

Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT202-008-S2024/it202-api-project-milestone-3-2024/grade/lm457>

IT202-008-S2024 - [IT202] API Project Milestone 3 2024

Submissions:

Submission Selection

1 Submission [active] 4/29/2024 6:12:26 PM

Instructions

[^ COLLAPSE ^](#)

Implement the Milestone 3 features from the project's proposal document: <https://docs.google.com/document/d/1XE96a8DQ52Vp49XACBDTNCq0xYDt3kF29cO88E>
Make sure you add your ucid/date as code comments where code changes are done
All code changes should reach the Milestone3 branch
Create a pull request from Milestone3 to dev and keep it open until you get the output PDF from this assignment.
Gather the evidence of feature completion based on the below tasks.
Once finished, get the output PDF and copy/move it to your repository folder on your local machine.
Run the necessary git add, commit, and push steps to move it to GitHub
Complete the pull request that was opened earlier
Create and merge a pull request from dev to prod
Upload the same output PDF to Canvas

Branch name: Milestone3

Tasks: 26 Points: 10.00

 API (1 pt.)

[^ COLLAPSE ^](#)

 Task #1 - Points: 1

Text: Data Related to Users

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	What's the concept/association?
<input checked="" type="checkbox"/> #2	1	What sort of relationship is it (one to many, many to one, many to many, etc)
<input checked="" type="checkbox"/> #3	1	Note any other considerations

Response:

A user can create their travel bucket list by favoriting their desired locations. So, the association is between the User table and my tourist_info table. Since many users can favorite many different locations, and each location can be favorited by multiple users, the relationship is a many-to-many relationship. Therefore, a new relationship entity table called UserLocations is created that associates user_ids with locations_ids.

Task #2 - Points: 1

Text: Updating Entities

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	When an update occurs either manually or from the API how does it affect associated data?
<input checked="" type="checkbox"/> #2	1	Do users see the old data, new data, does data need to be reassigned, etc?

Response:

When either more data is fetched from the API, or manual insertions/editing are made to entities, the associated data is also immediately updated and reflects the new changes. The users are presented with the new fresh data on all of their pages. For instance, if I were to edit the title of one of my locations that is associated with a certain user, the title change would now be reflected in their bucket list page (logged in user's associated page).

Handle Data Association (1 pt.)



Task #1 - Points: 1

Text: Screenshots of the code

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Option 1: Related pages will have a button to do association (like favorites or similar), Option 2: a separate page will be used to associate entities to a user by some other user (like assignment of entities)
<input checked="" type="checkbox"/> #2	1	Include ucid/date comments for each code screenshot
<input checked="" type="checkbox"/> #3	1	Clearly caption screenshots

Task Screenshots:

Small Medium Large

```

use \App\services\logger\Logger;
use \App\models\Output;

<?php
if (!isset($output)) {
    error_log("Using output partial without data");
    flash("Dev Alert: output called without data", "danger");
}
if (isset($output)) {
    <?php if (isset($output['username'])) : ?>
        <div class="card mx-auto" style="width: 18rem;">
            <?php if (isset($output['username'])) : ?>
                <div class="card-header">
                    <strong>Selected By:</strong><?php se($output, "username", "N/A"); ?>
                </div>
            <?php endif; ?>
            <?php if (isset($output['total_users'])) : ?>
                <div class="card-header">
                    <strong>Favorited By:</strong><?php se($output, "total_users", "N/A"); ?> users
                </div>
            <?php endif; ?>
            <div class="card-body">
                <div class="card-title"><strong><?php se($output, "name", "Unknown") ?></strong><?php se($output, 'rarity'); ?></div>
                <div class="card-text">
                    <ul class="list-group">
                        <li class="list-group-item"><strong>National ID:</strong><?php se($output, "nationalID", "Unknown"); ?></li>
                        <li class="list-group-item"><strong>Name:</strong><?php se($output, "name", "Unknown"); ?></li>
                        <li class="list-group-item"><strong>Rating:</strong><?php se($output, "rating", "Unknown"); ?></li>
                        <li class="list-group-item"><strong>Number of Reviews:</strong><?php se($output, "num_reviews", "Unknown"); ?></li>
                        <li class="list-group-item"><strong>Website:</strong><?php se($output, "website", "Unknown"); ?></li>
                        <li class="list-group-item"><strong>Address:</strong><?php se($output, "address", "Unknown"); ?></li>
                        <li class="list-group-item"><strong>Phone:</strong><?php se($output, "phone", "Unknown"); ?></li>
                    </ul>
                </div>
            <div class="card-body">
                <?php if (isset($output['id'])) : ?>
                    <a class="btn btn-secondary" href="<?php echo get_url('viewlocations.php?id=' . $output['id']); ?>">View</a>
                <?php endif; ?>
                <?php if (isset($output['user_id'])) : ?>
                    <a class="btn btn-primary" href="<?php echo get_url('profile.php?id=' . $output['user_id']); ?>"><?php se($output, "username"); ?>'s Profile</a>
                <?php endif; ?>
                <?php if (isset($output['user_id']) && isset($output['places_id'])) : ?>
                    <a class="btn btn-danger" href="<?php echo get_url('delete_association.php?user_id=' . $output['user_id'] . '&places_id=' . $output['places_id']); ?>">Remove</a>
                <?php endif; ?>
                <?php if ($output['is_favorite'] == 0 || $output['user_id'] === "N/A") : ?>
                    <div class="card-body">
                        <?php $id = isset($output['id']) ? $output['id'] : (isset($_GET['id']) ? $_GET['id'] : -1); ?>
                        <div class="d-flex justify-content-center">
                            <a href="<?php echo get_url('api/bucketlist_locations.php?places_id=' . $output['id']); ?>" class="btn btn-success">Add to Travel Bucket List</a>
                        </div>
                    </div>
                <?php else : ?>
                    <div class="card-body">
                        <div class="bg-warning text-dark text-center">Output not available</div>
                    </div>
                <?php endif; ?>
            </div>
        </div>
    <?php endif; ?>
</div>
</div>

```

My card layout file that processes my results on any given page and outputs each location in a card format. The ability to associate (favorite) a location to a user is only available if the current user has not favorited the location.

Checklist Items (3)

#1 Option 1: Related pages will have a button to do association (like favorites or similar), Option 2: a separate page will be used to associate entities to a user by some other user (like assignment of entities)

#2 Include ucid/date comments for each code screenshot

#3 Clearly caption screenshots

```

use \App\services\logger\Logger;
use \App\models\Output;

<?php
// This is an internal API endpoint to receive data and do something with it
// this is not a standalone page
// Note: no nav.php here because this is a temporary stop, it's not a user page
require(__DIR__ . "/../../../../lib/functions.php");
session_start();
if (isset($_GET["places_id"]) && is_logged_in()) {
    // [UNC] implement purchase logic (for now it's all free)
    $db = getDB();
    $query = "INSERT INTO `UserLocations` (user_id, places_id) VALUES (:user_id, :places_id)";
    try {
        $stmt = $db->prepare($query);
        $stmt->execute([":user_id" => get_user_id(), ":places_id" => $_GET["places_id"]]);
        flash("Congrats you added this location to your Travel Bucket List!", "success");
        redirect("my_locations.php");
    } catch (PDOException $e) {
        if ($e->getCode() == 1062) {
            flash("This location is already in your Travel Bucket List!", "warning");
        }
    }
}

```

```

    if ($e->errorInfo[1] == 1062) {
        flash("This location has already been favorited", "danger");
    } else {
        flash("Unhandled error occurred", "danger");
    }
    error_log("Error adding location: " . var_export($e, true));
} <- #16-23 catch (PDOException $e)
} <- #7-24 if (isset($_GET["places_id"])) && is_logged_in()

//for now I'll redirect, but if I later use AJAX I need to send a reply instead
redirect("locations.php"); | You, 2 days ago + handled api data association with user and cr

```

The actual file responsible for associating a user with a location. The previous card layout links the association handling to here, where an insert statement is used to create an entry for the logged-in user and their associated (favorited) location.

Checklist Items (3)

#1 Option 1: Related pages will have a button to do association (like favorites or similar), Option 2: a separate page will be used to associate entities to a user by some other user (like assignment of entities)

#2 Include ucid/date comments for each code screenshot

#3 Clearly caption screenshots

Task #2 - Points: 1

Text: Screenshot of the association table(s)

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Show the table(s) you made to handle the associations
<input checked="" type="checkbox"/> #2	1	Should have some example data
<input checked="" type="checkbox"/> #3	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

	Q	id int	user_id int	places_id int	is_active tinyint(1)	created timestamp	modified timestamp
1	> 1	23	22	53	1	2024-04-28 04:28:57	2024-04-28 04:28:57
2	> 2	44	2	36	1	2024-04-29 04:04:37	2024-04-29 04:04:37
3	> 3	45	2	38	1	2024-04-29 04:04:49	2024-04-29 04:04:49

✓ 3	43	2	50	1	2024-04-29 04:04:48	2024-04-29 04:04:48
✓ 4	51	2	53	1	2024-04-29 04:34:05	2024-04-29 04:34:05
✓ 5	52	23	61	1	2024-04-29 04:42:31	2024-04-29 04:42:31

My relationship entity table that handles all of my associations between users and locations. It contains the primary keys (user_id and places_id) of both my Users and tourist_info tables.

Checklist Items (3)

#1 Show the table(s) you made to handle the associations

#2 Should have some example data

#3 Clearly caption screenshots

Task #3 - Points: 1

Text: Explain solution

Checklist			*The checkboxes are for your own tracking
#	Points	Details	
<input checked="" type="checkbox"/> #1	1	Mention which option your application handles regarding association	
<input checked="" type="checkbox"/> #2	1	Describe each column/association table	

Response:

I chose option 1 for my project to allow the user the ability to associate with my API/custom tourist location data in the form of a Travel Bucket List. Each logged in user can browse through all of my locations and favorite all of their desired travel destinations which will be added to their bucket list. Such association is handled via a singular many-to-many relationship table between the Users table and the tourist_info table called the UserLocations table. It contains the primary keys of both tables, user_id and places_id, as its foreign keys and allows for many-to-many associations between users and locations. So several users can choose multiple locations and each location can be chosen by several users. The table also contains a virtual is_favorite column depending on the page that essentially stores a binary value of either 1 or 0 to indicate whether the current logged in user has favorited a certain entity or not.

Task #4 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
■ #1	1	Include the heroku prod link for the page that creates the association
■ #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

<https://it202-lm457-prod-cf97d0c34365.herokuapp.com/project/locations.php>

URL #2

<https://github.com/Kith07/spring2024-IT202/pull/69>

URL #3

<https://github.com/Kith07/spring2024-IT202/pull/76>

● Current User's Association Page (2 pts.)

[^COLLAPSE ^](#)

● Task #1 - Points: 1

Text: Screenshots of this page

● Task #1 - Points: 1

Text: Screenshots of this page

Checklist

*The checkboxes are for your own tracking

#	Points	Details
■ #1	1	Show the summary of the results with relevant information per entity
■ #2	1	Show the single view buttons/links, delete button/links, and delete all button/link
■ #3	1	Show variations of the number of shown items count and show the count of total number of associated items to the user
■ #4	1	Show variations of the filter/sort including no results (proper message should be visible)
■ #5	1	Make sure the heroku dev url is visible in the address bar
■ #6	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

Name	Rating	Reviews	Date	Sort	Order	Limit
Fremont Street Experience	4.5	102514	mm/dd/yyyy	Rating	+	10
Shelby American, Inc.	4.5	240034	mm/dd/yyyy	Rating	+	10
KAA by Cirque du Soleil	4.5	501163	mm/dd/yyyy	Rating	+	10

Rating: 4.0	Number of Reviews: 1547	Number of Reviews: 8619
Website: http://vegasexperience.com/viva-vision-light-show	Website: http://www.mgmgrand.com/enter-tainment/ka-cirque-du-soiel-show.aspx	Address: 3799 Las Vegas Blvd S MGM Grand, Las Vegas, NV 89109-4319
Address: 425 Fremont Street, Las Vegas, NV 89101-5620	Phone: +1 702-942-7325	Phone: +1 702-531-3626
Phone: +1 702-678-5600		
View	View	View
sai's Profile	sai's Profile	sai's Profile
Remove	Remove	Remove
Output not available		

My current logged in user's associated entities page (favorite locations in bucket list) upon page load. Only a glimpse of the total information is available on each entity here. Filter/sort functionality is also applicable here.

