**University of British Columbia, Vancouver**
Department of Computer Science

# CPSC 304 Project Cover Page

Milestone #: 4

Date: 2024-07-12

Group Number: 18

| Name | Student Number | CS Alias (Userid) | Preferred E-mail Address |
|---|---|---|---|
| Charity Grey | 81808313 | s1i9f | charity.grey1@gmail.com |
| Marcus Guay | 57747115 | u2y6v | marcus.guay.99@gmail.com |
| Sarah Yu | 77021384 | l0t3a | sarahjxyu07@gmail.com |

## Repository Link:

https://github.students.cs.ubc.ca/CPSC304-2024S-T2/project_l0t3a_s1i9f_u2y6v

## SQL script

**Script to create all the tables and data in the database:**

- **See file testVersionControlDemoTable.sql**

**The queries we use in our scripts can be found in the files in:** backend/src/controllers**, and are also specified for the respective query demo below.**

- **Due to a lack of available examples, if assertions or triggers are required, you may simply say what they would need to do, but not implement them:**

Yes, we do have to have an assertion for the following:

- Each row in folder has a foreign key to the blob with empty attribute filecontent

## Project Description

**A short description of the final project, and what it accomplished.**
We plan to model a database and a GUI for a version control system like Github. Users will be able to add, delete and edit files and folders, and commit those actions to a repository on any branch they would like, as well as the ability to revert those changes if necessary. Users can also add, edit, and delete issues, and add comments.

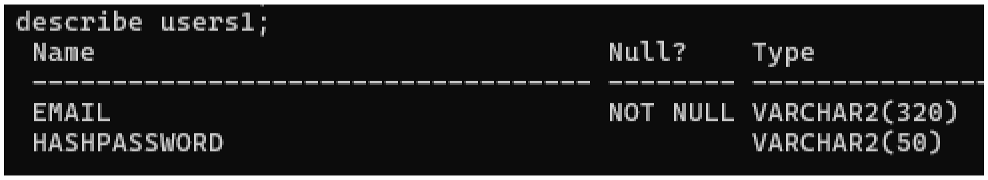
## Differing Schema declaration

**A description of how your final schema differed from the schema you turned in. If the final schema differed, explain why.**
We modified from in Milestone 2:

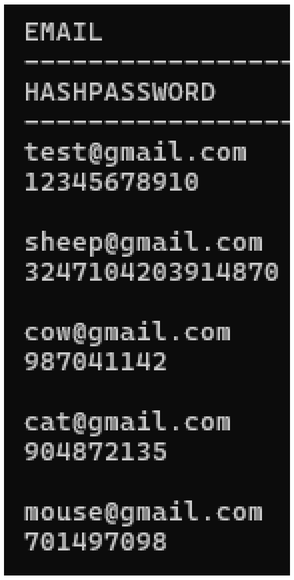
- for fixing some DDL bugs (missing semicolons, putting the wrong tablename in "INSERT INTO <Tablename>")
- Adding ON DELETE CASCADE inside the table Comments for the issue foreign key to demonstrate on delete functionality

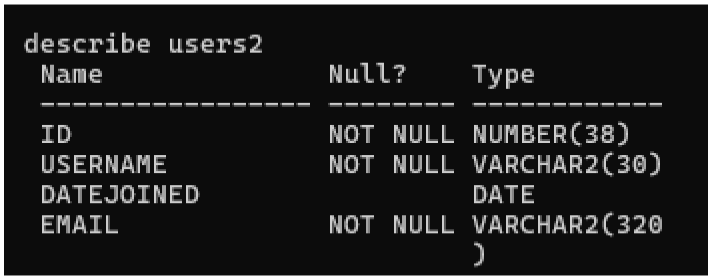
## Schema and data

**A copy of the schema and screenshots that show what data is present in each relation after the SQL initialization script is run.**

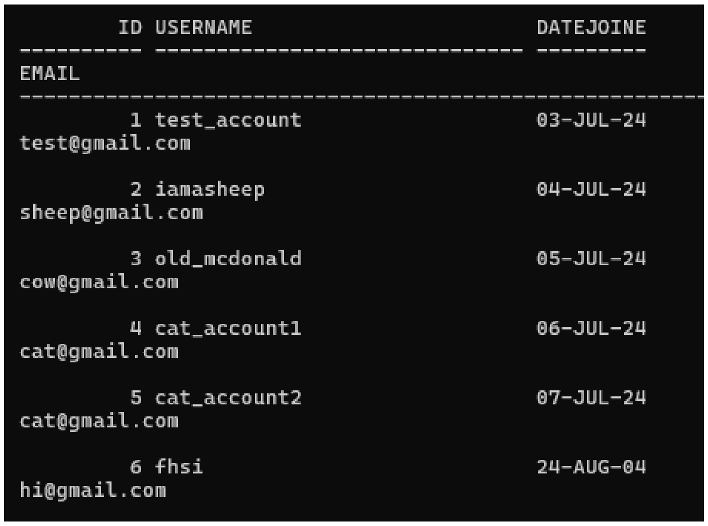
<table>
<tr><th>Table of insert</th><th>Schema and data</th></tr>
<tr><td>Users1</td><td>


```
describe users1;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 EMAIL                                     NOT NULL VARCHAR2(320)
 HASHPASSWORD                                       VARCHAR2(50)
```



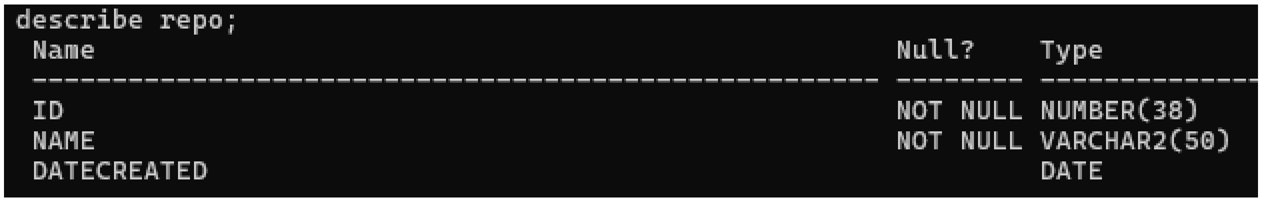
```
EMAIL
-----------------
HASHPASSWORD
-----------------
test@gmail.com
12345678910

sheep@gmail.com
3247104203914870

cow@gmail.com
987041142

cat@gmail.com
904872135

mouse@gmail.com
701497098
```
</td></tr>
<tr><td>Users2</td><td>


```
describe users2
 Name                 Null?    Type
 -------------------- -------- ------------
 ID                   NOT NULL NUMBER(38)
 USERNAME             NOT NULL VARCHAR2(30)
 DATEJOINED                    DATE
 EMAIL                NOT NULL VARCHAR2(320
                               )
```
</td></tr>
</table>

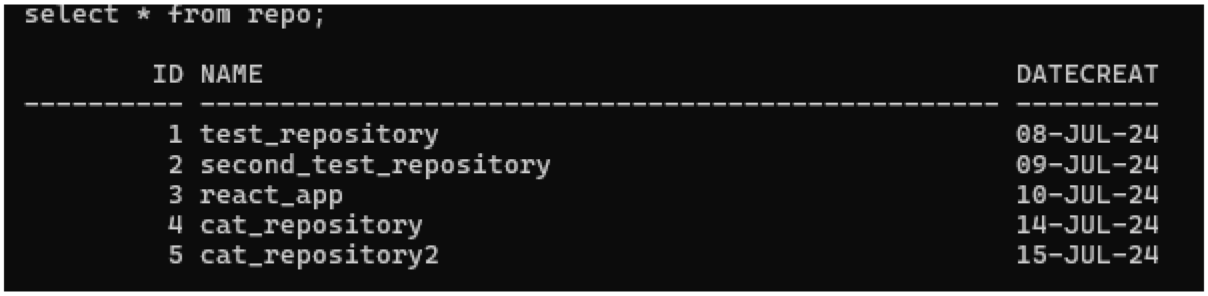


```
        ID USERNAME                      DATEJOINE
---------- ----------------------------- ---------
EMAIL
-------------------------------------------------------
         1 test_account                  03-JUL-24
test@gmail.com

         2 iamasheep                     04-JUL-24
sheep@gmail.com

         3 old_mcdonald                  05-JUL-24
cow@gmail.com

         4 cat_account1                  06-JUL-24
cat@gmail.com

         5 cat_account2                  07-JUL-24
cat@gmail.com

         6 fhsi                          24-AUG-04
hi@gmail.com
```

Repo

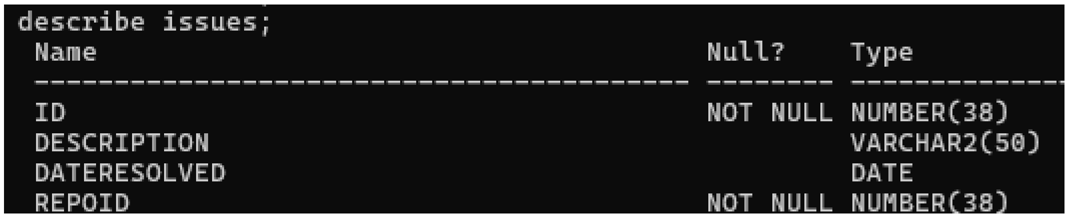


```
describe repo;
 Name                                                        Null?    Type
 ----------------------------------------------------------- -------- --------------
 ID                                                          NOT NULL NUMBER(38)
 NAME                                                        NOT NULL VARCHAR2(50)
 DATECREATED                                                          DATE
```

