

Project Milestone- 2

Workload design

1. Queries that will drive application:

→ To register a new user:

```
INSERT INTO
  Users (creationDate, creationTime, username, firstName, lastName,
  gender, birthday, locationIP, browserUsed, locationCityId, speaks,
  email, passwordHash)
VALUES
  (CURRENT_DATE, CURRENT_TIME, $username, $firstName, $lastName,
  $gender, $birthday, $locationIP, $browserUsed, $locationCityId,
  $speaks, $email, $passwordHash);
```

→ To log in a user

```
SELECT * FROM Users WHERE username = 'username'
AND passwordHash = 'passwordHash';
```

→ To retrieve a user's friend's all posted recipes:

```
SELECT * FROM Recipe
WHERE CreatoruserId IN (
  SELECT user2id FROM User_knows_User WHERE user1id='userId'
  UNION
  SELECT user1id FROM User_knows_User WHERE user2id='userId'
);
```

→ Query to retrieve all comments on a recipe

```
SELECT * FROM comment WHERE ParentRecipeId = 'recipe_id'
```

→ Query to retrieve all like on a recipe

```
SELECT * FROM User_likes_Recipe WHERE RecipeId = 'recipe_id'
```

→ To comment on a recipe

```
INSERT INTO
  Comment (creationDate, creationTime, locationIP, browserUsed,
  Content, length, CreatorUserId, locationCountryId, ParentRecipeId)
```

```
VALUES
    (CURRENT_DATE, CURRENT_TIME, $locationIP, $browserUsed, $content,
     $length, $CreatorUserId, $locationCityId, $ParentRecipeId);
-- insert into Comment_hasTag_Tag if any
```

→ To delete his own comment from a recipe

```
DELETE FROM Comment_hasTag_Tag WHERE CommentId = $id;
DELETE FROM comments WHERE id = $id;
```

→ To like a recipe

```
INSERT INTO
    User_likes_Recipe (creationDate, creationTime, UserId, RecipeID)
VALUES
    (CURRENT_DATE, CURRENT_TIME, creationTime, $UserId, $RecipeID);
```

→ To unlike the recipe

```
DELETE FROM User_likes_Recipe WHERE UserId = $userid
and RecipeId = $recipeid
```

→ To delete any comment from his own recipe post

```
DELETE FROM comments WHERE id = $id and ParentRecipeId in (
    SELECT id FROM Recipe where CreatorUserId = $userid);
```

→ To search for a specific username or name

```
SELECT * FROM User WHERE username LIKE '%searchQuery%'
OR firstName LIKE '%searchQuery%' OR lastName LIKE '%searchQuery%';
```

→ To see suggested friends

```
SELECT u.id, COUNT(*) as num_common_interests
FROM User u
INNER JOIN User_hasInterest_Tag uhit1 ON uhit1.UserId = u.id
INNER JOIN User_hasInterest_Tag uhit2 ON uhit2.UserId =
specified_user_id AND uhit2.TagId = uhit1.TagId
LEFT JOIN User_knows_User uk ON uk.user1id = specified_user_id AND
```

```

uk.user2id = u.id
WHERE u.id != specified_user_id AND uk.user2id IS NULL
GROUP BY u.id
HAVING COUNT(*) >= 2
ORDER BY num_common_interests DESC
LIMIT 10;

```

→ To follow / unfollow a new user

```

-- To follow a new user:
INSERT INTO User_knows_User (creationDate, creationTime, userid,
user2id)
VALUES (CURRENT_DATE, CURRENT_TIME, 'userid', 'followedUserId');

-- To unfollow a user:
DELETE FROM User_knows_User WHERE userid = 'userid' and user2id =
'user2id'

```

→ To retrieve all the recipe posted by user

```

SELECT * FROM Recipe WHERE CreatorUserId ='specific_userid'

```

→ To update his own profile

```

UPDATE users SET email = 'example@example.com',
firstName = 'name1', lastName = 'name2' WHERE username ='john_doe';

```

→ To insert new recipe (when user posts new recipe)

```

INSERT INTO recipes (user_id, title, ingredients, directions)
VALUES (1, 'Spaghetti Carbonara', 'pasta, eggs, bacon, parmesan
cheese', 'Cook the pasta, fry the bacon, beat the eggs, mix
everything together.');
```

-- insert into Recipe_hasTag_Tag if any

2. Index choices for optimizing the queries:

- Index on the `id` column of the `users` table, as this will be used frequently for joining with other tables and retrieving user information.
- Index on the `username` column of the `users` table, as this will be used for retrieving user information and for authentication purposes.
- Index on the `creationDate` column of the `recipe` table, as this will be used for displaying the most recent posts and for sorting posts by date.
- Index on the `CreatorUserId` column of the `recipe` table, as this will be used for joining with the `users` table to retrieve user information for posts.
- Index on the `ParentRecipeId` column of the `comments` table, as this will be used for joining with the `posts` table to retrieve comments for a specific post.
- Index on the `userId` column of the `User_Knows_User` table, as this will be used for retrieving followers of a specific user.
- Index on the `recipeId` column of the `User_likes_Recipe` table, as this will be used for retrieving likes for a specific recipe post.