Checklist Items (1)

#1 Show the summary of the results with relevant information per entity

The screenshot shows a web application interface titled "Tourist Attractions". At the top right are navigation links: Home, Profile, and Locations. A red trash can icon is in the top right corner of the main content area. The main title is "My Travel Bucket List". Below it are two search input fields: "ID Search" and "Name Search", followed by "Filter", "Clear", and "Remove All Locations" buttons. A message "Results: 3/3" is displayed above a card for "Fremont Street Experience". The card includes the following details:
Fremont Street Experience
National ID: 102514
Name: Fremont Street Experience
Rating: 4.0
Number of Reviews: 45217
Website: http://vegasexperience.com/viva-vision-light-show
Address: 425 Fremont Street, Las Vegas, NV 89101-5620
Phone: +1 702-678-5600
At the bottom of the card are four buttons: "View" (grey), "sai's Profile" (blue), "Remove" (red), and "Output not available" (yellow).

The single-user entity association deletion button at the bottom of the page. The single view entity button in the grey right on top of that. The public user profile button as well as the remove all user entity associations button in red at the top.

Checklist Items (1)

#2 Show the single view buttons/links, delete button/links, and delete all button/link

The screenshot shows a browser window with a toolbar at the bottom containing various icons and tabs. The address bar shows the URL: https://it202-lm457-dev-7b44497d46f0.herokuapp.com/project/my_locations.php. The browser interface includes standard controls like back, forward, and search.

My Travel Bucket List

ID Search Name Search Minimum Rating Minimum Reviews Minimum Date Maximum Date Sort Order Limit

ID Search Name Search Minimum Rating Minimum Reviews mm/dd/yyyy mm/dd/yyyy Rating + - 10

Filter Clear Remove All Locations

Results: 2/2

Selected By: sai	Selected By: sai
Fremont Street Experience	Shelby American, Inc.
National ID: 102514	National ID: 2400034
Name: Fremont Street Experience	Name: Shelby American, Inc.
Rating: 4.0	Rating: 4.5
Number of Reviews: 45217	Number of Reviews: 1547
Website: http://vegasexperience.com/las-vegas-light-show	Website: http://shelbyamerican.com/
Address: 425 Fremont Street, Las Vegas, NV 89101-5620	Address: 8405 Ensworth St, Las Vegas, NV 89119-3211
Phone: +1 702-678-5600	Phone: +1 702-942-7325
View	View
sai's Profile	sai's Profile
Remove	Remove
Output not available	

Part 1 of bucket list page with 2/2 entities currently displayed on page, indicating that 2 entities are associated with the logged in user and both of them are on display at the moment.

Checklist Items (1)

#3 Show variations of the number of shown items count and show the count of total number of associated items to the user

CS248 CS381 IT202 Copy of location: Miletto Kmetz Learn R202-Im Pull req. JMiletto H202-Im

https://it202-im457-dev-7b44497d46f0.herokuapp.com/project/my_locations.php?NationalID=&name=she&min_rating=&min_num_reviews=&date_min=&date_max=...

Tourist Attractions Home Profile Locations Available Locations My Travel Bucket List Admin Tourist Locations Logout

My Travel Bucket List

ID Search Name Search Minimum Rating Minimum Reviews Minimum Date Maximum Date Sort Order Limit

ID Search Name Search Minimum Rating Minimum Reviews mm/dd/yyyy mm/dd/yyyy Rating + - 10

Filter Clear Remove All Locations

Results: 1/2

Selected By: sai
Shelby American, Inc.
National ID: 2400034
Name: Shelby American, Inc.
Rating: 4.5
Number of Reviews: 1547
Website: http://shelbyamerican.com/
Address: 8405 Ensworth St, Las Vegas, NV 89119-3211
Phone: +1 702-942-7325
View
sai's Profile
Remove
Output not available

The same page but with a name filter applied. As shown in the image, now the results are 1/2, indicating that though there are still 2 entities associated with the logged-in user, only one of them 1 is currently on display due to the filter.

Checklist Items (1)

#3 Show variations of the number of shown items count and show the count of total number of associated items to the user

The screenshot shows a web browser window with multiple tabs open. The active tab displays a travel bucket list page titled "My Travel Bucket List". The page includes search filters for ID, name, rating, reviews, date, sort, order, and limit. Below the filters, there are four items listed:

- Fremont Street Experience**
National ID: 102514
Name: Fremont Street Experience
Rating: 4.0
Number of Reviews: 45217
Website: <http://vegasexperience.com/viva-vision-light-show>
Address: 425 Fremont Street, Las Vegas, NV 89101-5620
Phone: +1 702-678-5600
Buttons: View, sa's Profile, Remove, Output not available
- KAA by Cirque du Soleil**
National ID: 501163
Name: KAA by Cirque du Soleil
Rating: 4.5
Number of Reviews: 6619
Website: <http://www.mgmgrand.com/center-tainment/ka-cirque-du-soleil-show.aspx>
Address: 3799 Las Vegas Blvd S MGM Grand, Las Vegas, NV 89105-4319
Phone: +1 702-531-3826
Buttons: View, sa's Profile, Remove, Output not available
- High Roller**
National ID: 6509306
Name: High Roller
Rating: 4.5
Number of Reviews: 25877
Website: <http://www.caesars.com/linq/high-roller>
Address: 3645 Las Vegas Blvd S between The LINQ Hotel & Casino and Flamingo Las Vegas, Las Vegas, NV 89109-8921
Phone: +1 702-322-0593
Buttons: View, sa's Profile, Remove, Output not available
- The LINQ Promenade**
National ID: 6537951
Name: The LINQ Promenade
Rating: 4.5
Number of Reviews: 3347
Website: <https://www.caesars.com/linq>
Address: 3645 Las Vegas Blvd S between The LINQ Hotel & Casino and Flamingo Las Vegas, Las Vegas, NV 89109-8921
Phone: +1 702-322-0543
Buttons: View, sa's Profile, Remove, Output not available

Part 3 of another variation with more associated entities with the current user, leading to a result of 4/4.

Checklist Items (1)

#3 Show variations of the number of shown items count and show the count of total number of associated items to the user

The screenshot shows a web browser window with multiple tabs open. The active tab displays a travel bucket list page titled "My Travel Bucket List". The page includes search filters for ID, name, rating, reviews, date, sort, order, and limit. The "Minimum Rating" field is set to 8, and the "Minimum Num. Reviews" field is set to 20000. Below the filters, there are two items listed:

- Fremont Street Experience**
National ID: 102514
Name: Fremont Street Experience
Rating: 4.0
Number of Reviews: 45217
Website: <http://vegasexperience.com/viva-vision-light-show>
Address: 425 Fremont Street, Las Vegas, NV 89101-5620
Phone: +1 702-678-5600
Buttons: View, sa's Profile, Remove, Output not available
- High Roller**
National ID: 6509306
Name: High Roller
Rating: 4.5
Number of Reviews: 25877
Website: <http://www.caesars.com/linq/high-roller>
Address: 3645 Las Vegas Blvd S between The LINQ Hotel & Casino and Flamingo Las Vegas, Las Vegas, NV 89109-8921
Phone: +1 702-322-0590
Buttons: View, sa's Profile, Remove, Output not available

The minimum reviews filter of 20000 is being applied here. As you can see only 2 of the 4 associated entities pop up that have more than 20000 reviews.

Checklist Items (1)

#4 Show variations of the filter/sort including no results (proper message should be visible)

Lion Habitat Ranch	High Roller	The LINQ Promenade	Fremont Street Experience
National ID: 3819666	National ID: 6509306	National ID: 6637061	National ID: 102514
Name: Lion Habitat Ranch	Name: High Roller	Name: The LINQ Promenade	Name: Fremont Street Experience
Rating: 5.0	Rating: 4.5	Rating: 4.6	Rating: 4.0
Number of Reviews: 1639	Number of Reviews: 25877	Number of Reviews: 3347	Number of Reviews: 45217
Website: http://www.lionhabitatranch.com/	Website: http://www.caesars.com/linq/high-roller	Website: http://www.caesars.com/linq	Website: http://vegashexperience.com/viva-vision-light-show
Address: 382 Brunet Ave, Henderson, NV 89044-8746	Address: 3645 Las Vegas Blvd S between The LINQ Hotel & Casino and Flamingo Las Vegas, Las Vegas, NV 89109-8978	Address: 3645 Las Vegas Blvd S between The LINQ Hotel & Casino and Flamingo Las Vegas, Las Vegas, NV 89109-8921	Address: 425 Fremont Street, Las Vegas, NV 89101-5620
Phone: +1 702-595-6600	Phone: +1 702-322-0580	Phone: +1 702-322-0643	Phone: +1 702-678-5800
View	View	View	View
sa's Profile	sa's Profile	sa's Profile	sa's Profile
Remove	Remove	Remove	Remove
Output not available	Output not available	Output not available	Output not available

The sort by rating in descending order is applied here. As you can see, from left to right, the entities(locations) go from highest to lowest rating.

Checklist Items (1)

#4 Show variations of the filter/sort including no results (proper message should be visible)

High Roller
National ID: 6509306

Name:	High Roller
Rating:	4.5
Number of Reviews:	25877
Website:	http://www.caesars.com/lv/high-roller
Address:	3545 Las Vegas Blvd South The Linq Promenade, Las Vegas, NV 89109-6979
Phone:	+1 702-322-0583
View	
sa's Profile	
Remove	
Output not available	

The name search filter being applied, resulting in only one associated entity whose name has a partial match with the string 'High'.

Checklist Items (1)

#4 Show variations of the filter/sort including no results (proper message should be visible)

Tourist Attractions Home Profile Locations Available Locations My Travel Bucket List Admin Tourist Locations Logout

My Travel Bucket List

ID Search Name Search Minimum Rating Minimum Reviews Minimum Date Maximum Date Sort Order Limit

ID Search: Highy Name Search: Minimum Rating: Minimum Reviews: mm/dd/yyyy: mm/dd/yyyy: Sort: Rating: Order: * Limit: 10

Filter Clear Remove All Locations

Results: 0/4
No results to show

The name filter with an entry that doesn't match any associated entity so the no results message pops up.

Checklist Items (1)

#4 Show variations of the filter/sort including no results (proper message should be visible)

Task #2 - Points: 1
Text: Screenshot the code



Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Show the code related to fetching the user's associations (including the query)
<input checked="" type="checkbox"/> #2	1	Show the code related to the display of the results
<input checked="" type="checkbox"/> #3	1	Each record should have a button/link for single view
<input checked="" type="checkbox"/> #4	1	Each record should have a button/link for delete (this may be an admin-only thing but should be present for the specific role) Note: this is to delete the relationship and not the specific entity
<input checked="" type="checkbox"/> #5	1	Show the logic for deleting all associations for the user (this may be admin-only but should be present for the specific role)
<input checked="" type="checkbox"/> #6	1	Show the logic related to the count of all associated items to the user (even the ones not shown in the filtered results)
<input checked="" type="checkbox"/> #7	1	Show the logic related to the count of the items on the page (this value should change based on the filter applied)
<input checked="" type="checkbox"/> #8	1	Show the logic related to filter/sort (limit should be constrained to 1-100 otherwise default to 10)
<input checked="" type="checkbox"/> #9	1	Include ucid/date comments for each code screenshot
<input checked="" type="checkbox"/> #10	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

```

total_records = get_total_count("tourist_info t JOIN userlocations u ON t.id = u.places_id WHERE user_id = :user_id", [":user_id" => get_user_id()]);
$location = "SELECT username, t.id, location_id, language, currency, nationalID, name, ranking, description, rating, num_reviews, website, address, phone, write_review, monday_open, monday_close, tuesday_open, tuesday_close, wednesday_open, wednesday_close, thursday_open, thursday_close, friday_open, friday_close, saturday_open, saturday_close, sunday_open, sunday_close, popular_time_title, primary_category, price, partner, tour_url, product_code, is_api, t.created, user_id, ut.places_id,
IF (ut.user_id = :current_user_id, 1, 0) AS is_favorite FROM tourist_info t JOIN userlocations u ON t.id = u.places_id LEFT JOIN users u ON u.id = ut.user_id WHERE user_id = :user_id";
$queries = ["!current_user_id" => get_user_id(), ":user_id" => get_user_id()];

session_key = $_SERVER["SCRIPT_NAME"];
$ls_clear = isset($_GET["clear"]);
if ($ls_clear) {
    session_delete(session_key);
    unset($_GET["clear"]);
    redirect(session_key);
} else {
    $session_data = session_load(session_key);
}

if (count($_GET) == 0 && !isset($session_data) && count($session_data) > 0) { //ICD1 | LAST
    if ($session_data) {
        $_GET = $session_data;
    }
}

if (count($_GET) > 0) { //DATE: 4/29/2024
    if (count($_GET) > 1) {
        $url = http_build_query($_GET);
    } else {
        $url = $_GET[0];
    }
}

$stmt = $db->prepare($query);
$results = [];
try {
    $stmt->execute($params);
    $r = $stmt->fetchall();
    if ($r) {
        $results = $r;
    }
} catch (Exception $e) {
    error_log("Error Fetching Locations: " . var_export($e, true));
    flash("An error occurred, please try again", "danger");
}

```

The code and the query utilized to retrieve all associated entities given a logged in user's id.