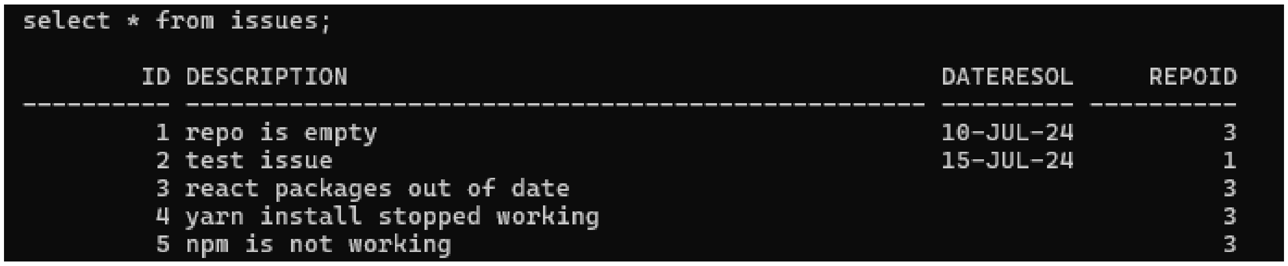


```
select * from repo;

        ID NAME                                               DATECREAT
---------- -------------------------------------------------- ---------
         1 test_repository                                    08-JUL-24
         2 second_test_repository                             09-JUL-24
         3 react_app                                          10-JUL-24
         4 cat_repository                                     14-JUL-24
         5 cat_repository2                                    15-JUL-24
```

Issues

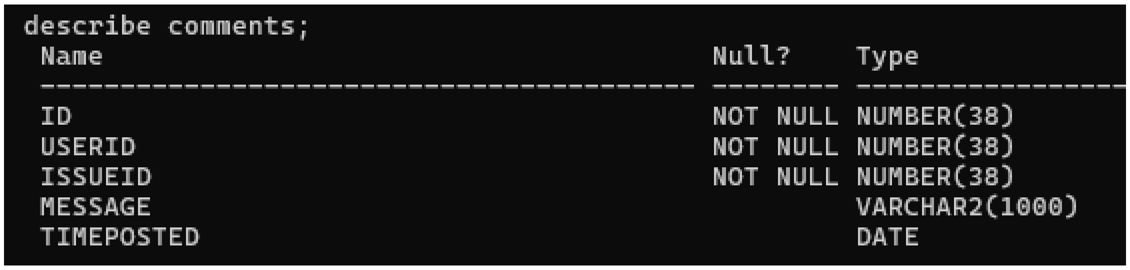


```
describe issues;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------
 ID                                        NOT NULL NUMBER(38)
 DESCRIPTION                                        VARCHAR2(50)
 DATERESOLVED                                       DATE
 REPOID                                    NOT NULL NUMBER(38)
```

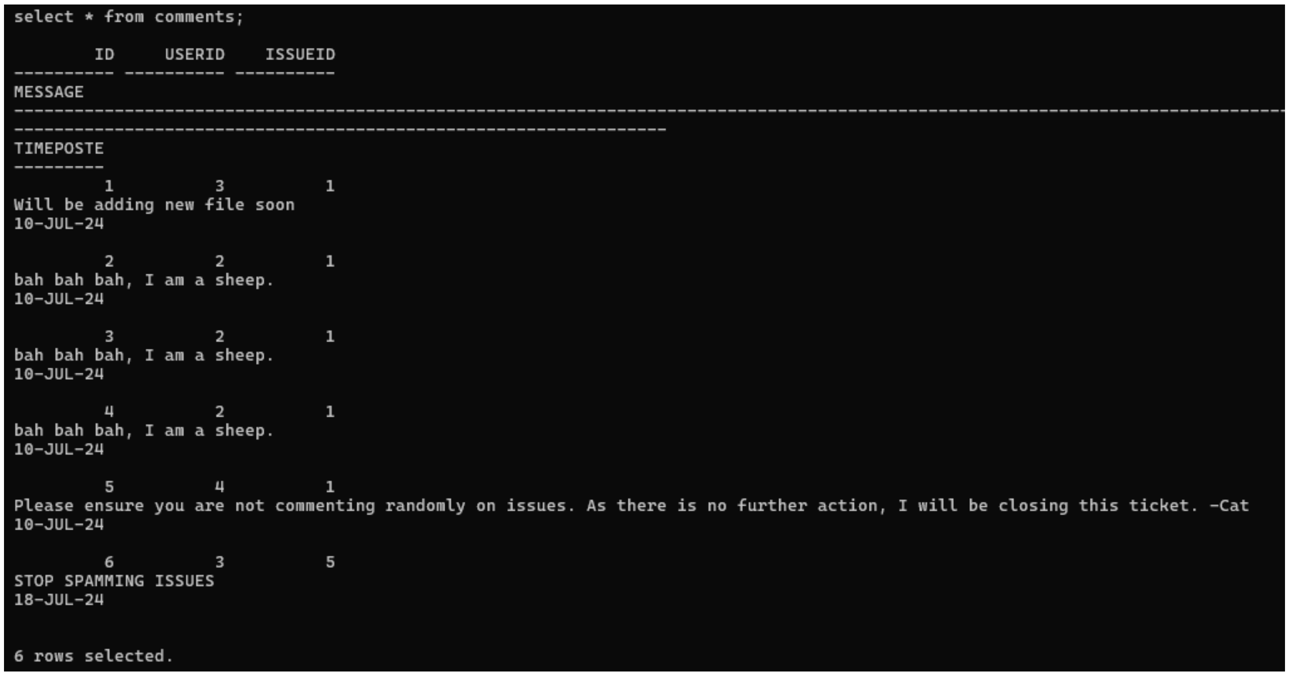


```
select * from issues;

        ID DESCRIPTION                                        DATERESOL     REPOID
---------- -------------------------------------------------- --------- ----------
         1 repo is empty                                      10-JUL-24          3
         2 test issue                                         15-JUL-24          1
         3 react packages out of date                                            3
         4 yarn install stopped working                                          3
         5 npm is not working                                                    3
```

Comments

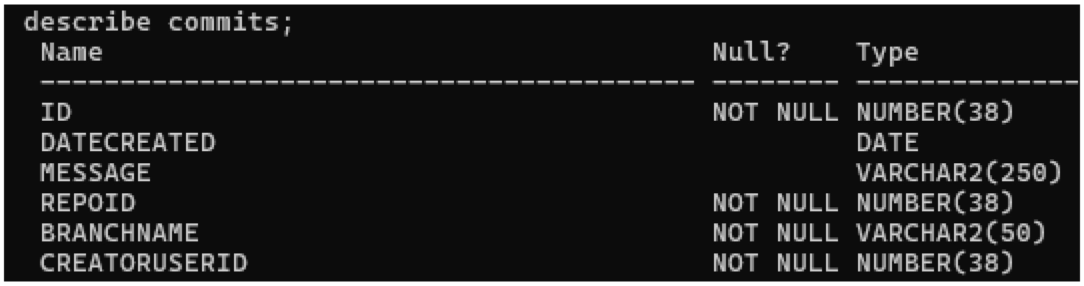


```
describe comments;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 ID                                        NOT NULL NUMBER(38)
 USERID                                    NOT NULL NUMBER(38)
 ISSUEID                                   NOT NULL NUMBER(38)
 MESSAGE                                            VARCHAR2(1000)
 TIMEPOSTED                                         DATE
```

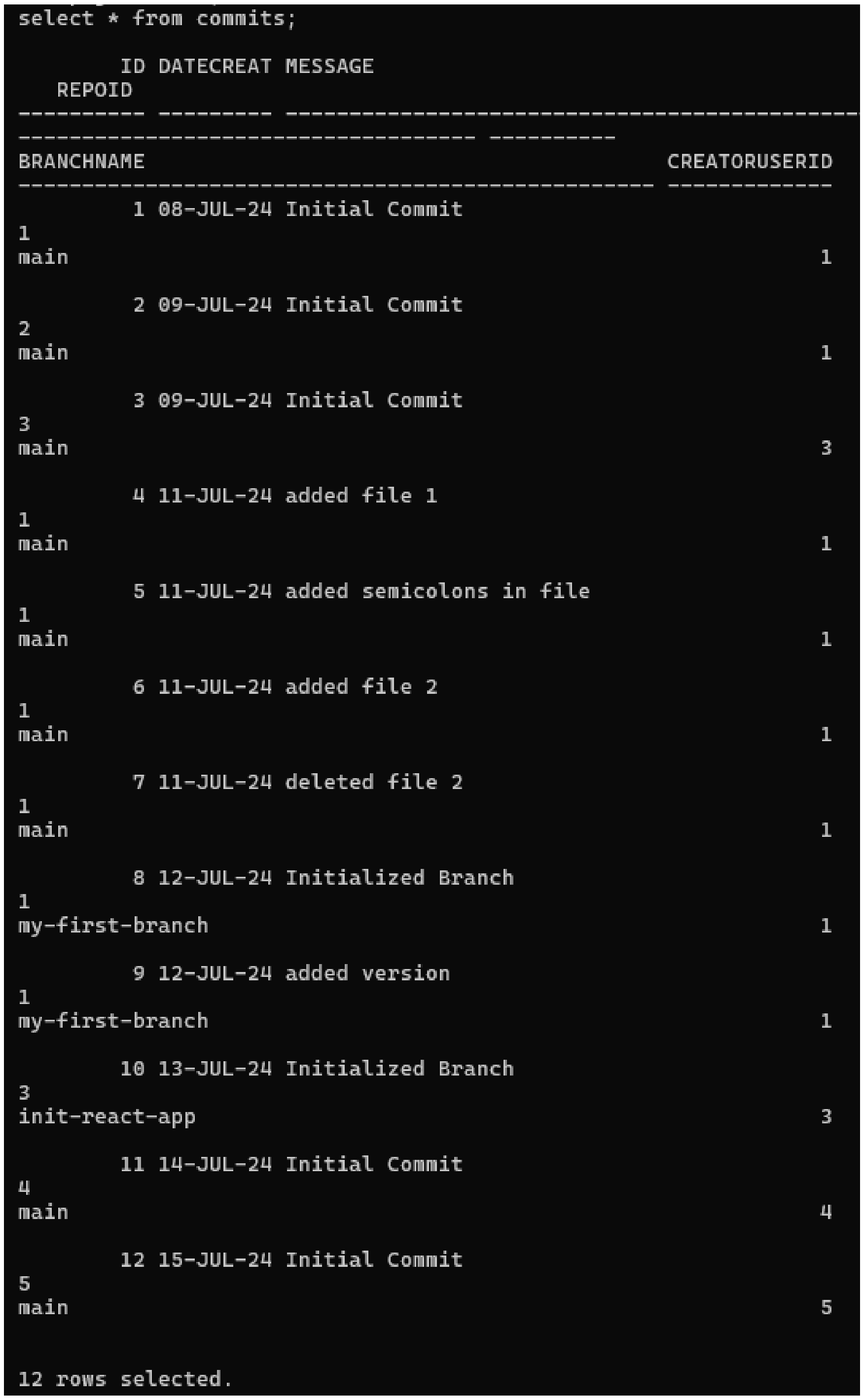


```
select * from comments;

        ID     USERID    ISSUEID
---------- ---------- ----------
MESSAGE
--------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------
TIMEPOSTE
---------
         1          3          1
Will be adding new file soon
10-JUL-24

         2          2          1
bah bah bah, I am a sheep.
10-JUL-24

         3          2          1
bah bah bah, I am a sheep.
10-JUL-24

         4          2          1
bah bah bah, I am a sheep.
10-JUL-24

         5          4          1
Please ensure you are not commenting randomly on issues. As there is no further action, I will be closing this ticket. -Cat
10-JUL-24

         6          3          5
STOP SPAMMING ISSUES
18-JUL-24


6 rows selected.
```