3. Database Size and performance:

- Database Size

COPY 1460	place
COPY 16080	tag
COPY 1739438	comment
COPY 1121226	recipe
COPY 10295	users
COPY 2176131	Comment_hasTag_Tag
COPY 751933	Recipe_hasTag_Tag
COPY 238052	User_hasInterest_Tag
COPY 1109813	User_likes_Comment
COPY 1435510	User_likes_Recipe
COPY 173014	User_knows_User

→ Performance

- error: permission denied (so used my local machine)

```
group_45⇒ EXPLAIN ANALYZE INSERT INTO
  Users (creationDate, creationTime, username, firstName, lastName, gender,
    birthday, locationIP, browserUsed, locationCityId, speaks, email, passwordHash)
VALUES
  (CURRENT_DATE, CURRENT_TIME, 'rohit23', 'Rohit', 'Kumar', 'male', '2002-02-23',
    '0.0.0.0', 'Chrome', 128, 'en', '23rohit@gmail.com', '0053448b90bd385ddd790743c3a4
    44c60a7b0f0b37730f673939c9221f43a35260d767288f37fca23b335fb42011aa17e3e5a06bcf29a4f97ae02de8b75313f9');
ERROR:  permission denied for sequence users_id_seq
group_45⇒
group_45⇒
```

➤ To register a new user

```
project45=# EXPLAIN ANALYZE INSERT INTO Users (creationDate, creationTime, username, firstName, lastName, gender,
  birthday, locationIP, browserUsed, locationCityId, speaks, email, passwordHash)
VALUES
  (CURRENT_DATE, CURRENT_TIME, 'rohit23', 'Rohit', 'Kumar', 'male', '2002-02-23',
    '0.0.0.0', 'Chrome', 128, 'en', '23rohit@gmail.com', '0053448b90bd385ddd790743c3a4
    44c60a7b0f0b37730f673939c9221f43a35260d767288f37fca23b335fb42011aa17e3e5a06bcf29a4f97ae02de8b75313f9');
QUERY PLAN

Insert on users  (cost=0.00..0.02 rows=1 width=320) (actual time=0.438..0.439 rows=0 loops=1)
→ Result (cost=0.00..0.02 rows=1 width=320) (actual time=0.124..0.125 rows=1 loops=1)
Planning Time: 0.134 ms
Execution Time: 0.685 ms
(4 rows)
```

➤ To log in a user

```
Index Scan using idx_users_username on users  (cost=0.29..8.30 rows=1 width
=8) (actual time=0.032..0.034 rows=1 loops=1)
Index Cond: ((username)::text = 'rohit23'::text)
Filter: ((passwordhash)::text = '0053448b90bd385ddd790743c3a4 44c60a7b0f0
b37730f673939c9221f43a35260d767288f37fca23b335fb42011aa17e3e5a06bcf29a4f97ae
02de8b75313f9'::text)
Planning Time: 0.112 ms
Execution Time: 0.058 ms
(5 rows)
```

➤ To retrieve a user's friends all posted recipe

```
QUERY PLAN
Nested Loop (cost=3492.71..12790.62 rows=8391 width=8) (actual time=12.698..14.267 rows=4311 loops=1)
  → HashAggregate (cost=3492.28..3492.73 rows=45 width=8) (actual time=12.686..12.694 rows=30 loops=1)
    Group Key: user_knows_user.user2id
    → Append (cost=0.42..3492.17 rows=45 width=8) (actual time=0.026..12.669 rows=30 loops=1)
      → Index Only Scan using idx_user_knows_user_userid on user_knows_user (cost=0.42..55.82 rows=23 width=8) (actual time=0.026..0.039 rows=30 loops=1)
        Index Cond: (userid = 16)
        Heap Fetches: 30
      → Seq Scan on user_knows_user user_knows_user_1 (cost=0.00..3435.68 rows=22 width=8) (actual time=12.623..12.623 rows=0 loops=1)
        Filter: (user2id = 16)
        Rows Removed by Filter: 173014
    → Index Scan using idx_recipe_creatoruserid on recipe (cost=0.43..204.75 rows=186 width=16) (actual time=0.005..0.040 rows=144 loops=30)
      Index Cond: (creatoruserid = user_knows_user.user2id)
Planning Time: 0.299 ms
Execution Time: 14.442 ms
(14 rows)
```