Checklist Items (1)

#1 Show the code related to fetching the user's associations (including the query)

```
$table = ["data" => $results, "title" => "List of Tourist Locations Data", "ignored_columns" => ["id", "location_id", "language", "currency", "description", "write_review", "monday_open", "monday_close", "tuesday_open", "tuesday_close", "wednesday_open", "wednesday_close", "thursday_open", "thursday_close", "friday_open", "friday_close", "saturday_open", "saturday_close", "sunday_open", "sunday_close", "popular_tour_title", "primary_category", "price", "partner", "tour_url", "product_code", "is_api"]];  
}  
  
<div class="container-fluid">  
    <h1>My Travel Bucket List</h1>  
    <form method="GET">  
        <div class="row mb-3" style="align-items: center;">  
            <?php foreach ($form as $k => $v) : ?>  
                <div class="col">  
                    <?php render_input($v); ?>  
                </div>  
            </?php endforeach; ?>  
        </div>  
        <div style="display: flex-and; align-items: center;">  
            <?php render_button(['text' => "Filter", "type" => "submit"]); ?>  
            <a href="#" class="btn btn-secondary ml-2" onclick="event.preventDefault(); document.getElementById('clear').click();">Clear</a>  
            <a href="#" class="btn btn-danger ml-2" onclick="if(confirm('Are you sure?')) event.preventDefault(); document.getElementById('removeAll').click();">Remove All Locations</a>  
        </div>  
    </form>  
    <?php render_result_counts(count($results), $total_records); ?>  
    <div class="row w-100 row-cols-auto row-cols-sm-1 row-cols-md-2 row-cols-lg-3 row-cols-xl-4 row-cols-xxl-5 g-4">  
        <?php foreach ($results as $loc) : ?>  
            <div class="col">  
                <?php render_tourist_card($loc); ?>  
            </div>  
        </?php endforeach; ?>  
        <?php if (count($results) === 0) : ?>  
            <div class="col">  
                No results to show  
            </div>  
        </?php endif; ?>  
    </div>  
</?php  
//note we need to go up 1 more directory  
require_once __DIR__ . '/../../partials/_flash.php';  
?>
```

The front-end code responsible for displaying the results from the query onto the webpage. Utilizes two helper functions, `render_result_counts` and `render_tourist_card` to display the number of results on the screen as well as output the entities in a card layout respectively.

Checklist Items (1)

#2 Show the code related to the display of the results

```
<div class="card body">  
    <?php if (isset($output["id"])) : ?>  
        <a class="btn btn-secondary" href="php echo get_url("viewLocations.php?id=" . $output["id"]); ?&gt;">View</a>  
    </?php endif; ?>
```

Code within the tourist card layout file responsible for adding a single view button to each associated user entity (location).

Checklist Items (1)

#3 Each record should have a button/link for single view

```
<?php if (isset($output["user_id"]) && isset($output["places_id"])) : ?>
    <a class="btn btn-danger" href="php echo get_url("delete_Association.php?user_id=" . $output["user_id"] . "&amp;places_id=" . $output["places_id"]); ?&gt;"Remove</a>
<?php endif; ?>
```

Code within the tourist card layout file responsible for adding a association removal button to each record/entity/location.

Checklist Items (1)

#4 Each record should have a button/link for delete (this may be an admin-only thing but should be present for the specific role) Note: this is to delete the relationship and not the specific entity

```
<?php
require(__DIR__ . "/../../partials/nav.php");
$db = getDB();
//remove all associations
```

```

if (isset($_GET["remove"])) {
    $query = "DELETE FROM `UserLocations` WHERE user_id = :user_id";
    try {
        $stmt = $db->prepare($query);
        $stmt->execute([":user_id" => get_user_id()]);
        flash("Successfully removed all brokers", "success");
    } catch (PDOException $e) {
        error_log("Error removing broker associations: " . var_export($e, true));
        flash("Error removing broker associations", "danger");
    }
}

redirect("my_locations.php");
} <- #5-17 if (isset($_GET["remove"]))

```

The code that is handled on the logged in user's associations page itself responsible for adding a remove all associations button at the top of the page to delete all associations the current logged in user has with different locations.

Checklist Items (1)

#5 Show the logic for deleting all associations for the user (this may be admin-only but should be present for the specific role)

```

<?php
if (!isset($result_count)) {
    $message = "Dev Note: result_count not set";
    error_log($message);
    flash($message);
    $result_count = -1;
} <- #2-7 if (!isset($result_count))
if (!isset($total_count)) {
    $message = "Dev Note: total_count not set"; //UCID: LM457
    error_log($message); //DATE: 4/29/2024
    flash($message);
    $total_records = -1;
} <- #8-13 if (!isset($total_count))
?>
<header>
    <div class="row">
        <div class="col-1">Results: <?php se("$result_count/$total_count"); ?></div>
    </div>
</header>| You, 2 days ago * added count total location functionality and st...

```

The code handled on a separate page and referenced as the function render_result_counts that is responsible for outputting the number of entities/locations associated with the logged in user, as well as the current number of entities on display on the screen in the event any filters are applied.

Checklist Items (2)

#6 Show the logic related to the count of all associated items to the user (even the ones not shown in the filtered results)

#7 Show the logic related to the count of the items on the page (this value should change based on the filter applied)

```
if ($count[$_GET] > 0) {
    session_saveSessionKey($_GET);
    $keys = array_keys($_GET);
}

foreach ($keys as $k => $v) {
    if (!in_array($v, "user", $keys)) {
        $keys[$k] = "value" + $_GET[$v];
        $keys[$v] = "user";
    }
} // end of foreach ($keys as $k => $v)

// National ID
if (empty($_GET["NationalID"])) {
    $query = " AND NationalID LIKE '%NationalID'";
    $params["NationalID"] = "%" . $_GET["NationalID"] . "%";
} // end of if

// Name
if (empty($_GET["name"])) {
    $query = " AND name LIKE '%name'";
    $params["name"] = "%" . $_GET["name"] . "%";
} // end of if

// Rating
if (empty($_GET["min_rating"])) {
    $query = " AND rating > min_rating";
    $params["min_rating"] = $_GET["min_rating"];
} // end of if

// Number of reviews
if (empty($_GET["min_num_reviews"])) {
    $query = " AND num_reviews > min_num_reviews";
    $params["min_num_reviews"] = $_GET["min_num_reviews"];
} // end of if

// Date range
if (empty($_GET["date_min"])) {
    $query = " AND created > date_min";
    $params["date_min"] = $_GET["date_min"];
} // end of if

if (empty($_GET["date_max"])) {
    $query = " AND created <= date_max";
    $params["date_max"] = $_GET["date_max"];
} // end of if

// Sort and order
$sort = $_GET["sort"];
if (!in_array($sort, ["rating", "num_reviews"])) {
    $sort = "created";
}
if ($sort == "created" || $sort == "modified") {
    $sort = "+$sort";
} else {
    $sort = "-$sort";
}

$border = val($_GET, "border", "desc", false);
if (!in_array($border, ["asc", "desc"])) {
    $border = "desc";
}

// IMPORTANT note: make sure you fully validate/trust $sort and $border (sql injection possibility)
$query .= " ORDER BY $sort $border";
// limit
try {
    $client = (int)val($_GET, "limit", "10", false);
} catch (Exception $e) {}
$client = 10;
if ($client < 1 || $client > 100) {
    $client = 10;
} // IMPORTANT note: make sure you fully validate/trust $client (sql injection possibility)
$query .= " LIMIT $client";
// end of if (val($_GET, "limit") < 0)
```

All of the code responsible for handling all of the different filtering fields and sorting by either rating, reviews or date in ascending/descending order. Also, the code also handles the LIMIT functionality where the number of records on display is constrained to 1-100, but defaults to a value of 10.

Checklist Items (1)

#8 Show the logic related to filter/sort (limit should be constrained to 1-100 otherwise default to 10)

Task #3 - Points: 1

Text: Explain the solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Mention how you determine the result list (include the association logic and filters)
<input checked="" type="checkbox"/> #2	1	Mention the logic behind the two counts (total items and visible result count)
<input checked="" type="checkbox"/> #3	1	Mention the logic for single delete and delete all associations for the specific user

Response:

The result is determined by the query I use to fetch only those entities/locations associated with the current logged in

user. The query joins my tourist_info table with my relationship table UserLocations on the place_id and left joins my user table on user_id. Additionally, the current logged in user's id is retrieved and used in a WHERE clause to retrieve only associated entities. Further clauses are appended to the query based on whether or not parameters are passed through the filter/sort code below and if a value other than the default value is specified for the LIMIT clause. The total records is a separate query that is passed to obtain the count of all associated entities and the visible results are simply a count of the results array that we initially obtained, which changes based on filters. To handle the single delete functionality, the results array is parsed in a loop for each record/entity where each record is checked to see if it's user_id and places_id values are populated to ensure the existence of a relationship between them, and if so, a button to remove/delete the association is presented to the user. The button links to a separate file responsible for handling the single removal by obtaining the user_id and places_id using a \$_GET request and appending them as part of the DELETE SQL statement to remove the associated entry in the UserLocations relationship table. The bulk delete to remove all associations between the logged in user and their entities is done on the page itself where a button is presented at the top of the screen alongside the filter button to disassociate all entities with the logged in user. It essentially append the string '?remove' to the URL which is retrieved via a \$_GET request and then a DELETE SQL statement is executed on the UserLocations table to remove all associations with the logged in user's user_id.

Task #4 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Include the heroku prod link for the page that creates the association
<input checked="" type="checkbox"/> #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

https://it202-lm457-prod-cf97d0c34365.herokuapp.com/project/my_locations.php

URL #2

<https://github.com/Kith07/spring2024-IT202/pull/76>

URL #3

<https://github.com/Kith07/spring2024-IT202/pull/77>

All Users Association Page (likely an admin page) (2 pts.)

Task #1 - Points: 1

Text: Screenshots of this page

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Show the summary of the results with relevant information per entity
<input checked="" type="checkbox"/> #2	1	Show the single view buttons/links, delete button/links

#3	1	Show the username related to the specific entity and that it's clickable
#4	1	Show variations of the number of shown items count and show the count of total number of associated items
#5	1	Show variations of the filter/sort including no results (proper message should be visible)
#6	1	Make sure the heroku dev url is visible in the address bar
#7	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

The screenshot shows a web application interface for managing travel locations. At the top, there's a navigation bar with links like Home, Profile, Locations, Available Locations, My Travel Bucket List, Admin, Tourist Locations, and Logout. Below the navigation is a search/filter section with fields for Username, ID Search, Name Search, Min Rating, Min Reviews, Minimum Date, Maximum Date, Sort, Order, and Limit. A 'Filter' and 'Clear' button are also present. The main content area displays five travel locations in a grid:

- Fremont Street Experience**: National ID: 102514, Name: Fremont Street Experience, Rating: 4.0, Number of Reviews: 45217. Buttons: View, sai's Profile, Remove.
- The LINQ Promenade**: National ID: 9937981, Name: The LINQ Promenade, Rating: 4.5, Number of Reviews: 3047. Buttons: View, sats Profile, Remove.
- KAA by Cirque du Soleil**: National ID: 501183, Name: KAA by Cirque du Soleil, Rating: 4.5, Number of Reviews: 8819. Buttons: View, abc's Profile, Remove.
- Mount Charleston**: National ID: 135685, Name: Mount Charleston, Rating: 4.5, Number of Reviews: 1490. Buttons: View, abc's Profile, Remove.
- Red Rock Canyon National Conservation Area**: National ID: 103329, Name: Red Rock Canyon National Conservation Area, Rating: 4.5, Number of Reviews: 25441. Buttons: View, sesha's Profile, Remove.

My all associations page with all records in the relationship database table associated with each user upon page load.

Checklist Items (1)

#1 Show the summary of the results with relevant information per entity

This screenshot shows a simplified view of the travel location details compared to the previous one. It displays a single travel location card with the following information:

- Fremont Street Experience**: National ID: 102514. Buttons: View, sai's Profile, Remove.

Name:	Fremont Street Experience
Rating:	4.0
Number of Reviews:	45217
Website:	http://vegasexperience.com/viva-vision-light-show
Address:	425 Fremont Street, Las Vegas, NV 89101-5620
Phone:	+1 702-678-5600
View	
sai's Profile	
Remove	
Output not available	

Each entity/location that is displayed on screen has a single view button that links to a more detailed view of the entity as well as a remove button that disassociates the relationship between the given entity and its associated user.