Commits

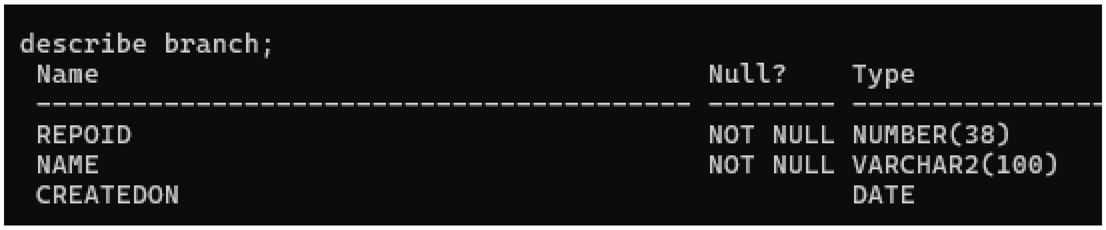


```
describe commits;
 Name                                      Null?    Type
 ----------------------------------------- -------- -------------
 ID                                        NOT NULL NUMBER(38)
 DATECREATED                                        DATE
 MESSAGE                                            VARCHAR2(250)
 REPOID                                    NOT NULL NUMBER(38)
 BRANCHNAME                                NOT NULL VARCHAR2(50)
 CREATORUSERID                             NOT NULL NUMBER(38)
```

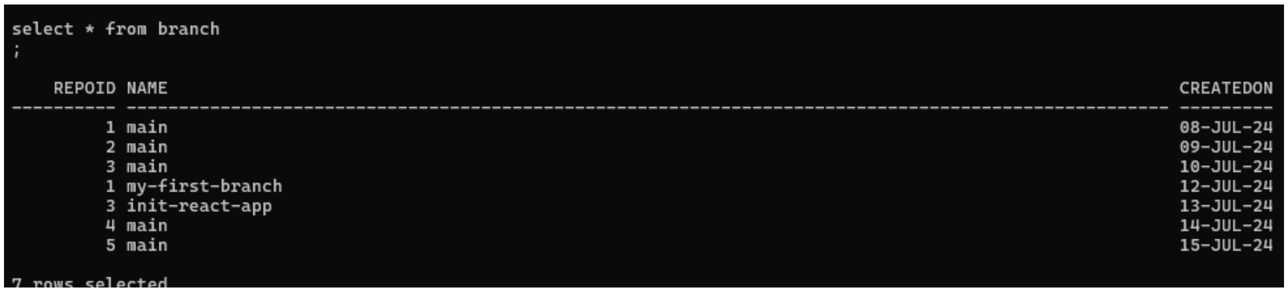


```
select * from commits;

        ID DATECREAT MESSAGE
   REPOID
---------- --------- ---------------------------------------------
------------------------------------ ----------
BRANCHNAME                                         CREATORUSERID
-------------------------------------------------- -------------
         1 08-JUL-24 Initial Commit
1
main                                                           1

         2 09-JUL-24 Initial Commit
2
main                                                           1

         3 09-JUL-24 Initial Commit
3
main                                                           3

         4 11-JUL-24 added file 1
1
main                                                           1

         5 11-JUL-24 added semicolons in file
1
main                                                           1

         6 11-JUL-24 added file 2
1
main                                                           1

         7 11-JUL-24 deleted file 2
1
main                                                           1

         8 12-JUL-24 Initialized Branch
1
my-first-branch                                                1

         9 12-JUL-24 added version
1
my-first-branch                                                1

        10 13-JUL-24 Initialized Branch
3
init-react-app                                                 3

        11 14-JUL-24 Initial Commit
4
main                                                           4

        12 15-JUL-24 Initial Commit
5
main                                                           5


12 rows selected.
```

Branch

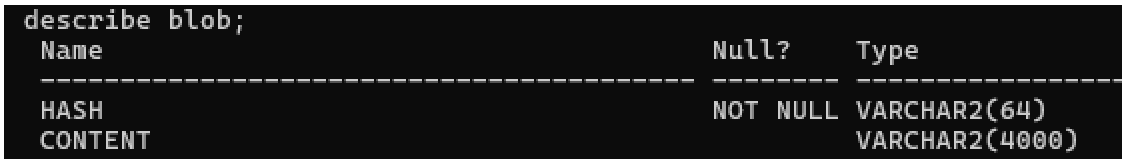


```
describe branch;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 REPOID                                    NOT NULL NUMBER(38)
 NAME                                      NOT NULL VARCHAR2(100)
 CREATEDON                                          DATE
```

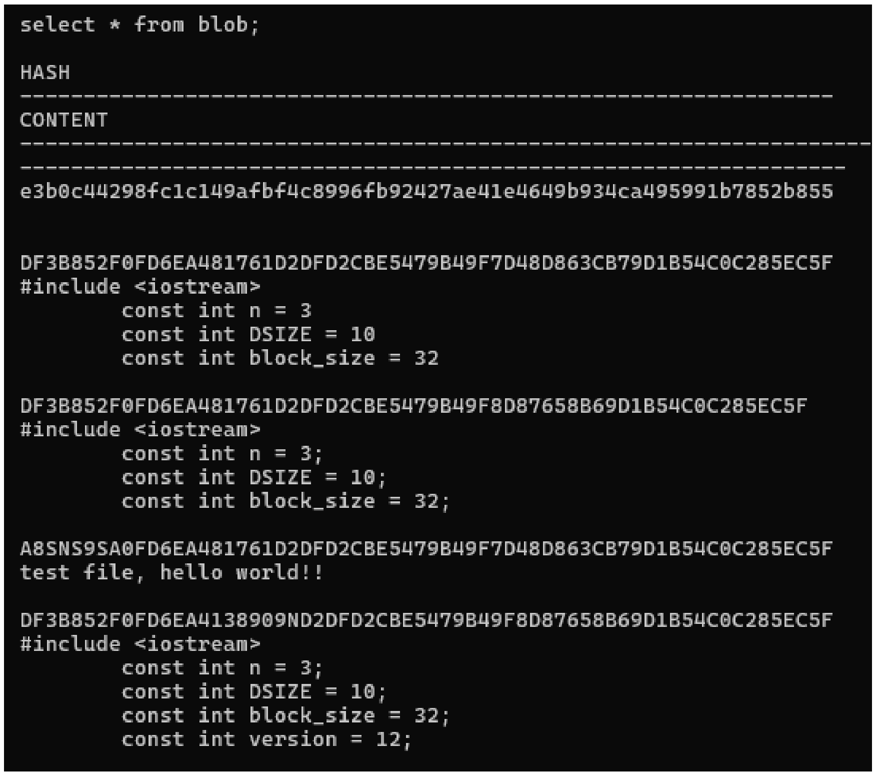


```
select * from branch
;

    REPOID NAME                                                                                                 CREATEDON
---------- ---------------------------------------------------------------------------------------------------- ---------
         1 main                                                                                                 08-JUL-24
         2 main                                                                                                 09-JUL-24
         3 main                                                                                                 10-JUL-24
         1 my-first-branch                                                                                      12-JUL-24
         3 init-react-app                                                                                       13-JUL-24
         4 main                                                                                                 14-JUL-24
         5 main                                                                                                 15-JUL-24

7 rows selected
```