➤ Query to retrieve all comments on a recipe

```
project45=# EXPLAIN ANALYSE SELECT * FROM comment WHERE ParentRecipeId = 2199023255573;
QUERY PLAN
Index Scan using idx_comment_parentrecipeid on comment (cost=0.43..27.23 rows=6 width=113) (actual time=0.020..0.021 rows=0 loops=1)
  Index Cond: (parentrecipeid = '2199023255573'::bigint)
Planning Time: 0.091 ms
Execution Time: 0.043 ms
(4 rows)
```

➤ Query to retrieve all likes on a recipe

```
project45=# EXPLAIN ANALYSE SELECT * FROM User_likes_Recipe WHERE RecipeId = 2199023255573;
QUERY PLAN
Index Scan using idx_user_likes_recipe_recipeid on user_likes_recipe (cost=0.43..252.31 rows=65 width=28) (actual time=0.059..0.060 rows=0 loops=1)
  Index Cond: (recipeid = '2199023255573'::bigint)
Planning Time: 0.334 ms
Execution Time: 0.086 ms
(4 rows)
```

➤ Query to insert a new comment on a recipe by its id

```
project45=# EXPLAIN ANALYSE INSERT INTO
  Comment (creationDate, creationTime, locationIP, browserUsed, content,
length, CreatorUserId, locationCountryId, ParentRecipeId)
VALUES
  (CURRENT_DATE, CURRENT_TIME, '0.0.0.0', 'Chrome', 'awesome',
7, 16, 128, 2336467153684);
QUERY PLAN
Insert on comment (cost=0.00..0.02 rows=1 width=152) (actual time=0.376..0.377 rows=0 loops=1)
  → Result (cost=0.00..0.02 rows=1 width=152) (actual time=0.130..0.131 rows=1 loops=1)
Planning Time: 0.054 ms
Execution Time: 0.412 ms
(4 rows)
```

➤ To delete his own comment

```
project45=# EXPLAIN ANALYSE DELETE FROM comment WHERE id = 1;
               QUERY PLAN
-----
Delete on comment (cost=0.00..53795.65 rows=1 width=6) (actual time=122.600..122.601 rows=0 loops=1)
  → Seq Scan on comment (cost=0.00..53795.65 rows=1 width=6) (actual time=122.599..122.599 rows=0 loops=1)
    Filter: (id = 1)
    Rows Removed by Filter: 1739438
Planning Time: 0.077 ms
Execution Time: 122.628 ms
(6 rows)
```

➤ Query to insert a new like on a recipe by a user

```
project45=# EXPLAIN ANALYSE INSERT INTO
  User_likes_recipe (creationDate, creationTime, UserId, RecipeID)
VALUES
  (CURRENT_DATE, CURRENT_TIME, 16, 2336467153684);
               QUERY PLAN
-----
Insert on user_likes_recipe (cost=0.00..0.02 rows=1 width=28) (actual time=0.182..0.184 rows=0 loops=1)
  → Result (cost=0.00..0.02 rows=1 width=28) (actual time=0.006..0.007 rows=1 loops=1)
Planning Time: 0.044 ms
Execution Time: 0.213 ms
(4 rows)
```

➤ To unlike a recipe (deleting a like by its userid and recipeid)

```
project45=# EXPLAIN ANALYSE DELETE FROM User_likes_recipe WHERE
  UserId = 16 and RecipeId = 2336467153684;
               QUERY PLAN
-----
Delete on user_likes_recipe (cost=0.43..252.48 rows=1 width=6) (actual time=0.034..0.035 rows=0 loops=1)
  → Index Scan using idx_user_likes_recipe_recipeid on user_likes_recipe (cost=0.43..252.48 rows=1 width=6) (actual time=0.017..0.018 rows=1 loops=1)
    Index Cond: (recipeid = '2336467153684'::bigint) id and recipeid
    Filter: (userid = 16)
Planning Time: 0.099 ms
Execution Time: 0.079 ms
(6 rows)
```

➤ To search for a specific user using username of name

```
project45=# EXPLAIN ANALYSE SELECT id, username FROM Users WHERE username
  LIKE '%rohit%' OR firstName LIKE '%rohit%' OR lastName LIKE '%rohit%';
               QUERY PLAN
-----
Seq Scan on users (cost=0.00..648.16 rows=2 width=27) (actual time=5.788..5.788 rows=1 loops=1)
  Filter: (((username)::text ~ '%rohit%'::text) OR ((firstname)::text ~ '%rohit%'::text) OR ((lastname)::text ~ '%rohit%'::text))
  Rows Removed by Filter: 10295
Planning Time: 0.228 ms
Execution Time: 5.808 ms
(5 rows)
```