Checklist Items (1)

#2 Show the single view buttons/links, delete button/links

Selected By: sesha
FAvorited By: 1 Users
Red Rock Canyon National Conservation Area
National ID: 103329
Name: Red Rock Canyon National Conservation Area
Rating: 4.5
Number of Reviews: 26441
Website: http://www.redrockcanyonlv.org/
Address: 1000 Scenic Loop Dr, Las Vegas, NV 89161
Phone: +1 702-515-5350
View
sesha's Profile
Remove
Add to Travel Bucket List



Each entity on the page has a heading at the top indicating who it is favorited/selected by to be in their bucket list and also has a link to that user's public profile as a blue button towards the bottom.

Checklist Items (1)

#3 Show the username related to the specific entity and that it's clickable



https://it202-lm457-dev-7b44497d46f0.herokuapp.com/project/admin/location_Associations.php

Tourist Attractions Home Profile Locations Available Locations My Travel Bucket List Admin Tourist Locations Logout

Favorited Travel Locations (Associated)

Username	ID Search	Name Search	Min Rating	Min Reviews	Minimum Date	Maximum Date	Sort	Order	Limit
Username Search	ID Search	Name Search	Minimum Rating	Minimum Reviews	mm/dd/yyyy	mm/dd/yyyy	Rating	+/-	10

Filter **Clear**

Results: 0/4

Selected By: sal

Favorited By: 1 Users

Fremont Street Experience

National ID: 102514
Name: Fremont Street Experience
Rating: 4.0
Number of Reviews: 45217
Website: <http://vegasexperience.com/viva-vision-light-show>
Address: 425 Fremont Street, Las Vegas, NV 89101-5620
Phone: +1 702-676-5600

View
sal's Profile
Remove
Output not available

Selected By: sal

Favorited By: 1 Users

The LINQ Promenade

National ID: 6637961
Name: The LINQ Promenade
Rating: 4.5
Number of Reviews: 3547
Website: <https://www.caesars.com/linq>
Address: 3345 Las Vegas Blvd S between The LINQ Hotel & Casino and Flamingo Las Vegas, Las Vegas, NV 89108-8521
Phone: +1 702-322-0543

View
sal's Profile
Remove
Output not available

Selected By: abc

Favorited By: 1 Users

KAA by Cirque du Soleil

National ID: 501163
Name: KAA by Cirque du Soleil
Rating: 4.5
Number of Reviews: 8619
Website: <http://www.mgmgrand.converter/ka-mgm-cirque-du-soleil-show.aspx>
Address: 3799 Las Vegas Blvd S MGM Grand, Las Vegas, NV 89108-4219
Phone: +1 702-531-3826

View
abc's Profile
Remove
Add to Travel Bucket List

Selected By: sesha

Favorited By: 1 Users

Red Rock Canyon National Conservation Area

National ID: 103329
Name: Red Rock Canyon National Conservation Area
Rating: 4.5
Number of Reviews: 25441
Website: <http://www.redrockcanyonnv.org/>
Address: 1000 Scenic Loop Dr, Las Vegas, NV 89161
Phone: +1 702-615-6360

View
sesha's Profile
Remove
Add to Travel Bucket List

(Part 1) Currently, the page has a total number of 4 associated records on the page, with all 4 of them on display as well.

Checklist Items (1)

#4 Show variations of the number of shown items count and show the count of total number of associated items

https://it202-lm457-dev-7b44497d46f0.herokuapp.com/project/admin/location_Associations.php?username=sesha&NationalID=&name=&min_rating=&min_num_reviews=&date_min=&...

Tourist Attractions Home Profile Locations Available Locations My Travel Bucket List Admin Tourist Locations Logout

Favorited Travel Locations (Associated)

Username	ID Search	Name Search	Min Rating	Min Reviews	Minimum Date	Maximum Date	Sort	Order	Limit
sesha	ID Search	Name Search	Minimum Rating	Minimum Reviews	mm/dd/yyyy	mm/dd/yyyy	Rating	+/-	10

Filter **Clear** **Remove All Locations**

Results: 1/4

Selected By: sesha

Favorited By: 1 Users

Red Rock Canyon National Conservation Area

National ID: 103329
Name: Red Rock Canyon National Conservation Area
Rating: 4.5
Number of Reviews: 25441
Website: <http://www.redrockcanyonnv.org/>
Address: 1000 Scenic Loop Dr, Las Vegas, NV 89161
Phone: +1 702-615-6360

View
sesha's Profile
Remove
Add to Travel Bucket List

If I do a username filter with 'sesha', there is only one associated entity/location. Therefore, the results on page is modified to 1/4, to represent the fact that only one entity is on display currently, despite there being a total of 4 total

Checklist Items (1)

#4 Show variations of the number of shown items count and show the count of total number of associated items

The screenshot shows a web application for managing travel locations. At the top, there's a navigation bar with tabs for 'Tourist Attractions', 'Home', 'Profile', 'Locations', 'Available Locations', 'My Travel Bucket List', 'Admin', 'Travel Locations', and 'Logout'. Below the navigation is a search bar with fields for 'Username', 'ID Search', 'Name Search', 'Min Rating', 'Min Reviews', 'Minimum Date', 'Maximum Date', 'Sort', 'Order', and 'Limit'. There are also 'Filter', 'Clear', and 'Remove All Locations' buttons. The main content area displays 'Favorited Travel Locations (Associated)' with a result count of 2/4. Two locations are listed: 'Fremont Street Experience' and 'Red Rock Canyon National Conservation Area'. Each location card contains detailed information and actions like 'View', 'Profile', 'Remove', and 'Add to Travel Bucket List'.

Another page that showcases how the count of the number of shown items has changed to 2/4 now due to the minimum reviews filter being applied. A minimum of 25000 reviews results in only two of the associated entries being displayed on screen.

Checklist Items (1)

#5 Show variations of the filter/sort including no results (proper message should be visible)

This screenshot shows the same application interface as the previous one, but with a different filter applied. The 'Min Reviews' field is set to 25000, which results in no results being displayed. The 'Results: 0/4' message is clearly visible. The rest of the interface, including the search bar and location cards, remains the same.

Address: 428 Fremont Street, Las Vegas, NV 89101-0620 Phone: +1 702-578-0600	Address: 1000 Bonnefond Loop Dr. Las Vegas, NV 89161 Phone: +1 702-515-5350	Address: 3999 Las Vegas Blvd S MGM Grand, Las Vegas, NV 89109-4319 Phone: +1 702-531-3526	Address: 3999 Las Vegas Blvd S Caesars Palace, Las Vegas, NV 89109-6921 Phone: +1 702-322-0543
View sara's Profile Remove Output not available	View masha's Profile Remove Add to Travel Bucket List	View alex's Profile Remove Add to Travel Bucket List	View sara's Profile Remove Output not available

Another variation demonstrating the sort functionality of the page where all associated entities are now formatted in descending order based on the number of reviews. Essentially, the one with the most reviews is at the left and the one with the least is at the right.

Checklist Items (1)

#5 Show variations of the filter/sort including no results (proper message should be visible)

Task #2 - Points: 1

Text: Screenshot the code

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Show the code related to fetching all associations (including the query)
<input checked="" type="checkbox"/> #2	1	Show the code related to the display of the results
<input checked="" type="checkbox"/> #3	1	Each record should have a button/link for single view
<input checked="" type="checkbox"/> #4	1	Each record should have a username field that is clickable to go to the user's profile page
<input checked="" type="checkbox"/> #5	1	Each record should have a button/link for delete (this may be an admin-only thing but should be present for the specific role) Note: this is to delete the relationship and not the specific entity
<input checked="" type="checkbox"/> #6	1	Show the logic related to the count of all associated items (even the ones not shown in the filtered results)
<input checked="" type="checkbox"/> #7	1	Show the logic related to the count of the items on the page (this value should change based on the filter applied)
<input checked="" type="checkbox"/> #8	1	Show the logic related to filter/sort (should include a partial match for username) (limit should be constrained to 1-100 otherwise default to 10)
<input checked="" type="checkbox"/> #9	1	Include ucid/date comments for each code screenshot
<input checked="" type="checkbox"/> #10	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small

Medium

Large



The code and query responsible for fetching all of the associated entities/locations from the relationship database table.

Checklist Items (1)

#1 Show the code related to fetching all associations (including the query)

```
Stable = [{"data" => $results, "title" => "List of Tourist Locations Data", "ignored_columns" => [
    "id", "location_id", "language", "currency", "description", "write_review", "monday_open", "monday_close", "tuesday_open", "tuesday_close", "wednesday_open", "wednesday_close",
    "thursday_open", "thursday_close", "friday_open", "friday_close", "saturday_open", "saturday_close", "sunday_open", "sunday_close", "popular_tour_title", "primary_category", "price", "partner", "tour_url", "product_code", "is_api"}];
}



### Favorites Travel Locations (Associated)



All
Popular
New



| Location Name      | Category   | Price Range  | Actions                                     |
|--------------------|------------|--------------|---------------------------------------------|
| Golden Gate Bridge | Landmark   | \$10 - \$50  | <a href="#">Edit</a> <a href="#">Delete</a> |
| Alcatraz Island    | Attraction | \$20 - \$100 | <a href="#">Edit</a> <a href="#">Delete</a> |
| Golden Gate Bridge | Landmark   | \$10 - \$50  | <a href="#">Edit</a> <a href="#">Delete</a> |
| Alcatraz Island    | Attraction | \$20 - \$100 | <a href="#">Edit</a> <a href="#">Delete</a> |



Create New Location



Logout


```

The code responsible for displaying all of the associated entities/locations on the page using two previously mentioned helper functions, `render_result_counts` and `render_tourist_card` that each respectively display the total number of

Helper functions, render_entities and render_tourist_card that each respectively display the total number of entities on display and associated and the layout in which each entity/location is formatted on the page.

Checklist Items (1)

#2 Show the code related to the display of the results

```
<?php if (isset($output["id"])) : ?>
    <a class="btn btn-secondary" href=<?php echo get_url("viewLocations.php?id=" . $output["id"]); ?>>View</a>
<?php endif; ?>
```

The tourist card layout file code responsible for handling the addition of a single view button to each associated entity/location.

Checklist Items (1)

#3 Each record should have a button/link for single view

```
<?php if (isset($output["user_id"])) : ?>
    <a class="btn btn-primary" href=<?php echo get_url("profile.php?id=" . $output["user_id"]); ?>><?php se($output, "username"); ?>'s Profile</a>
<?php endif; ?>
```

The tourist card layout file code that is also responsible for handling the addition of a public profile link to each associated entity so that users/admins can navigate to the user associated with each entity/location

Checklist Items (1)

#4 Each record should have a username field that is clickable to go to the user's profile page

```
<?php if (isset($output["user_id"]) && isset($output["places_id"])) : ?>
  <a class="btn btn-danger" href="<?php echo get_url("delete_Association.php?user_id=" . $output["user_id"] . "&places_id=" . $output["places_id"]); ?>">Remove</a>
<?php endif; ?>
```

The tourist card layout file code responsible for handling the inclusion of a remove/delete button that will disassociate a user from a given entity/location.

Checklist Items (1)

#5 Each record should have a button/link for delete (this may be an admin-only thing but should be present for the specific role) Note: this is to delete the relationship and not the specific entity

```
<?php if (isset($_GET["username"])) : ?>
    <a class="btn btn-danger" href=<?php echo get_url("admin/removeAllAssociations.php?username=" . $_GET["username"]); ?>>Remove All Locations</a>
<?php endif; ?>
```

The code on the all associations page responsible for adding a button when a username query is entered to disassociate all of the matching entities/locations from their users.

Checklist Items (1)

#5 Each record should have a button/link for delete (this may be an admin-only thing but should be present for the specific role) Note: this is to delete the relationship and not the specific entity

```
<?php
require_once(__DIR__ . "/../../../../partials/nav.php");

if (!has_role("Admin")) {
    flash("You don't have permission to perform this action", "warning");
    redirect("home.php");
}

if (isset($_GET["username"])) {
    $username = $_GET["username"]; //UCID: LM457
    $partialName = "%$username%"; //DATE: 4/29/2024 You, now * Uncommitted changes

    $db = getDB();
    $stmt = $db->prepare("DELETE FROM UserLocations WHERE user_id IN (SELECT id FROM Users WHERE username LIKE :username)");

    try {
        $stmt->execute([":username" => $partialName]);
        flash("Associations removed successfully", "success");
    } catch (PDOException $e) {
        flash("Error removing associations: " . $e->getMessage(), "danger");
    }
} else {
    flash("Missing username flag", "danger");
}
redirect("admin/location_Associations.php");

?>
<?php require(__DIR__ . "/../../../../partials/flash.php");?>
```

The separate file that the remove all associations button links to that handles the removal of all partial matching username filter entities/locations using the SQL LIKE clause.