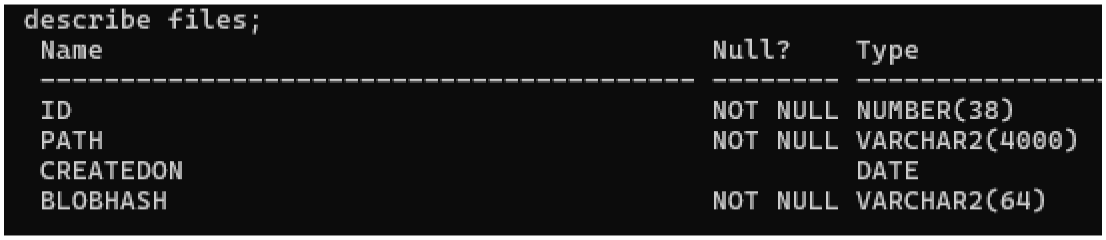
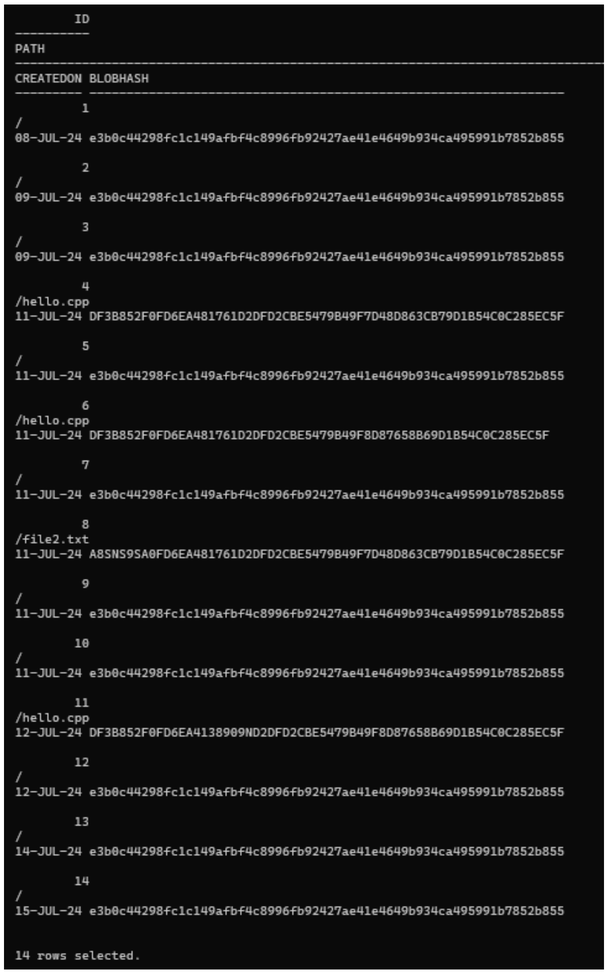
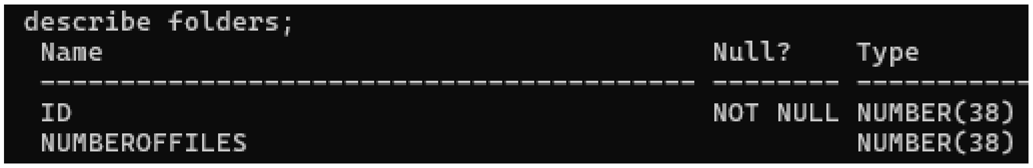
Blob



```
describe blob;
 Name                                      Null?    Type
 ----------------------------------------- -------- ------------------
 HASH                                      NOT NULL VARCHAR2(64)
 CONTENT                                            VARCHAR2(4000)
```



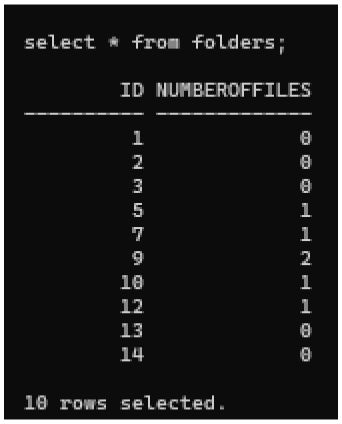
```
select * from blob;

HASH
----------------------------------------------------------------
CONTENT
------------------------------------------------------------------
-----------------------------------------------------------------
e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855


DF3B852F0FD6EA481761D2DFD2CBE5479B49F7D48D863CB79D1B54C0C285EC5F
#include <iostream>
        const int n = 3
        const int DSIZE = 10
        const int block_size = 32

DF3B852F0FD6EA481761D2DFD2CBE5479B49F8D87658B69D1B54C0C285EC5F
#include <iostream>
        const int n = 3;
        const int DSIZE = 10;
        const int block_size = 32;

A8SNS9SA0FD6EA481761D2DFD2CBE5479B49F7D48D863CB79D1B54C0C285EC5F
test file, hello world!!

DF3B852F0FD6EA4138909ND2DFD2CBE5479B49F8D87658B69D1B54C0C285EC5F
#include <iostream>
        const int n = 3;
        const int DSIZE = 10;
        const int block_size = 32;
        const int version = 12;
```

| | |
|---|---|
| Files |   |
| Folders |  |

```
describe files;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 ID                                        NOT NULL NUMBER(38)
 PATH                                      NOT NULL VARCHAR2(4000)
 CREATEDON                                          DATE
 BLOBHASH                                  NOT NULL VARCHAR2(64)
```

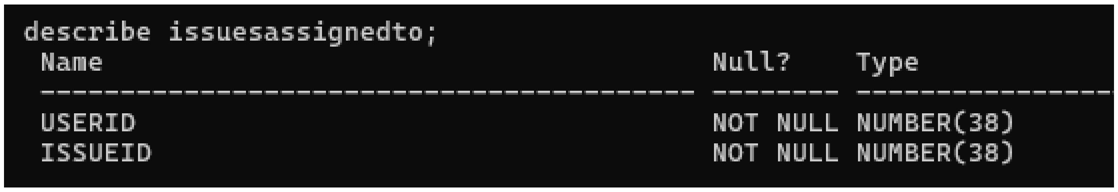
```
        ID
----------
PATH
--------------------------------------------------------------------------------
CREATEDON BLOBHASH
--------- ----------------------------------------------------------------
         1
/
08-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

         2
/
09-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

         3
/
09-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

         4
/hello.cpp
11-JUL-24 DF3B852F0FD6EA481761D2DFD2CBE5479B49F7D48D863CB79D1B54C0C285EC5F

         5
/
11-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

         6
/hello.cpp
11-JUL-24 DF3B852F0FD6EA481761D2DFD2CBE5479B49F8D87658B69D1B54C0C285EC5F

         7
/
11-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

         8
/file2.txt
11-JUL-24 A8SNS9SA0FD6EA481761D2DFD2CBE5479B49F7D48D863CB79D1B54C0C285EC5F

         9
/
11-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

        10
/
11-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

        11
/hello.cpp
12-JUL-24 DF3B852F0FD6EA4138909ND2DFD2CBE5479B49F8D87658B69D1B54C0C285EC5F

        12
/
12-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

        13
/
14-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

        14
/
15-JUL-24 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855


14 rows selected.
```

```
describe folders;
 Name                                      Null?    Type
 ----------------------------------------- -------- -----------
 ID                                        NOT NULL NUMBER(38)
 NUMBEROFFILES                                      NUMBER(38)
```

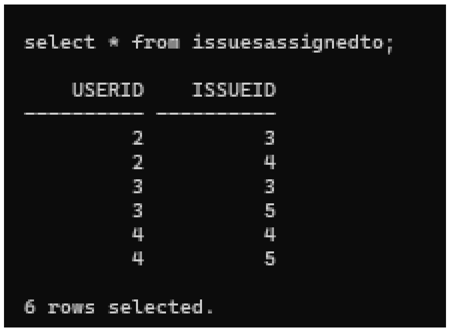


```
select * from folders;

        ID NUMBEROFFILES
---------- -------------
         1             0
         2             0
         3             0
         5             1
         7             1
         9             2
        10             1
        12             1
        13             0
        14             0

10 rows selected.
```

IssuesAssignedTo

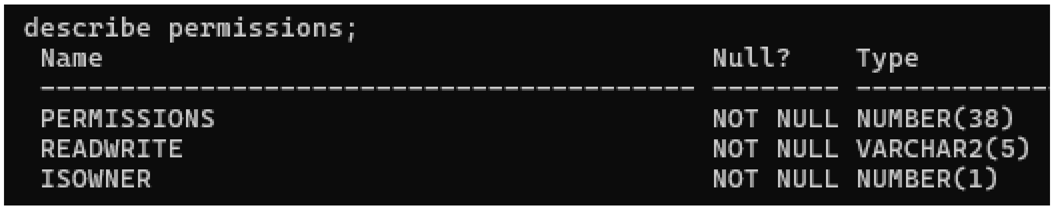


```
describe issuesassignedto;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 USERID                                    NOT NULL NUMBER(38)
 ISSUEID                                   NOT NULL NUMBER(38)
```

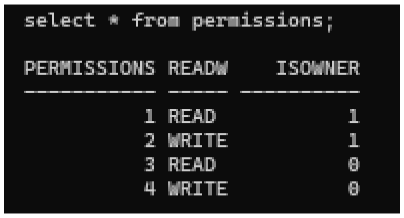


```
select * from issuesassignedto;

    USERID    ISSUEID
---------- ----------
         2          3
         2          4
         3          3
         3          5
         4          4
         4          5

6 rows selected.
```

Permissions



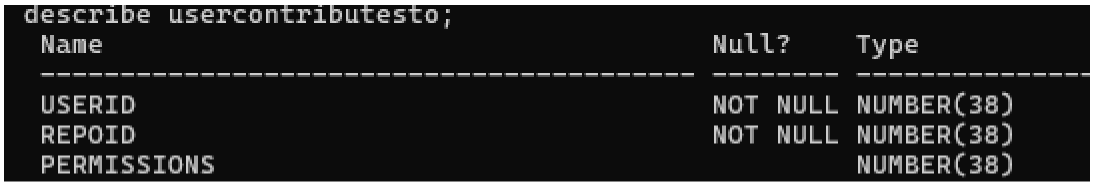
```
describe permissions;
 Name                                      Null?    Type
 ----------------------------------------- -------- ------------
 PERMISSIONS                               NOT NULL NUMBER(38)
 READWRITE                                 NOT NULL VARCHAR2(5)
 ISOWNER                                   NOT NULL NUMBER(1)
```



