KhutV6JwJZFtRz0/IMG20170430_225111.jpg	0	-
2	FALSE	KhutV6JwJZFtRz0/IMG20170430_233753.jpg	0	-
3	FALSE	KhutV6JwJZFtRz0/IMG20170501_001221.jpg	0	-
4	FALSE	KhutV6JwJZFtRz0/IMG20170501_010516.jpg	0	-
5	FALSE	RCT6tp1lFcpxuRE/IMG20170501_044031.jpg	0	-
6	FALSE	RCT6tp1lFcpxuRE/IMG20170501_141909.jpg	0	-
7	FALSE	RCT6tp1lFcpxuRE/IMG20170501_044318.jpg	0	-

View of "not birds;" note that all isBird indicators are False

	isBird boolean	filepath character varying(120)	birdPoints integer	g n
1	TRUE	RCT6tp1lFcpxuRE/IMG20170501_044408.jpg	98	-
2	TRUE	RCT6tp1lFcpxuRE/IMG20170503_030003.jpg	90	-

View of "all birds;" not all isBird indicators are True

Test Output Summary:

This test produced the expected results. Accessing the pre-constructed views within the database from PGAdminIII, allBirds and notBirds, the picture data is separated and viewable (i.e. filtered) effectively on the isBird indicator of each picture's record.

Appendix A – Team Review Sign-off

We, the members of the development team, undersign here to indicate that every member of the team has reviewed and approved the contents and format of this Testing Report, and that their comments and/or concerns (if any) are listed here.

Name: _____ Signature: _____ Date: _____

Comments:

_____

_____

Name: _____ Signature: _____ Date: _____

Comments:

_____

_____

Name: _____ Signature: _____ Date: _____

Comments:

_____

_____

Name: _____ Signature: _____ Date: _____

Comments:

_____

_____

Name: _____ Signature: _____ Date: _____

Comments:

_____

_____

Appendix B – Document Contributions

Below is an estimate of the contribution from each team member, including specific work done and percentage of total document completed.

Kyle Fritz
Contributions:
- First draft of Section 3 Table
- Conducted Use Case #1 Test and fixes for that Use Case
- Edits made throughout document for grammar
- Document Review

Est. Contribution: 15%

Laras Istiqomah
Contributions:
- 2.2 Testing Sessions coauthor
- Document review

Est. Contribution: 10%

David Leiberg
Contributions:
- 3.2 Test Results for Use Cases 8 - 13
- Document formatting

Est. Contribution: 20%

Julian Sniffen
Contributions:
- 2.2 Testing Sessions coauthor
- 2.3 Impressions of Process
- 3.1 Testing Suite
- 3.2 Test Results for Use Cases 1 - 7

Est. Contribution: 45%

Nicholay Topin
Contributions:
- Co-conducted Use Case 2-5 Tests
- Document formatting, review, and generous edits

Est. Contribution: 10%