Checklist Items (1)

#5 Each record should have a button/link for delete (this may be an admin-only thing but should be present for the specific role) Note: this is to delete the relationship and not the specific entity

```
$total_records = get_total_count("tourist_info t JOIN `UserLocations` ut ON t.id = ut.places_id");
```

The query that is passed to obtain the number of total associated records.

Checklist Items (2)

#6 Show the logic related to the count of all associated items (even the ones not shown in the filtered results)

#7 Show the logic related to the count of the items on the page (this value should change based on the filter applied)

```
You, 5 minutes ago | + author (YOU) | Delete
<?php
if (!isset($result_count)) {
    $message = "Dev Note: result_count not set";
    error_log($message);
    flash($message);
    $result_count = -1;
} <- #2-7 if (!isset($result_count))
if (!isset($total_count)) { //UCID: LM457
    $message = "Dev Note: total_count not set"; //DATE: 4/29/2024
    error_log($message);
    flash($message);
    $total_records = -1;
} <- #8-13 if (!isset($total_count))
?>
<header>
    <div class="row">
        <div class="col-1">Results: <?php se("$result_count/$total_count"); ?></div>
    </div>
</header>| You, 2 days ago + added count total location functionality and st...
```

The code responsible for handling both the count of the number of all associated entities as well as the count of the number of entities/locations on display. (Result count is passed as the count of the results array from our all associations page).

Checklist Items (2)

#6 Show the logic related to the count of all associated items (even the ones not shown in the filtered results)

#7 Show the logic related to the count of the items on the page (this value should change based on the filter applied)

The code/logic responsible for handling the filtering and sorting functionality of the all associations page. It also handles the LIMIT functionality that has values constrained to 1-100 with a default value of 10.

Checklist Items (1)

#8 Show the logic related to filter/sort (should include a partial match for username) (limit should be constrained to 1-100 otherwise default to 10)

Task #3 - Points: 1

Text: Explain the solution

Checklist

***The checkboxes are for your own tracking**

<input checked="" type="checkbox"/> #2	1	Mention the logic behind the two counts (total items and visible result count)
<input checked="" type="checkbox"/> #3	1	Mention the logic for single delete
<input checked="" type="checkbox"/> #4	1	Mention the logic for handling the username requirements

Response:

Once again, the result list/array is determined by the query I used to fetch only those entities/locations that have an association with any user in the UserLocations table. Essentially the query dumps all locations with a places_id entry in its table. The query joins my tourist_info table with my relationship table UserLocations on the places_id and joins my Users table on user_id. Additionally, an IF clause is used to count the number of user_ids associated with each entity/location that is found after joining the UserLocations table on the tourist_info table based on the places_id. Further clauses are appended to the query based on whether or not parameters are passed through the filter/sort code below and if a value other than the default value is specified for the LIMIT clause.

The total records is a separate query that is passed to obtain the count of all associated entities. The query joins the tourist_info table on the relationship table UserLocations once again on places_id and is appended to the query used by the function get_total_count which counts the number of the parameters passed, which in this case are all of the associated entities/locations. The visible results are simply a count of the results array that we initially obtained, which changes based on the application of filters. Essentially it counts the number of entities fetched from our results query.

To handle the single delete functionality, the results array is parsed in a loop for each record/entity where each record is checked to see if its user_id and places_id values are populated to ensure the existence of a relationship between them, and if so, a button to remove/delete the association is presented to the user. The button links to a separate file responsible for handling the single removal by obtaining the user_id and places_id using a \$_GET request and appending them as part of the DELETE SQL statement to remove the associated entry in the UserLocations relationship table.

The logic to print which entity has been selected/associated by which user and the link to that user's public profile page is handled on my tourist card layout file which is what receives my final results array with the corresponding records/entities/locations. The code first checks to see if each record has a username field that is populated after previously joining the Users table with the tourist_info table on user_id, and since only records with associations are passed in the first place, every record is bound to have the field populated with some value. As such, the username is then displayed on screen on each individual record preceded by a "Selected By:" statement to demonstrate who the location was chosen by. Several users can choose the same location to add to their bucket list, in which case, duplicate entries of the same location are displayed with each entity being associated with the respective user. As for the link to the user's public profile, the conditional check now ensures the user_id field is populated with a value for each record, and if so, creates a button that links to the public profile page of the associated user.

Task #4 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Include the heroku prod link for the page that creates the association

#2

1

Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

https://it202-lm457-prod-cf97d0c34365.herokuapp.com/project/admin/location_Associations.php

URL #2

<https://github.com/Kith07/spring2024-IT202/pull/77>

URL #3

<https://github.com/Kith07/spring2024-IT202/pull/78>

URL #4

<https://github.com/Kith07/spring2024-IT202/pull/80>

URL #5

<https://github.com/Kith07/spring2024-IT202/pull/83>

Unassociated Page (2 pts.)

[^COLLAPSE ^](#)

Task #1 - Points: 1

Text: Screenshots of this page

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Show the summary of the results with relevant information per entity
<input checked="" type="checkbox"/> #2	1	Show the single view buttons/links
<input checked="" type="checkbox"/> #3	1	Show variations of the number of shown items count and show the count of total number of associated items
<input checked="" type="checkbox"/> #4	1	Show variations of the filter/sort including no results (proper message should be visible)
<input checked="" type="checkbox"/> #5	1	Make sure the heroku dev url is visible in the address bar
<input checked="" type="checkbox"/> #6	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

The screenshot shows a web browser window with the following details:

- Address Bar:** https://it202-lm457-dev-7b44497d46f0.herokuapp.com/project/available_locations.php
- Navigation Bar:** Tourist Attractions, Home, Profile, Locations, Available Locations, My Travel Bucket List, Admin, Tourist Locations, Logout.
- Section Header:** Available Tourist Locations (Unassociated)
- Search and Filter:** ID Search, Name Search, Minimum Rating, Minimum Reviews, Minimum Date, Maximum Date, Sort, Order, Limit (set to 10).
- Buttons:** Filter, Clear.
- Results:** Results: 55
- Table Data:**

ID	Name	Rating	Reviews	Date	Details
Shelby American, Inc.	Seventy Seven Acres	Mount Charleston	High Roller	Lion Habitat Ranch	National ID: 2409034 National ID: 7777777 National ID: 135565 National ID: 0509306 National ID: 3619966

Name: Shady American, Inc.	Name: Seventy Seven Acres	Name: Mount Charleston	Name: High Roller	Name: Lion Habitat Ranch
Rating: 4.5	Rating: 4.9	Rating: 4.5	Rating: 4.5	Rating: 5.0
Number of Reviews: 1547	Number of Reviews: 7777	Number of Reviews: 1490	Number of Reviews: 25877	Number of Reviews: 1639
Website: http://ShadyAmerican.com/	Website: www.777acres.com	Website: http://www.gomtcharleston.com	Website: http://www.caesars.com/linq/high-roller	Website: http://www.lionhabitatranch.com/
Address: 6405 Ensworth St, Las Vegas, NV 89119-3211	Address: 777 Seventy Lane, Sacramento, CA	Address: Robbers Roost Trail, Las Vegas, NV 89109	Address: 3545 Las Vegas Blvd South The Linq Promenade, Las Vegas, NV 89109-8978	Address: 382 Bruner Ave, Henderson, NV 89044-0748
Phone: +1 702-942-7325	Phone: +1 777-777-7777	Phone: +1 702-672-5486	Phone: +1 702-322-0593	Phone: +1 702-595-6665
View	View	View	View	View
Add to Travel Bucket List	Add to Travel Bucket List	Add to Travel Bucket List	Add to Travel Bucket List	Add to Travel Bucket List

The unassociated available entities/locations page upon page load with all relevant information per location. (Locations that have not been added to a bucket list by users)

Checklist Items (1)

#1 Show the summary of the results with relevant information per entity

Shelby American, Inc.

National ID: 2409034

Name: Shelby American, Inc.

Rating: 4.5

Number of Reviews: 1547

Website:
<http://shelbyamerican.com/>

Address: 6405 Ensworth St, Las Vegas, NV 89119-3211

Phone: +1 702-942-7325

View

Add to Travel Bucket List

Each entity/location has a single view button attached towards the bottom that leads to a more detailed page of the specific location.

Checklist Items (1)

#2 Show the single view buttons/links

A screenshot of a web browser window. The address bar shows a URL for a project on herokuapp.com. The main content area displays a table with several rows of data, likely representing location information. The browser interface includes standard navigation buttons (back, forward, search) and a toolbar with various icons.

Available Tourist Locations (Unassociated)

ID Search Name Search Minimum Rating Minimum Reviews Minimum Date Maximum Date Sort Order Limit

ID Search S Minimum Rating Minimum Review mm/dd/yyyy mm/dd/yyyy Rating + - 10

Filter **Clear**

Results: 3/5

Shelby American, Inc. National ID: 2400934 Name: Shelby American, Inc. Rating: 4.5 Number of Reviews: 1547 Website: http://shelbyamerican.com/ Address: 8405 Emsworth St, Las Vegas, NV 89119-3211 Phone: +1 702-942-7325 View Add to Travel Bucket List	Mount Charleston National ID: 135565 Name: Mount Charleston Rating: 4.5 Number of Reviews: 1490 Website: http://www.gomtcharleston.com Address: Robbers Roost Trail, Las Vegas, NV 89109 Phone: +1 702-872-5486 View Add to Travel Bucket List	Seventy Seven Acres National ID: 7777777 Name: Seventy Seven Acres Rating: 4.9 Number of Reviews: 7777 Website: www.777acres.com Address: 777 Seventy Lane, Sacramento, CA Phone: +1 777-777-7777 View Add to Travel Bucket List
---	--	--

The page after filtering the results by the location name with a partial search on the letter 's'. The results indicate that only 3 entities/locations are currently on display out of a total of 5 unassociated entities/locations.

Checklist Items (2)

#3 Show variations of the number of shown items count and show the count of total number of associated items

#4 Show variations of the filter/sort including no results (proper message should be visible)

ID Search Name Search Minimum Rating Minimum Reviews Minimum Date Maximum Date Sort Order Limit

ID Search Name Search Minimum Rating Minimum Review mm/dd/yyyy mm/dd/yyyy Rating + - 10

Filter **Clear**

Results: 4/4

Shelby American, Inc. National ID: 2400934 Name: Shelby American, Inc. Rating: 4.5 Number of Reviews: 1547 Website: http://shelbyamerican.com/ Address: 8405 Emsworth St, Las Vegas, NV 89119-3211 Phone: +1 702-942-7325 View Add to Travel Bucket List	Mount Charleston National ID: 135565 Name: Mount Charleston Rating: 4.5 Number of Reviews: 1490 Website: http://www.gomtcharleston.com Address: Robbers Roost Trail, Las Vegas, NV 89109 Phone: +1 702-872-5486 View Add to Travel Bucket List	High Roller National ID: 6509306 Name: High Roller Rating: 4.5 Number of Reviews: 25877 Website: http://www.caesars.com/lng/high-roller Address: 3545 Las Vegas Blvd South The Linq Promenade, Las Vegas, NV 89109-8978 Phone: +1 702-322-0593 View Add to Travel Bucket List	Lion Habitat Ranch National ID: 3810966 Name: Lion Habitat Ranch Rating: 5.0 Number of Reviews: 1630 Website: http://www.lionhabitatranch.com/ Address: 382 Bruner Ave, Henderson, NV 89044-8746 Phone: +1 702-595-6666 View Add to Travel Bucket List
---	--	--	---

The page after favoriting one of the previously available entities so that the total number of unassociated entities decreases to a count of 4 and so does the number of entities on display.

Checklist Items (1)

#3 Show variations of the number of shown items count and show the count of total number of associated items

Tourist Attractions Home Profile Locations Available Locations My Travel Bucket List Admin Tourist Locations Logout

Available Tourist Locations (Unassociated)

ID Search	Name Search	Minimum Rating	Minimum Reviews	Minimum Date	Maximum Date	Sort	Order	Limit
1	Name Search	Minimum Rating	Minimum Review:	mm/dd/yyyy	mm/dd/yyyy	Rating	+	10

Results: 2/4

Mount Charleston
National ID: 135565
Name: Mount Charleston
Rating: 4.5
Number of Reviews: 1490
Website: <http://www.gomtcharleston.com>
Address: Robbers Roost Trail, Las Vegas, NV 89109
Phone: +1 702-872-5486

Lion Habitat Ranch
National ID: 3819966
Name: Lion Habitat Ranch
Rating: 5.0
Number of Reviews: 1639
Website: <http://www.lionhabitatranch.com/>
Address: 382 Bruner Ave, Henderson, NV 89044-8746
Phone: +1 702-595-6666

View **Add to Travel Bucket List** **View** **Add to Travel Bucket List**

Another variation of filtering that shows how a partial search on location_id using the number 1 results in only 2 of the 4 available unassociated entities on display since only those contain a 1 in their location id.