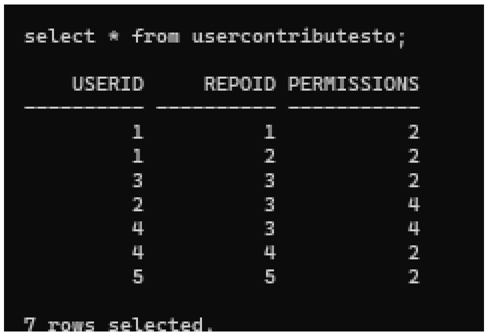
```
select * from permissions;

PERMISSIONS READW    ISOWNER
----------- ----- ----------
          1 READ           1
          2 WRITE          1
          3 READ           0
          4 WRITE          0
```

**NOTE: we confirmed with Jessica that only 4 rows is okay.**

UserContributesTo

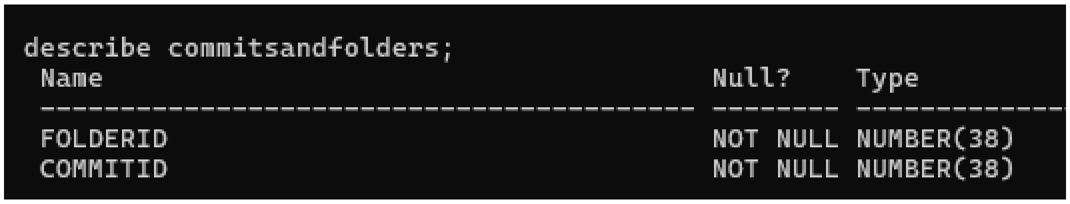


```
describe usercontributesto;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 USERID                                    NOT NULL NUMBER(38)
 REPOID                                    NOT NULL NUMBER(38)
 PERMISSIONS                                        NUMBER(38)
```

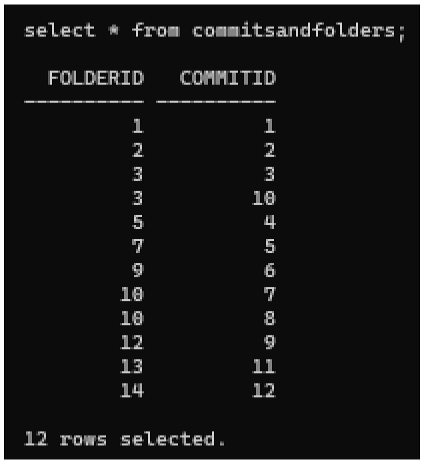


```
select * from usercontributesto;

    USERID     REPOID PERMISSIONS
---------- ---------- -----------
         1          1           2
         1          2           2
         3          3           2
         2          3           4
         4          3           4
         4          4           2
         5          5           2

7 rows selected.
```

CommitsAndFolders

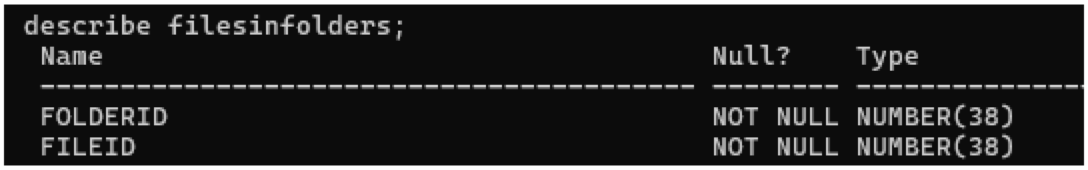


```
describe commitsandfolders;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 FOLDERID                                  NOT NULL NUMBER(38)
 COMMITID                                  NOT NULL NUMBER(38)
```

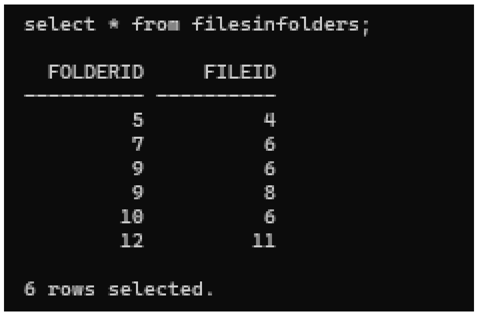


```
select * from commitsandfolders;

  FOLDERID   COMMITID
---------- ----------
         1          1
         2          2
         3          3
         3         10
         5          4
         7          5
         9          6
        10          7
        10          8
        12          9
        13         11
        14         12

12 rows selected.
```

FilesInFolders



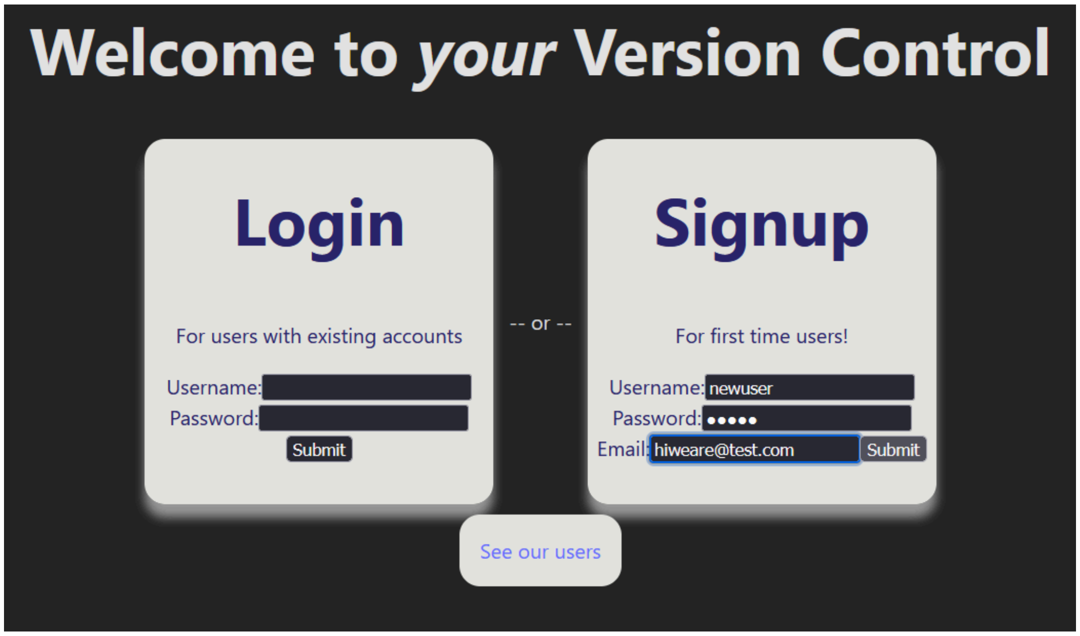
```
describe filesinfolders;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 FOLDERID                                  NOT NULL NUMBER(38)
 FILEID                                    NOT NULL NUMBER(38)
```

| | |
|---|---|
| |  |

```
select * from filesinfolders;

  FOLDERID     FILEID
---------- ----------
         5          4
         7          6
         9          6
         9          8
        10          6
        12         11

6 rows selected.
```

## QUERY DEMOS:

| | |
|---|---|
| | **NOTE IN GENERAL:**<br><br>**In general, our functionality is as follows:**<br>**1)** User takes action to click a button that requires a query to be called or a reactComponent that requires a query is rendered. This is done to our frontend server.<br>This is found in files in our: **/frontend/src**<br>**2)** The fetch request is called to the backend server<br>The fetch GET/POST request can be found in either the react component itself, or the **frontend/src/controller/controllers.jsx** file<br>**3)** The request is received by our backend server, which makes a call to the database. This is where the SQL query is actually found.<br>This is found in the files inside the folder: **backend/src/controllers** |
| **Queries: INSERT Operation** | SQL Query can be found in: **backend/src/controllers/mainController.js**<br>Under the async function **addUserToDB(req, res)**<br><br>BEFORE: adding rows into User1 and User2. |



# Welcome to *your* Version Control

## Login

For users with existing accounts

Username:
Password:
Submit

-- or --

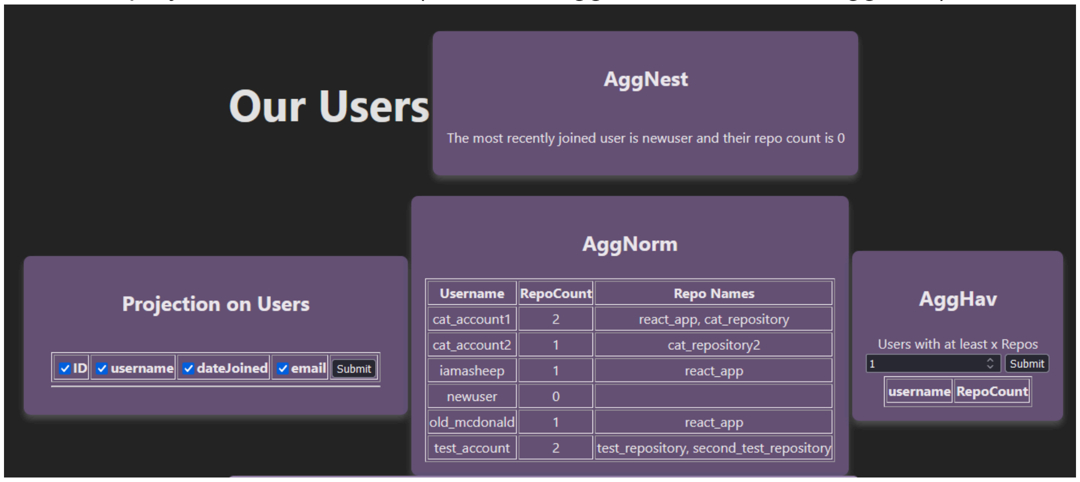
## Signup

For first time users!

Username: newuser
Password: ●●●●●
Email: hiweare@test.com
Submit

See our users

AFTER: proof that the `COMMIT;` operation was called and the newuser we added is now displayed on our userlist (see table AggNorm below and AggNest)



## Our Users

### AggNest

The most recently joined user is newuser and their repo count is 0

### AggNorm

| Username | RepoCount | Repo Names |
|---|---|---|
| cat_account1 | 2 | react_app, cat_repository |
| cat_account2 | 1 | cat_repository2 |
| iamasheep | 1 | react_app |
| newuser | 0 | |
| old_mcdonald | 1 | react_app |
| test_account | 2 | test_repository, second_test_repository |

### Projection on Users

✓ ID ✓ username ✓ dateJoined ✓ email Submit

### AggHav

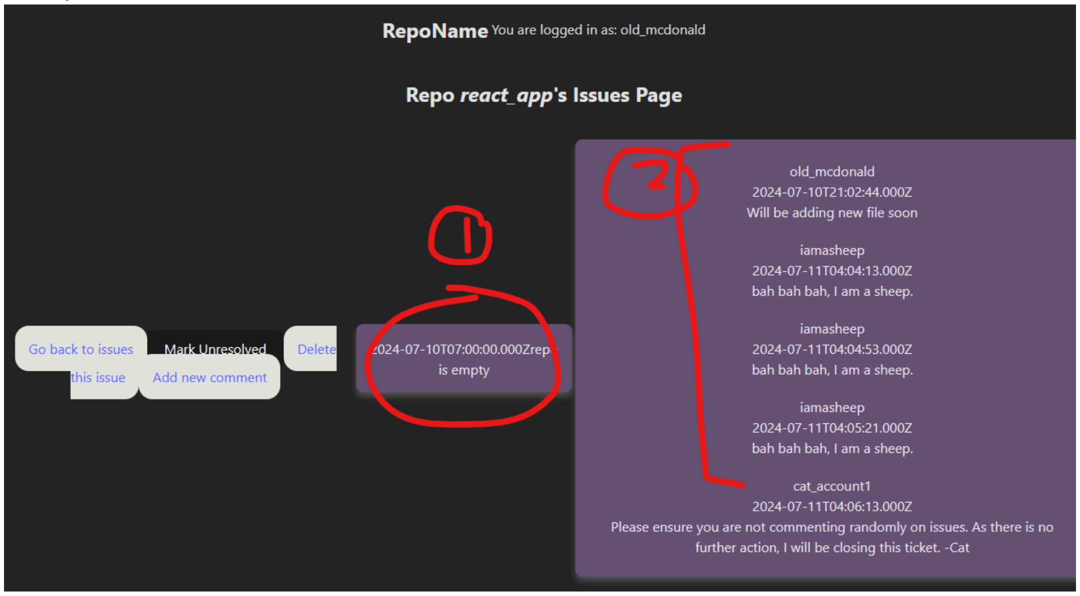
Users with at least x Repos
1 Submit

| username | RepoCount |
|---|---|

**Queries: DELETE Operation**

SQL Query can be found in: **backend/src/controllers/mainController.js**
Under the async function **deleteIssue(req, res)**
Delete on  Issues (cascades to comments)

BEFORE:
1) Is info about the issue
2) Is comments on the issue



**RepoName** You are logged in as: old_mcdonald

**Repo *react_app*'s Issues Page**

Go back to issues | Mark Unresolved | Delete this issue | Add new comment

2024-07-10T07:00:00.000Zrepo is empty

old_mcdonald
2024-07-10T21:02:44.000Z
Will be adding new file soon

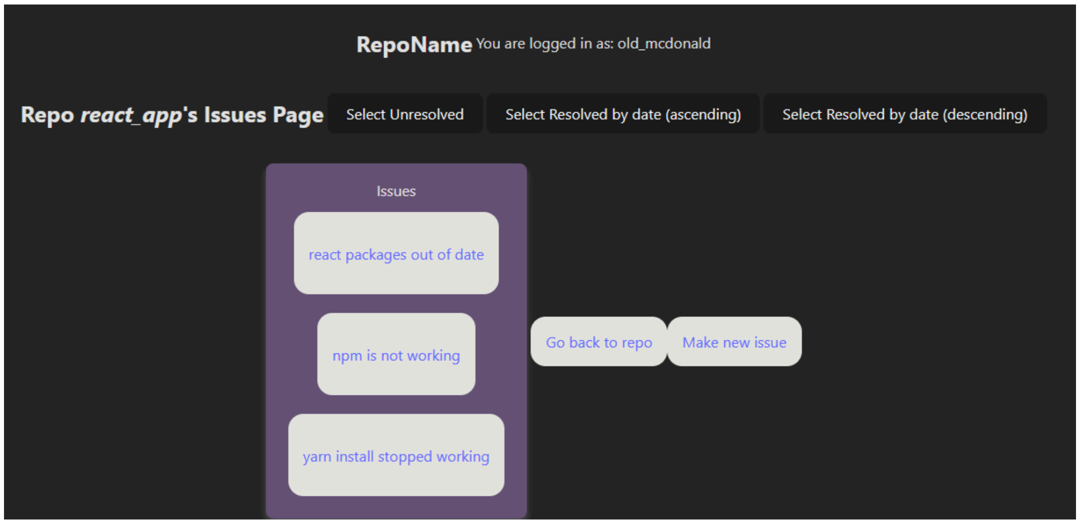
iamasheep
2024-07-11T04:04:13.000Z
bah bah bah, I am a sheep.

iamasheep
2024-07-11T04:04:53.000Z
bah bah bah, I am a sheep.

iamasheep
2024-07-11T04:05:21.000Z
bah bah bah, I am a sheep.

cat_account1
2024-07-11T04:06:13.000Z
Please ensure you are not commenting randomly on issues. As there is no further action, I will be closing this ticket. -Cat

AFTER: pressing *delete this issue*, we are redirected to the issues page for this repo, which no longer shows this issue: (There is no issue named "repo is empty")



**RepoName** You are logged in as: old_mcdonald

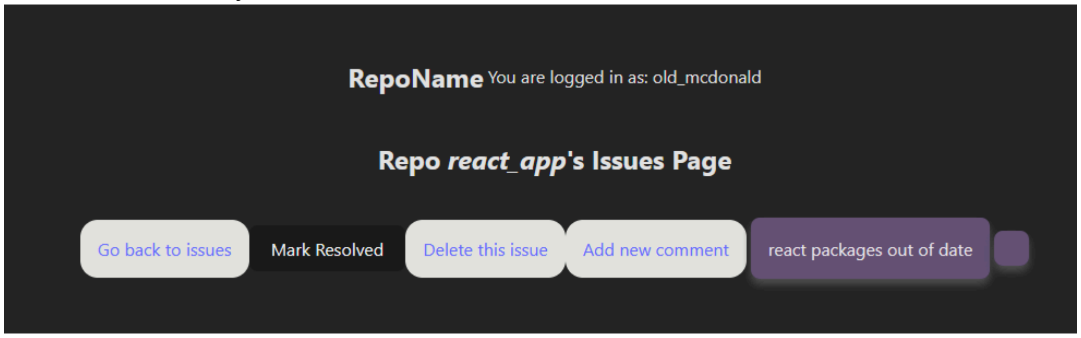
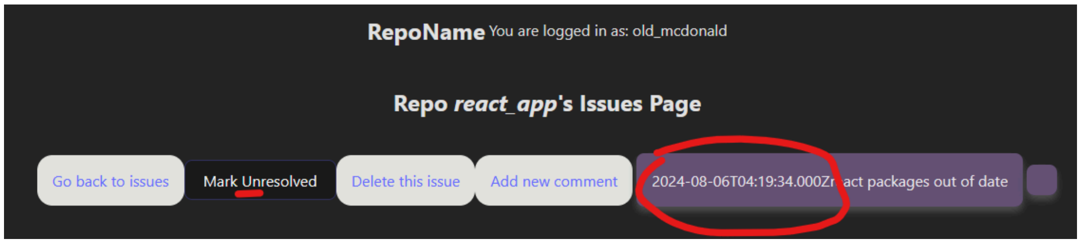
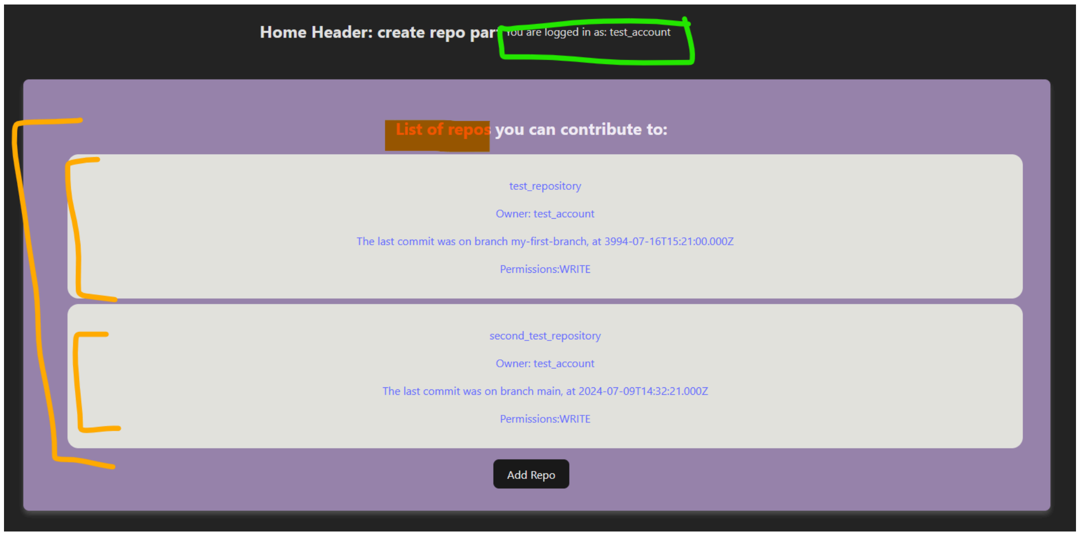
**Repo *react_app*'s Issues Page** | Select Unresolved | Select Resolved by date (ascending) | Select Resolved by date (descending)