Checklist Items (3)

#3 Show variations of the number of shown items count and show the count of total number of associated items

#4 Show variations of the filter/sort including no results (proper message should be visible)

#5 Make sure the heroku dev url is visible in the address bar

Tourist Attractions Home Profile Locations Available Locations My Travel Bucket List Admin Tourist Locations Logout

Available Tourist Locations (Unassociated)

ID Search	Name Search	Minimum Rating	Minimum Reviews	Minimum Date	Maximum Date	Sort	Order	Limit
ID Search	Name Search	Minimum Rating	Minimum Review:	mm/dd/yyyy	mm/dd/yyyy	Date	+	10

Results: 4/4

Lion Habitat Ranch
National ID: 3819966
High Roller
National ID: 6509306
Mount Charleston
National ID: 135565
Shelby American, Inc.
National ID: 2409034

View **Add to Travel Bucket List** **View** **Add to Travel Bucket List** **View** **Add to Travel Bucket List** **View** **Add to Travel Bucket List**

Name: Lion Habitat Ranch
Rating: 5.0
Number of Reviews: 1639
Website: http://www.lionhabitatranch.com/
Address: 382 Bruner Ave, Henderson, NV 89044-8746
Phone: +1 702-595-6666
View
Add to Travel Bucket List

Name: High Roller
Rating: 4.5
Number of Reviews: 25877
Website: http://www.caesars.com/linq/high-roller
Address: 3545 Las Vegas Blvd South The Linq Promenade, Las Vegas, NV 89109-8978
Phone: +1 702-322-0593
View
Add to Travel Bucket List

Name: Mount Charleston
Rating: 4.5
Number of Reviews: 1490
Website: http://www.gomtcharleston.com
Address: Robbers Roost Trail, Las Vegas, NV 89109
Phone: +1 702-672-5486
View
Add to Travel Bucket List

Name: Shelby American, Inc.
Rating: 4.5
Number of Reviews: 1547
Website: http://shelbyamerican.com/
Address: 6405 Ensworth St, Las Vegas, NV 89119-3211
Phone: +1 702-942-7325
View
Add to Travel Bucket List

The sort functionality on display that formats the available entities/locations by their creation date in descending order, meaning the latest to the oldest created locations in our database.

Checklist Items (1)

#5 Make sure the heroku dev url is visible in the address bar

Another variation of the filter functionality, this time with the minimum date filter which I set to tomorrow, so that no entries pop up on display, thus leading to the no results to show message.

Checklist Items (1)

#4 Show variations of the filter/sort including no results (proper message should be visible)

Task #2 - Points: 1

COLLAPSE ▾

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Show the code related to fetching all unassociated entities (including the query)
<input checked="" type="checkbox"/> #2	1	Show the code related to the display of the results
<input checked="" type="checkbox"/> #3	1	Each record should have a button/link for single view
<input checked="" type="checkbox"/> #4	1	Show the logic related to the count of all unassociated items (even the ones not shown in the filtered results)
<input checked="" type="checkbox"/> #5	1	Show the logic related to the count of the items on the page (this value should change based on the filter applied)
<input checked="" type="checkbox"/> #6	1	Show the logic related to filter/sort (should include a partial match for username) (limit should be constrained to 1-100 otherwise default to 10)
<input checked="" type="checkbox"/> #7	1	Include ucid/date comments for each code screenshot
<input checked="" type="checkbox"/> #8	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

```

$query = "SELECT t.id, location_id, language, currency, NationalID, name, ranking, description, rating, num_reviews, website, address, phone, write_review, monday_open, monday_close, tuesday_open, tuesday_close, wednesday_open, wednesday_close, thursday_open, thursday_close, friday_open, friday_close, saturday_open, saturday_close, sunday_open, sunday_close, popular_tour_title, primary_category, price, partner, tour_url, product_code, is_api, t.created, COUNT(*) FROM userlocations WHERE places_id = t.id AND user_id = :user_id) AS is_favorite FROM tourist_info t WHERE t.id NOT IN (SELECT places_id FROM userlocations)";
$parameters = [":user_id" => get_user_id()];
$session_key = $_SERVER['SCRIPT_NAME'];
$session_clear = isset($_GET["clear"]);
if ($session_clear) {
    session_delete($session_key);
    unset($_GET["clear"]);
    redirect($session_key);
} else {
    $session_data = session_load($session_key);
}

if (count($_GET) == 0 && !isset($session_data) && count($session_data) > 0) {
    if ($session_data) {
        $_GET = $session_data;
    }
}
// 402-46 IF (count($_GET) == 0 && !isset($session_data) && count($session_data) > 0)
if (count($_GET) > 0) {
}
// 407-116 IF (count($_GET) > 0)

$db = pdoDB();
$stmt = $db->prepare($query);
$results = [];
try {
    $stmt->execute($parameters);
    $r = $stmt->fetchall();
    if ($r) {
        $results = $r;
    }
} catch (Exception $e) {
    error_log("Error Fetching locations: " . var_export($e, true));
    flash("An error occurred, please try again", "danger");
}

```

The code and query responsible for handling the fetching of all unassociated entities/locations from the relationship database table.

#1 Show the code related to fetching all unassociated entities (including the query)

The code responsible for displaying the fetched results from the query and database onto the screen using the helper function `render_tourist_card`.

Checklist Items (1)

#2 Show the code related to the display of the results

```
<?php if ($output["id"]): ?>
    <div class="card-body">
        <div class="bg-warning text-dark text-center">Output not available</div>
    </div>
<?php endif; ?>
</div>
</div>
```

The tourist card layout file that was previously called that renders the output in a neatly organized card layout.

Checklist Items (1)

#2 Show the code related to the display of the results

```
<?php if (isset($output["id"])) : ?>
    <a class="btn btn-secondary" href=<?php echo get_url("viewLocations.php?id=" . $output["id"]); ?>>View</a>
<?php endif; ?>
```

The functionality in the card layout file associated with rendering a button for the single view for each entity/location on display on the unassociated entities page.

Checklist Items (1)

#3 Each record should have a button/link for single view

```
$total_records = get_total_count("tourist_info t WHERE t.id NOT IN (SELECT places_id FROM `UserLocations` )");
```

The query responsible for obtaining the total number of records that are unassociated from users.

Checklist Items (2)

#4 Show the logic related to the count of all unassociated items (even the ones not shown in the filtered results)

#5 Show the logic related to the count of the items on the page (this value should change based on the filter applied)

```
<?php render_result_counts(count($results), $total_records); ?>
```

How the results are formatted/displayed on the unassociated page code. The count of the results array shows how the count for the number of entities on display changes based on what results are obtained after passing any queries from the filter/sort form.

Checklist Items (2)

#4 Show the logic related to the count of all unassociated items (even the ones not shown in the filtered results)

#5 Show the logic related to the count of the items on the page (this value should change based on the filter applied)

```

<?php
if (!isset($result_count)) {
    $message = "Dev Note: result_count not set";
    error_log($message);
    flash($message);
    $result_count = -1;
}

if (!isset($total_count)) {
    $message = "Dev Note: total_count not set"; //UCID: LM457
    error_log($message); //DATE: 4/29/2024
    flash($message);
    $total_records = -1;
}
?>

<header>
    <div class="row">
        <div class="col-1">Results: <?php se("$result_count/$total_count"); ?></div>
    </div>
</header> You, 2 days ago * added count total location functionality and st...

```

The render results code that is referenced previously responsible for displaying the total number of entities on display in addition to the count of all of the unassociated entities/locations.

Checklist Items (0)

```

if (empty($_GET) > 0) {
    session_start();
    $keys = array_keys($_GET);

    foreach ($keys as $k => $v) {
        if (is_array($_GET[$v])) {
            if (in_array($v, $keys)) {
                $query .= $k . "=" . $_GET[$v];
            }
        }
    }
} else {
    foreach ($keys as $k => $v) {
        if (empty($_GET[$v])) {
            if ($v == "NationalID") {
                $query .= " AND NationalID LIKE '%" . $_GET[$v] . "%'";
                $params["NationalID"] = "%" . $_GET[$v] . "%";
            }
        }
    }
}

if (empty($_GET["name"])) {
    $query .= " AND name LIKE '%name%'";
    $params["name"] = "%" . $_GET["name"] . "%";
}

//Rating
if (empty($_GET["min_rating"])) {
    $query .= " AND rating >= min_rating";
    $params["min_rating"] = $_GET["min_rating"];
}

//Number of Reviews
if (empty($_GET["min_num_reviews"])) {
    $query .= " AND num_reviews >= min_num_reviews";
    $params["min_num_reviews"] = $_GET["min_num_reviews"];
}

//Date Range
if (empty($_GET["date_min"])) {
    $query .= " AND created >= date_min";
    $params["date_min"] = $_GET["date_min"];
}

if (empty($_GET["date_max"])) {
    $query .= " AND created <= date_max";
    $params["date_max"] = $_GET["date_max"];
}

//Sort and Order
$sort = $_GET["sort"];
$sort .= ", created DESC";
if (in_array($sort, ["rating", "num_reviews"])) {
    $sort = "created";
}
if ($sort === "created" || $sort === "modified") {
    $sort = "t." . $sort;
}

$order = $_GET["order"];
$sort .= ", " . $order;
if (in_array($sort, ["t.id", "t.date"])) {
    $sort = "date";
}

//IMPORTANT make sure you fully validate/trust sort and order (sql injection possibility)
$query .= " ORDER BY $sort $order";
try {
    $stmt = $conn->prepare("SELECT * FROM locations WHERE $query");
} catch (Exception $e) {
    $stmt = null;
}
if ($stmt < 1 || $stmt > 100) {
    $stmt = null;
}

//IMPORTANT make sure you fully validate/trust limit (sql injection possibility)
$query .= " LIMIT $limit";
if ($limit > 100) {
    $limit = 100;
}

```

The code/logic on the unassociated entities page responsible for handling the functionality of filtering the locations

based on certain fields like name and id and/or sorting it by reviews, ratings or date. The code also handles the LIMIT functionality whose values are constrained to 1-100 with a default value of 10 to limit server-side troubles.

Checklist Items (1)

#6 Show the logic related to filter/sort (should include a partial match for username) (limit should be constrained to 1-100 otherwise default to 10)

Task #3 - Points: 1

Text: Explain the solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Mention how you determine the result list (include the unassociated logic and filters)
<input checked="" type="checkbox"/> #2	1	Mention the logic behind the two counts (total items and visible result count)

Response:

Once again, the result list/array is determined by the query I used to fetch only those entities/locations that have no association with any user from the tourist_info table. Essentially the query uses the WHERE clause in SQL to check for all of the places_ids within the tourist_info table that are not in the relationship table UserLocations, since only those are the ones with no associations with any user. A subquery to fetch all places_ids within the UserLocations table is used and only those places_ids in the tourist_info table that do not match these fetched results are formatted to the final results array. Further clauses like AND and ORDER BY are appended to the query based on whether or not parameters are passed through the filter/sort code below and if a value other than the default value is specified for the LIMIT clause. If any such parameters are passed, a \$_GET request is made to retrieve those values which are then appended to the query.

The total records is a separate query that is passed to obtain the count of all associated entities. This time, the query also mimics the results query WHERE it looks for places_ids within the tourist_info table that do not correspond with values found in the subquery that fetches places_ids from the UserLocations relationship table. This portion is then appended to the query used by the function get_total_count which counts the number of the parameters passed, which in this case are all of the non-associated entities/locations. The visible results continue to simply be a count of the results array that we initially obtained, which changes based on the application of filters. Essentially it counts the number of entities fetched from our results query.

Task #4 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Include the heroku prod link for the page that creates the association
<input checked="" type="checkbox"/> #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1https://it202-lm457-prod-cf97d0c34365.herokuapp.com/project/available_Locations.php**URL #2**<https://github.com/Kith07/spring2024-IT202/pull/79>

● Admin Association Management (like UserRoles) (1 pt.)

^COLLAPSE ^

● Task #1 - Points: 1

● Text: Screenshots of the page

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show the search form with valid data
<input type="checkbox"/> #2	1	Show the results of the search
<input type="checkbox"/> #3	1	Show the result of entities and users being associated and unassociated
<input type="checkbox"/> #4	1	Make sure the heroku dev url is visible in the address bar
<input type="checkbox"/> #5	1	Clearly caption screenshots

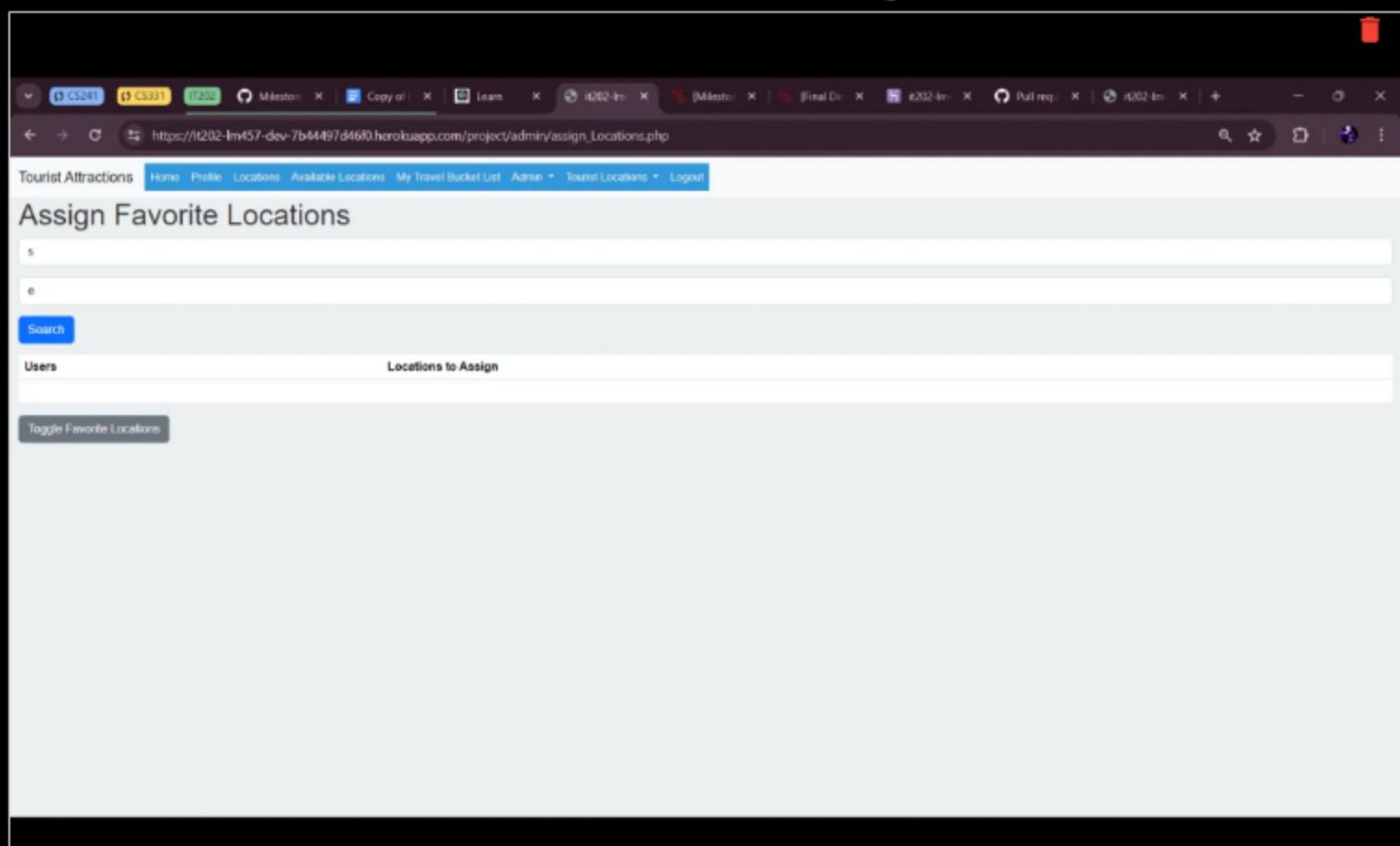
Task Screenshots:

Gallery Style: Large View

Small

Medium

Large



My assigning relationship page with valid entries in both the username and location name form fields.

Checklist Items (1)

#1 Show the search form with valid data

The screenshot shows a web browser with multiple tabs open. The active tab is titled 'Assign Favorite Locations' and has the URL https://rt202-lm457-dev-7b44497d46f0.herokuapp.com/project/admin/assign_Locations.php?username=s&places=e. The page displays a search interface with two input fields ('s' and 'e') and a 'Search' button. Below the inputs, there are two sections: 'Users' and 'Locations to Assign'. The 'Users' section lists several users with their status: Justinchen (No Favorite Locations), saevi (No Favorite Locations), ssi (Fremont Street Experience, Seventy Seven Acres, The LINQ Promenade, Ninety nine land), sample2 (No Favorite Locations), sesha (Red Rock Canyon National Conservation Area), shannoar (No Favorite Locations), test1 (No Favorite Locations), and tester2 (No Favorite Locations). The 'Locations to Assign' section lists various locations with checkboxes: Fremont Street Experience, Shelby American, Inc., Seventy Seven Acres, KAAB by Cirque du Soleil, Mount Charleston, Red Rock Canyon National Conservation Area, High Roller, The LINQ Promenade, and Ninety nine land. A 'Toggle Favorite Locations' button is located at the bottom left of the page.

The results of my entries on the previous screenshot for the username and location name. All partially matched values have been displayed on the page.

Checklist Items (0)

The screenshot shows a web browser with multiple tabs open. The active tab is titled 'Assign Favorite Locations' and has the URL https://rt202-lm457-dev-7b44497d46f0.herokuapp.com/project/admin/assign_Locations.php?username=s&places=e. The page displays a search interface with two input fields ('s' and 'e') and a 'Search' button. Below the inputs, there are two sections: 'Users' and 'Locations to Assign'. The 'Users' section lists several users with their status: Justinchen (No Favorite Locations) with a checked checkbox, saevi (No Favorite Locations), ssi (Fremont Street Experience, Seventy Seven Acres, The LINQ Promenade, Ninety nine land), sample2 (No Favorite Locations), sesha (Red Rock Canyon National Conservation Area), shannoar (No Favorite Locations), test1 (No Favorite Locations), and tester2 (No Favorite Locations). The 'Locations to Assign' section lists various locations with checkboxes: Fremont Street Experience, Shelby American, Inc., Seventy Seven Acres, KAAB by Cirque du Soleil, Mount Charleston, Red Rock Canyon National Conservation Area, High Roller, The LINQ Promenade, and Ninety nine land. A 'Toggle Favorite Locations' button is located at the bottom left of the page.

[Toggle Favorite Locations](#)

The screenshot depicts the user justinchen who currently has no associations with any locations prior to being associated with the location Fremont Street Experience.

Checklist Items (1)

#3 Show the result of entities and users being associated and unassociated

Added to your Bucket List!

Assign Favorite Locations

Users	Locations to Assign
justinchen	Fremont Street Experience
saiyi	No Favorite Locations
sai	Fremont Street Experience, Seventy Seven Acres, The LINQ Promenade, Ninety nine land
sample2	No Favorite Locations
sasha	Red Rock Canyon National Conservation Area
shannoor	No Favorite Locations
test11	No Favorite Locations
test12	No Favorite Locations

[Toggle Favorite Locations](#)

The result after toggling that association, resulting in a success message, as well as the user justinchen now having a favorite location in the form of Fremont Street Experience.

Checklist Items (1)

#3 Show the result of entities and users being associated and unassociated

Removed from Bucket List!

Assign Favorite Locations

Users	Locations to Assign
justinchen	Fremont Street Experience
saiyi	No Favorite Locations
sai	Seventy Seven Acres
sample2	No Favorite Locations
sasha	KAA by Cirque du Soleil
shannoor	Mount Charleston
test11	Red Rock Canyon National Conservation Area
test12	High Roller
	The LINQ Promenade
	Ninety nine land

Users		Locations to Assign
<input type="checkbox"/> justinchen	No Favorite Locations	<input type="checkbox"/> Fremont Street Experience
<input type="checkbox"/> saevi	No Favorite Locations	<input type="checkbox"/> Shelby American, Inc.
<input type="checkbox"/> sal	Fremont Street Experience, Seventy Seven Acres, The LINQ Promenade, Ninety nine land	<input type="checkbox"/> Seventy Seven Acres
<input type="checkbox"/> sample2	No Favorite Locations	<input type="checkbox"/> KAAB by Cirque du Soleil
<input type="checkbox"/> sesha	Red Rock Canyon National Conservation Area	<input type="checkbox"/> Mount Charleston
<input type="checkbox"/> shannoor	No Favorite Locations	<input type="checkbox"/> Red Rock Canyon National Conservation Area
<input type="checkbox"/> test1	No Favorite Locations	<input type="checkbox"/> High Roller
<input type="checkbox"/> tester2	No Favorite Locations	<input type="checkbox"/> The LINQ Promenade
		<input type="checkbox"/> Ninety nine land

Total Favorite Locations: 0

The page after the same checkboxes are applied once again, resulting in the removal of the association between justinchen and the location Fremont Street Experience alongside a success message indicating the success disassociation.

Checklist Items (1)

#3 Show the result of entities and users being associated and unassociated

Task #2 - Points: 1

Text: Screenshots of the code

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Search form field for finding a partial match of usernames
<input checked="" type="checkbox"/> #2	1	Search form field for finding a partial match of entities
<input checked="" type="checkbox"/> #3	1	Code related to getting a max of 25 results for each field (i.e., 25 users and 25 entities limit)
<input checked="" type="checkbox"/> #4	1	Code that generates the checkboxes next to each list (users and entities)
<input checked="" type="checkbox"/> #5	1	Code related to submitting the checkbox lists
<input checked="" type="checkbox"/> #6	1	Code related to applying the associations upon submission (i.e., add the relationship if it doesn't exist and remove the relationship if it does exist)
<input checked="" type="checkbox"/> #7	1	Include ucid/date comments for each code screenshot
<input checked="" type="checkbox"/> #8	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

```
//search for user by username
$users = [];
$username = "";
```

```

if (isset($_GET["username"])) {
    $username = $_GET["username"];
    if (!empty($username)) {
        $db = getDB();
        $stmt = $db->prepare("SELECT Users.id, username,
        (SELECT GROUP_CONCAT(name SEPARATOR ', ') from
        UserLocations ut JOIN tourist_info t on ut.places_id = t.id WHERE ut.user_id = Users.id) as places
        from Users WHERE username like :username LIMIT 25");
        try {
            $stmt->execute([":username" => "%$username%"]);
            $results = $stmt->fetchAll(PDO::FETCH_ASSOC);
            if ($results) {
                $users = $results;
            }
        } catch (PDOException $e) {
            flash(var_export($e->errorInfo, true), "danger");
        }
    } else {
        flash("Username must not be empty", "warning");
    }
} <- #66-84 if (!empty($username))

```

The code on my assigning relationships page responsible for searching through my existing usernames in the Users table and fetching partial/complete matches based on the data retrieved from the username form field.

Checklist Items (1)

#1 Search form field for finding a partial match of usernames

```

//get active places
$active_places = [];
$places = "";
if (isset($_GET["places"])) { //UCID: LM457
    $places = $_GET["places"]; //DATE: 4/29/2024
    if (!empty($places)) {
        $db = getDB();
        $stmt = $db->prepare("SELECT id, name FROM tourist_info WHERE name like :name LIMIT 25");
        try {
            $stmt->execute(["name" => "%$places%"]);
            $results = $stmt->fetchAll(PDO::FETCH_ASSOC);
            if ($results) {
                $active_places = $results;
            }
        } catch (PDOException $e) {
            flash(var_export($e->errorInfo, true), "danger");
        }
    } else {
        flash("Location must not be empty", "warning");
    }
} <- #42-59 if (isset($_GET["places"]))

```

The code on my assigning relationships page responsible for searching through my existing location names in the tourist_info table and fetching partial/complete matches based on the data retrieved from the location name form field.

Checklist Items (1)

#2 Search form field for finding a partial match of entities

```
$db = getDB();
$stmt = $db->prepare("SELECT id, name FROM tourist_info WHERE name like :name LIMIT 25");
$stmt->
```

My location name query responsible for limiting the number of fetched results to a maximum of 25.

Checklist Items (1)

#3 Code related to getting a max of 25 results for each field (i.e., 25 users and 25 entities limit)

```
$stmt = $db->prepare("SELECT Users.id, username,
(SELECT GROUP_CONCAT(name SEPARATOR ', ') from
UserLocations ut JOIN tourist_info t on ut.places_id = t.id WHERE ut.user_id = Users.id) as places
from Users WHERE username like :username LIMIT 25");
$stmt->
```

My username query responsible for limiting the number of fetched results to a maximum of 25.