Issues
- react packages out of date
- npm is not working
- yarn install stopped working

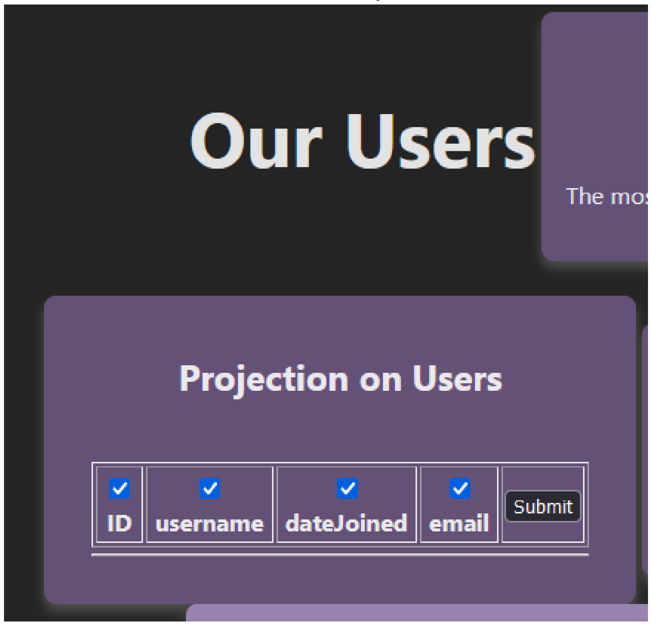
Go back to repo | Make new issue

**Queries: UPDATE Operation**

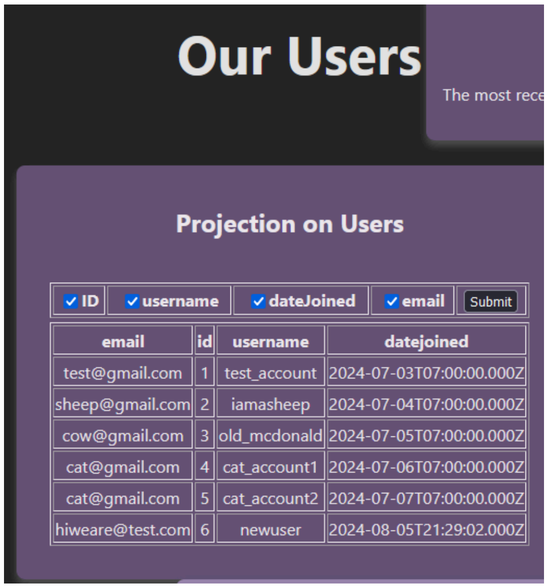
SQL Query can be found in: **backend/src/controllers/mainController.js**

| | |
|---|---|
| | Under the async function **setResolved(req, res)**<br>BEFORE: currently unresolved<br> RepoName You are logged in as: old_mcdonald<br>Repo *react_app*'s Issues Page<br>Go back to issues<br>Mark Resolved<br>Delete this issue<br>Add new comment<br>react packages out of date<br>AFTER:<br> RepoName You are logged in as: old_mcdonald<br>Repo *react_app*'s Issues Page<br>Go back to issues<br>Mark Unresolved<br>Delete this issue<br>Add new comment<br>2024-08-06T04:19:34.000Zreact packages out of date |
| **Queries: Selection** | SQL Query can be found in: **backend/src/controllers/mainController.js**<br>Under the async function **getRepos(req, res)**<br><br>BEFORE: n/a as query is called on rendering into HomePage<br><br>AFTER: shows the list of repositories belonging to the user you are logged in as (see green box which identifies which user you are)<br><br> Home Header: create repo par<br>You are logged in as: test_account<br>List of repos you can contribute to:<br>Add Repo<br>test_repository<br>Owner: test_account<br>The last commit was on branch my-first-branch, at 3994-07-16T15:21:00.000Z<br>Permissions:WRITE<br>second_test_repository<br>Owner: test_account<br>The last commit was on branch main, at 2024-07-09T14:32:21.000Z<br>Permissions:WRITE |
| **Queries: Projection** | SQL Query can be found in: **backend/src/controllers/userListController.js**<br>Under the async function **projectionPost(req, res)** |

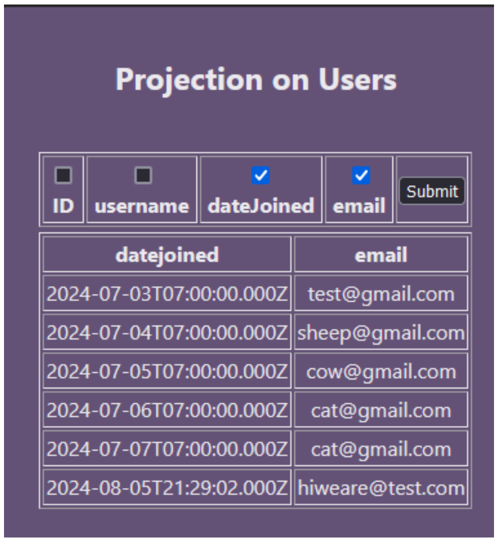
BEFORE: Submit button not pressed



AFTER: Columns which are checked will show up



| email | id | username | datejoined |
|---|---|---|---|
| test@gmail.com | 1 | test_account | 2024-07-03T07:00:00.000Z |
| sheep@gmail.com | 2 | iamasheep | 2024-07-04T07:00:00.000Z |
| cow@gmail.com | 3 | old_mcdonald | 2024-07-05T07:00:00.000Z |
| cat@gmail.com | 4 | cat_account1 | 2024-07-06T07:00:00.000Z |
| cat@gmail.com | 5 | cat_account2 | 2024-07-07T07:00:00.000Z |
| hiweare@test.com | 6 | newuser | 2024-08-05T21:29:02.000Z |



| datejoined | email |
|---|---|
| 2024-07-03T07:00:00.000Z | test@gmail.com |
| 2024-07-04T07:00:00.000Z | sheep@gmail.com |
| 2024-07-05T07:00:00.000Z | cow@gmail.com |
| 2024-07-06T07:00:00.000Z | cat@gmail.com |
| 2024-07-07T07:00:00.000Z | cat@gmail.com |
| 2024-08-05T21:29:02.000Z | hiweare@test.com |

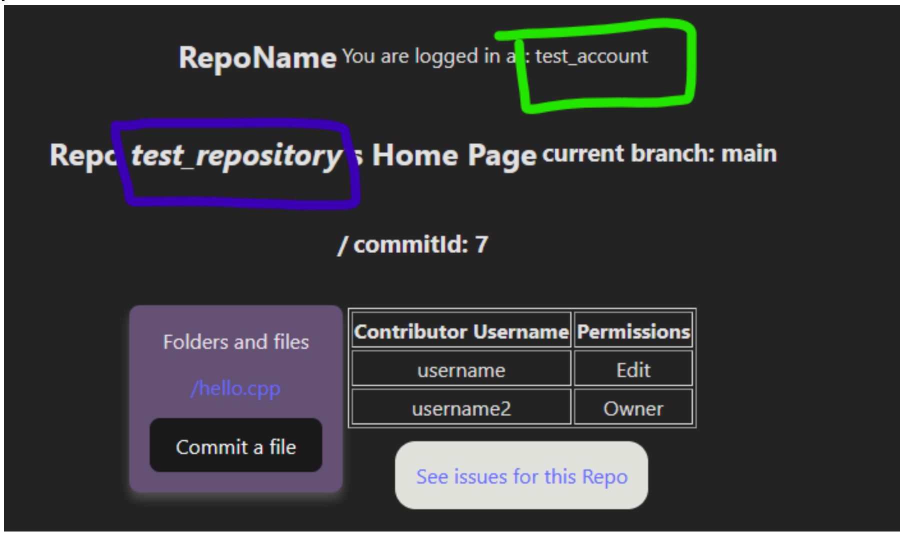
**Queries: Join**

SQL Query can be found in: **backend/src/controllers/fileController.js**
Under the async function **getFilesAndFolders(req, res)**

The join is on Users2 u2, UserContributesTo uc, Repo r, Permissions p, and Users1 u1.

BEFORE: n/a as query is called when user enters RepoPage for a specific repository

AFTER: the folders and files found in the repository name (blue box) where the permissions for the account is checked will show.