Checklist Items (1)

#3 Code related to getting a max of 25 results for each field (i.e., 25 users and 25 entities limit)

```
table class="table">
    <?php foreach ($users as $user) : ?>
        <tr>
            <td>
                <?php render_input(["type" => "checkbox", "id" => "user_".$user['id'], "", false], "name" => "users[]", "label" => $user['username'], "", false, "value" => $user['id'], "", false); ?>
            </td>
            <td><?php se($user, "places", "No Favorite Locations"); ?></td>
                <!--CID: 18457-->
                <!--DATE: 4/29/2024-->
        </tr>
    <?php endforeach; ?>
</table>

<?php foreach ($active_places as $place) : ?>
    <div>
        <?php render_input(["type" => "checkbox", "id" => "places_".$place['id'], "", false], "name" => "places[]", "label" => $place['name'], "", false, "value" => $place['id'], "", false); ?>
    </div>
<?php endforeach; ?>
```

HTML code responsible for creating checkboxes next to each of the users and entities/locations list where an admin user can toggle the relationships between many users and many locations.

Checklist Items (1)

#4 Code that generates the checkboxes next to each list (users and entities)

```
<form method="GET">
    <?php render_input(["type" => "search", "name" => "username", "placeholder" => "Username Search", "value" => $username]; /*lazy value to check if form submitted, not ideal*/ ?>
    <?php render_input(["type" => "search", "name" => "places", "placeholder" => "Location Search", "value" => $places]; /*lazy value to check if form submitted, not ideal*/ ?>
    <?php render_button(["text" => "Search", "type" => "submit"]); ?>
</form>
```

The HTML code responsible for creating the button which searches through the username and location name fields.

Checklist Items (1)

#5 Code related to submitting the checkbox lists

```
</tbody>
</table>
<?php render_button(["text" -> "Toggle Favorite Locations", "type" -> "submit", "color" -> "secondary"]); ?>
```

The HTML code responsible for creating the button which toggles the relationships between the selected users and entities.

Checklist Items (1)

#5 Code related to submitting the checkbox lists

```
if (isset($_POST["users"]) && isset($_POST["places"])) {
    $user_ids = $_POST["users"];
    $places_ids = $_POST["places"];
    if (empty($user_ids) || empty($places_ids)) {
        flash("Both users and places need to be selected", "warning"); //UCID: LM457
    } else {
        $db = getDB();
        $db->query("DELETE FROM user_places WHERE user_id IN ($user_ids) AND place_id IN ($places_ids)"); //UCID: LM458
    }
}
```

```

$stmt = $db->prepare("SELECT user_id, places_id FROM UserLocations WHERE user_id = :uid AND places_id = :pid");
$insert_stmt = $db->prepare("INSERT INTO UserLocations (user_id, places_id) VALUES (:uid, :pid)");
$delete_stmt = $db->prepare("DELETE FROM UserLocations WHERE user_id = :uid AND places_id = :pid");
foreach ($user_ids as $uid) {
    foreach ($places_ids as $pid) {
        try {
            $stmt->execute([":uid" => $uid, ":pid" => $pid]);
            $is_favorite = !$stmt->fetchColumn();
            if ($is_favorite) {
                $delete_stmt->execute([":uid" => $uid, ":pid" => $pid]);
                flash("Removed from Bucket List!", "success");
            } else {
                $insert_stmt->execute([":uid" => $uid, ":pid" => $pid]);
                flash("Added to your Bucket List!", "success");
            }
        } catch (PDOException $e) {
            flash(var_export($e->errorInfo, true), "danger");
        }
    }
} <- #28-34 foreach ($places_ids as $pid)
} <- #19-35 foreach ($user_ids as $uid)
} <- #14-36 else
} <- #9-37 if (isset($_POST["users"]) && isset($_POST["places"]))

```

The code responsible for handling the application and removal of relationships/associations between users and locations upon submission. It'll add a relationship between the selected user and entity if it doesn't already exist, and remove it if it already exists.

Checklist Items (1)

#6 Code related to applying the associations upon submission (i.e., add the relationship if it doesn't exist and remove the relationship if it does exist)

Task #3 - Points: 1

Text: Explain the solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Describe the steps for the search and how it works for users and entities
<input checked="" type="checkbox"/> #2	1	Mention how you built the UI for this
<input checked="" type="checkbox"/> #3	1	Describe the steps for the associate/unassociate logic for the combination of users and entities

Response:

The search logic for both the users and the entities/locations works in a similar manner. First, an array to store the results and a separate variable to hold the passed in values for each form field are declared. Assuming a value is entered for each of the two required form fields, a \$_GET request is made to retrieve those values and attach to the previously declared variables. Specifically for the users search, a SQL query is prepared that fetches both the user_id and username from the Users table corresponding to the form value using a LIKE clause to make partial matches in addition to a GROUP CONCAT function that fetches all the names of locations in a neatly separated manner from the tourist_info table after joining it on the UserLocations table on places_id WHERE the user_ids match as well as a LIMIT clause to ensure no more than 25 records are fetched at a single time to reduce server side burden. This essentially allows us to fetch both the names of the users corresponding to the matches from the form value as well as previously associated entities per fetched user. For the entities/locations query, it is a bit more straightforward in that only ids and names of each location are fetched from the tourist_info table where yet again a LIKE clause is made to partially

search given the form value and a LIMIT clause of 25 records is put in place yet again as well. All results are then embedded into the respective arrays that were initially declared.

The UI for this page is nearly identical to my assign roles page, where upon page load, a user is presented with two form fields alongside a search button. Following the submission of the form, any matches results are displayed in a list/table with two columns, one for the matching users and the other for the matching location names, with checkboxes beside each user and location to toggle new associations or remove previously existing ones. The results array from both searches are parsed through per record, so each record is formatted per row on their respective column. A toggle relationship button is found below to submit new/remove associations after selecting the desired checkboxes.

Upon clicking the toggle relationships button, the code responsible for handling the creation/deletion of those associations occurs on the same page as well. Firstly, we check to ensure at least one of each field/checkbox from the users and the locations column were selected via a \$_POST request. Upon checking, three distinct SQL queries are prepared, each necessary for performing different operations. The first is a SELECT query that checks to see if a relationship between the selected users and locations exists previously. If any records are returned, then that means an association between the toggled user/location combination exists, and vice-versa if no records are returned. In the event records are returned, indicating an existing association, the second query which is a DELETE statement is utilized to remove the corresponding entry/record from the UserLocations relationship table WHERE the user_id and places_id exist in the table. On the contrary, in the event that no records are returned, indicating an absence of an association, the third query which is an INSERT statement is utilized to insert a new record into the UserLocations relationship table with the corresponding user_id and places_id, thus creating a new relationship/associated between those two variables.

Task #4 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Include the heroku prod link for the page that creates the association
<input checked="" type="checkbox"/> #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

https://it202-lm457-prod-cf97d0c34365.herokuapp.com/project/admin/assign_Locations.php

URL #2

<https://github.com/Kith07/spring2024-IT202/pull/81>

URL #3

<https://github.com/Kith07/spring2024-IT202/pull/85>

Misc (1 pt.)

Task #1 - Points: 1

Text: Screenshot of your project board from GitHub (tasks should be in the proper column)

Task Screenshots:

Gallery Style: Large View

Small Medium Large

The screenshot shows a GitHub project board with three columns: Todo, In Progress, and Done. The Todo column has one item: "This item hasn't been started". The In Progress column has one item: "This is actively being worked on". The Done column has five items, each with a checkmark and a link:

- spring2024-IT202 #70 API Data Association
- spring2024-IT202 #71 Handle the association of data to a user
- spring2024-IT202 #72 Logged in user's associated entities page
- spring2024-IT202 #73 All Users association page
- spring2024-IT202 #74 Create a page that shows data not associated with any user

Github project board with all issues in the done column and closed.

Task #2 - Points: 1

Text: Provide a direct link to the project board on GitHub

URL #1

<https://github.com/Kith07/projects/1/views/1>

Task #3 - Points: 1

Text: Talk about any issues or learnings during this assignment

Response:

Overall, this project was a very rewarding and enriching experience, especially as an intro into web development for someone like me who's a Data Science major. I would be lying if I said this class was easy, since it was anything but that. It is so highly involved with so many different languages and interfaces that for someone who hasn't previously

had much exposure to it, it was a very tough road in terms of properly understanding it and then being able to apply it practically. I certainly learned so much about PHP, JS their respective roles in the server side and client side, HTML and CSS for styling and formatting as well as working with MySQL databases to run SQL queries. I think the biggest obstacle to the learning experience of this class is very much the course syllabus/content itself, which works as a double-edged sword. While it is what makes the course worthwhile, I think the transition from learning the basics of each language to building a web application was incredibly tough for myself and it took me an extensive amount of time to accommodate for this discrepancy. If it wasn't for the time duration/length, there was little to no difficulty in terms of the content of the course itself. Professor Toegel made the class all the more convenient and it was a very enjoyable class and I would definitely do it all over again!

Task #4 - Points: 1

Text: WakaTime Screenshot

Details:

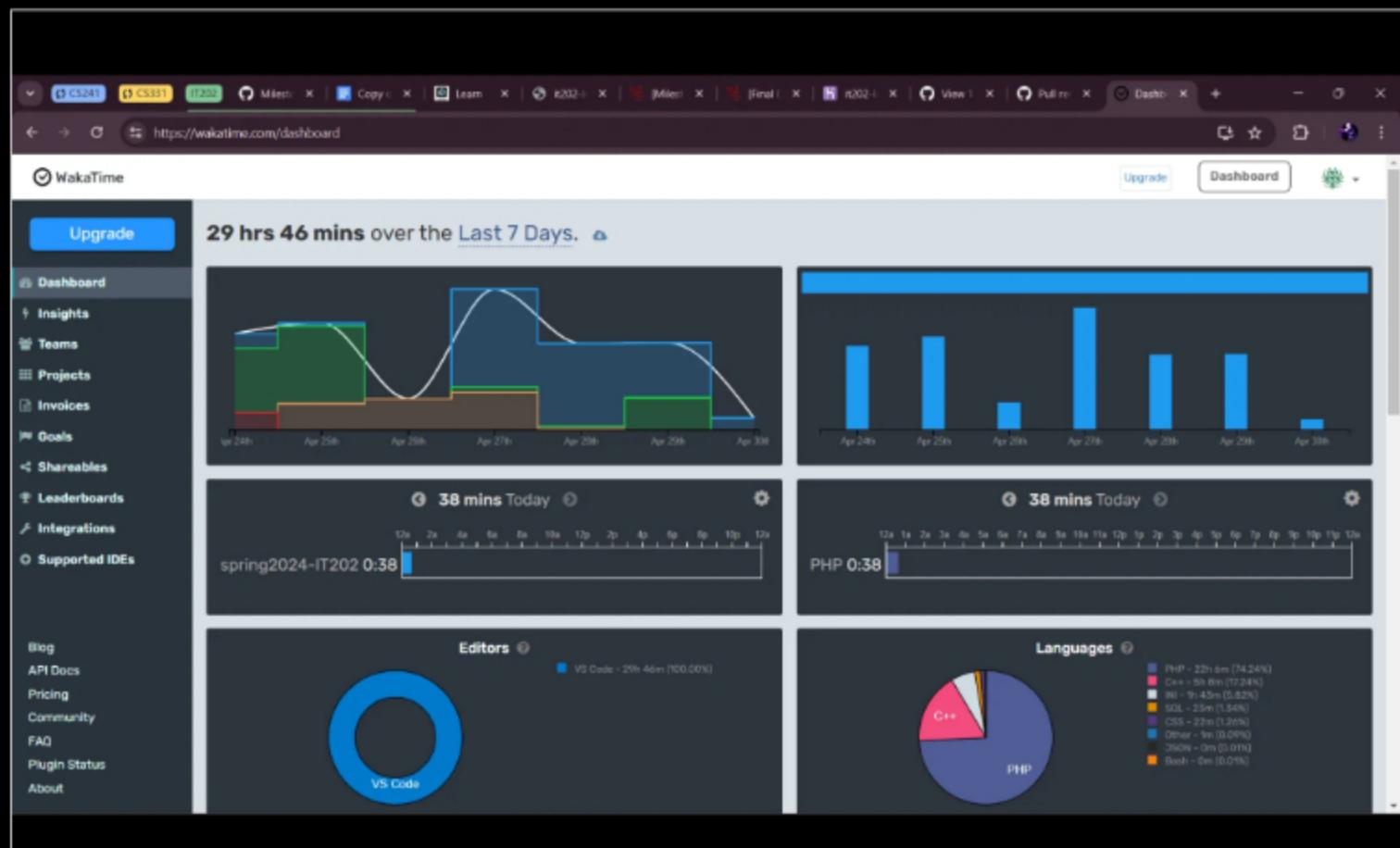
Task Screenshots:

Gallery Style: Large View

Small

Medium

Large



WakaTime dashboard summary of time spent on MS3.

End of Assignment