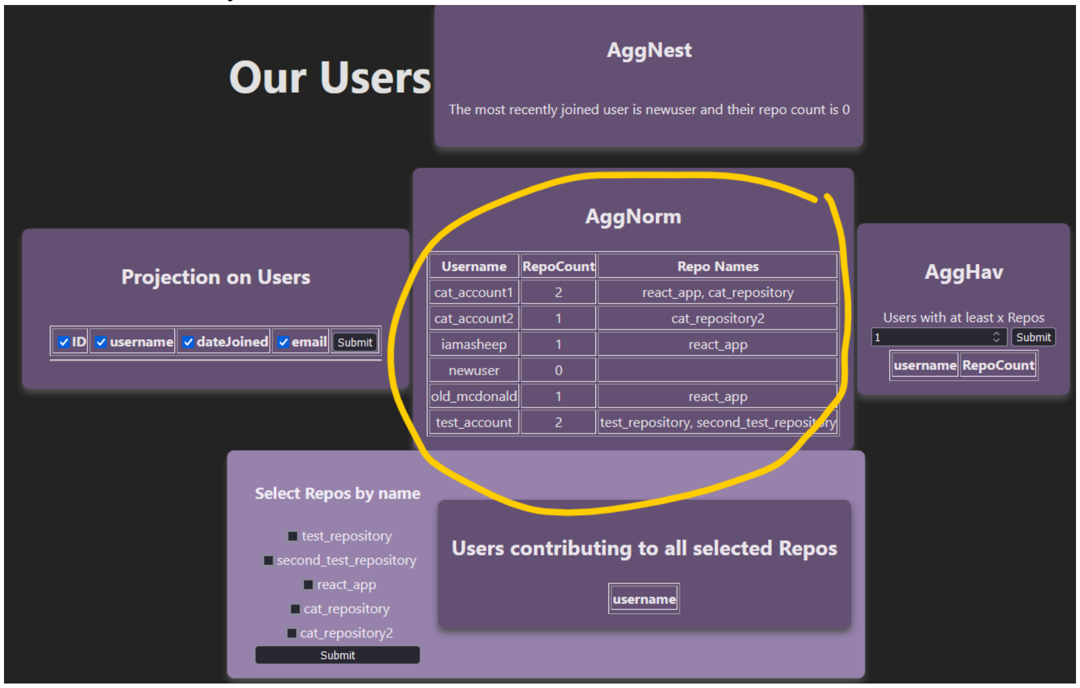

RepoName You are logged in as: test_account
Repo test_repository 's Home Page current branch: main
/ commitId: 7
Folders and files
/hello.cpp
Commit a file
See issues for this Repo

| Contributor Username | Permissions |
|---|---|
| username | Edit |
| username2 | Owner |

**Queries: Aggregation with Group By**

SQL Query can be found in: **backend/src/controllers/userListController.js**
Under the async function **query_AggNorm(req, res)**
BEFORE:n/a as it is called automatically on render

AFTER: note the yellow circle



The most recently joined user is newuser and their repo count is 0

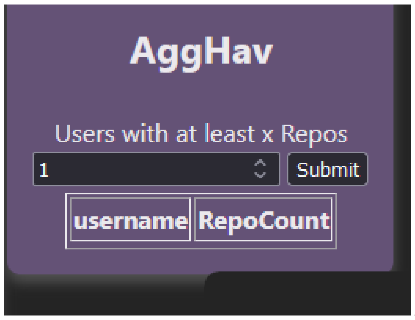
| Username | RepoCount | Repo Names |
|---|---|---|
| cat_account1 | 2 | react_app, cat_repository |
| cat_account2 | 1 | cat_repository2 |
| iamasheep | 1 | react_app |
| newuser | 0 | |
| old_mcdonald | 1 | react_app |
| test_account | 2 | test_repository, second_test_repository |

**Queries: Aggregation with**

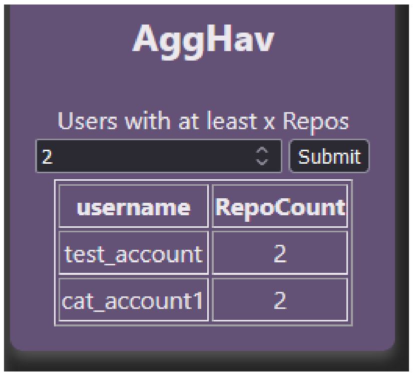
SQL Query can be found in: **backend/src/controllers/userListController.js**

**Having**

Under the async function **query_AggHav(req, res)**

BEFORE: 

| username | RepoCount |
|---|---|

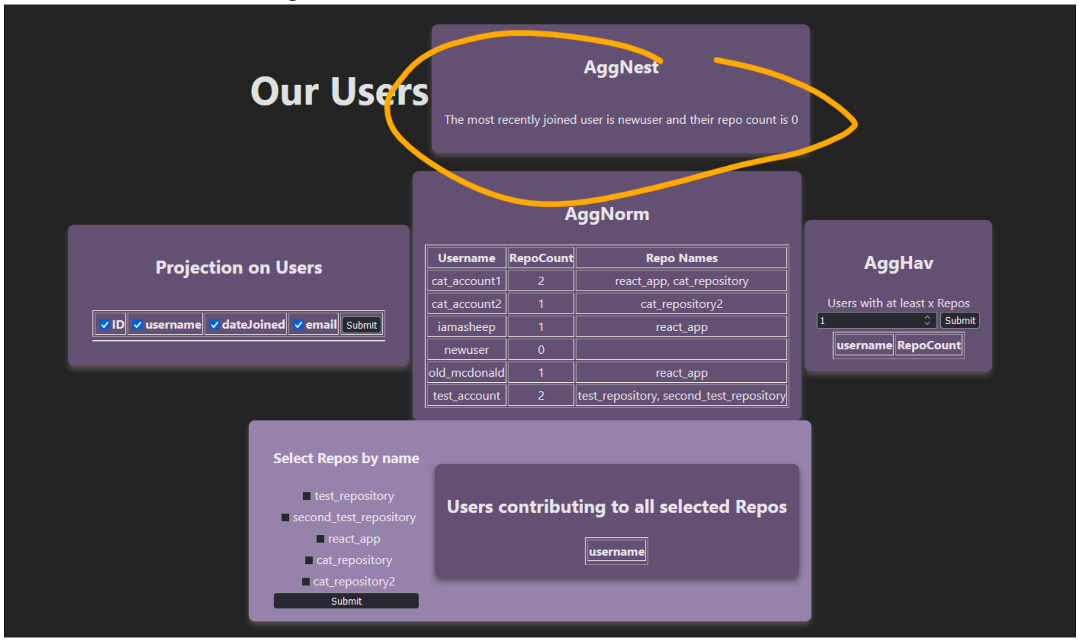
AFTER: 

| username | RepoCount |
|---|---|
| test_account | 2 |
| cat_account1 | 2 |

**Queries: Nested Aggregation with Group By**

SQL Query can be found in: **backend/src/controllers/userListController.js**
Under the async function **query_AggNest(req, res)**

BEFORE: n/a as it is called automatically on render
AFTER: note the orange circle



The most recently joined user is newuser and their repo count is 0

| Username | RepoCount | Repo Names |
|---|---|---|
| cat_account1 | 2 | react_app, cat_repository |
| cat_account2 | 1 | cat_repository2 |
| iamasheep | 1 | react_app |
| newuser | 0 | |
| old_mcdonald | 1 | react_app |
| test_account | 2 | test_repository, second_test_repository |

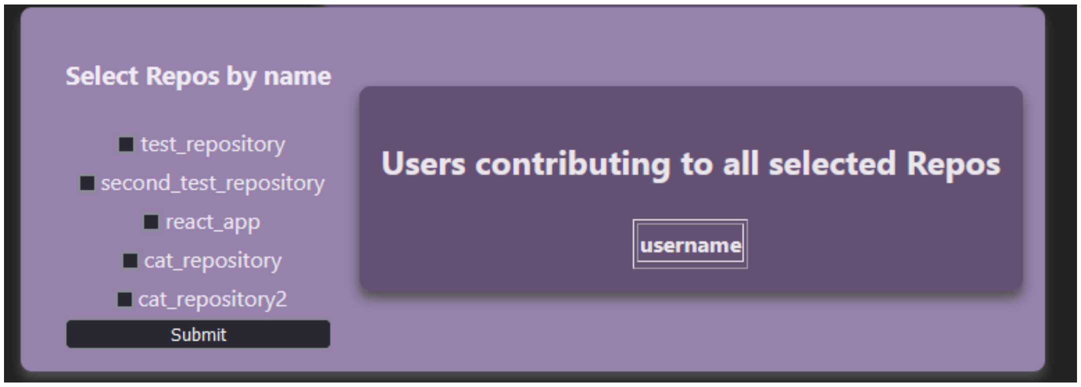
| username | RepoCount |
|---|---|

| username |
|---|

**Queries: Division**

SQL Query can be found in: **backend/src/controllers/userListController.js**
Under the async function **divisionPost(req, res)**

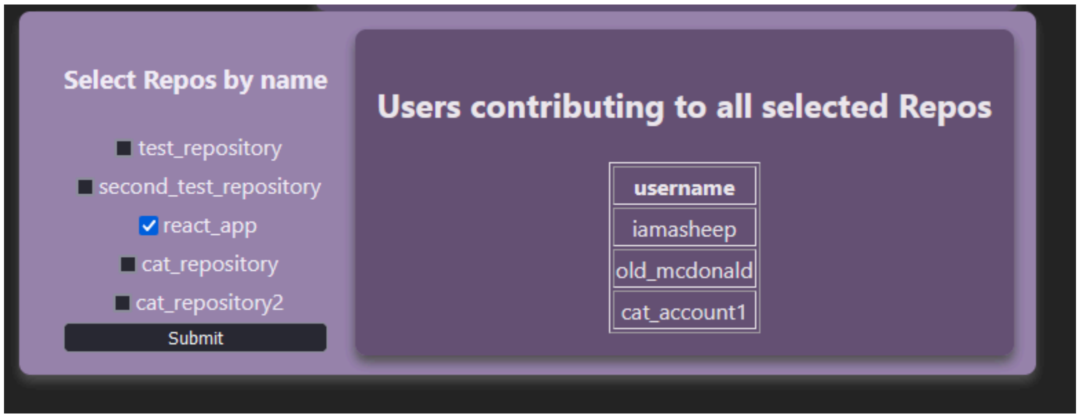
BEFORE:



Users contributing to all selected Repos

| username |
|---|

AFTER:



Users contributing to all selected Repos

| username |
|---|
| iamasheep |
| old_mcdonald |
| cat_account1 |