- To see suggested friends based on a shared interest (at least 2)

```
QUERY PLAN
Limit (cost=12192.16..12192.18 rows=10 width=16) (actual time=50.564..53.159 rows=10 loops=1)
  → Sort (cost=12192.16..12205.73 rows=5428 width=16) (actual time=50.562..53.157 rows=10 loops=1)
    Sort Key: (count(*)) DESC
    Sort Method: top-N heapsort  Memory: 25kB
    → Finalize HashAggregate (cost=12020.58..12074.86 rows=5428 width=16) (actual time=50.514..53.137 rows=95 loops=1)
      Group Key: u.id
      → Gather (cost=11653.38..12004.61 rows=3193 width=16) (actual time=49.895..53.078 rows=95 loops=1)
        Workers Planned: 1
        Workers Launched: 1
        → Partial HashAggregate (cost=10653.38..10685.31 rows=3193 width=16) (actual time=47.236..47.249 rows=48 loops=2)
          Group Key: u.id
          → Hash Anti Join (cost=6367.16..10637.42 rows=3193 width=8) (actual time=42.684..47.213 rows=48 loops=2)
            Hash Cond: (u.id = uk.user2id)
            → Hash Join (cost=6311.06..10540.79 rows=3200 width=8) (actual time=42.641..47.163 rows=52 loops=2)
              Hash Cond: (uhit1.userid = u.id)
              → Parallel Hash Join (cost=5585.69..9807.02 rows=3200 width=8) (actual time=39.123..43.621 rows=52 loops=2)
                Hash Cond: (uhit2.tagid = uhit1.tagid)
                → Parallel Seq Scan on user_hasinterest_tag uhit2 (cost=0.00..3501.38 rows=18 width=8) (actual time=3.128..6.233 rows=0 loops=2)
                  Filter: (userid = 16)
                  Rows Removed by Filter: 119026
                → Parallel Hash (cost=3151.31..3151.31 rows=140031 width=16) (actual time=27.656..27.657 rows=119026 loops=2)
                  Buckets: 131072  Batches: 4  Memory Usage: 4032kB
                  → Parallel Seq Scan on user_hasinterest_tag uhit1 (cost=0.00..3151.31 rows=140031 width=16) (actual time=0.008..11.534 rows=119026 loops=2)
                    Buckets: 16384  Batches: 1  Memory Usage: 531kB
                    → Seq Scan on users u (cost=0.00..596.69 rows=10294 width=8) (actual time=0.011..2.070 rows=10295 loops=2)
                      Filter: (id <> 16)
                      Rows Removed by Filter: 1
          → Hash (cost=55.82..55.82 rows=23 width=8) (actual time=0.063..0.064 rows=30 loops=1)
            Buckets: 1024  Batches: 1  Memory Usage: 10kB
            → Index Only Scan using idx_user_knows_user_userid on user_knows_user uk (cost=0.42..55.82 rows=23 width=8) (actual time=0.053..0.060 rows=30 loops=1)
              Index Cond: (userid = 16)
              Heap Fetches: 30
Planning Time: 0.833 ms
Execution Time: 53.286 ms
(35 rows)
```

- To follow a new user

```
project45=# EXPLAIN ANALYSE INSERT INTO User_knows_User
(creationDate, creationTime,userid, user2id) VALUES (CURRENT_DATE, CURRENT_TIME, 2, 16);
QUERY PLAN

Insert on user_knows_user (cost=0.00..0.02 rows=1 width=28) (actual time=0.130..0.130 rows=0 loops=1)
  → Result (cost=0.00..0.02 rows=1 width=28) (actual time=0.006..0.006 rows=1 loops=1)
Planning Time: 0.045 ms
Execution Time: 0.162 ms
(4 rows)
```

- To unfollow a user

```
project45=# EXPLAIN ANALYSE DELETE FROM User_knows_User WHERE (userid = 1 AND user2id = 16) OR (userid=16 AND user2id=1);
QUERY PLAN

Delete on user_knows_user (cost=8.86..12.88 rows=1 width=6) (actual time=0.028..0.030 rows=0 loops=1)
  → Bitmap Heap Scan on user_knows_user (cost=8.86..12.88 rows=1 width=6) (actual time=0.027..0.028 rows=0 loops=1)
    Recheck Cond: (((userid = 1) AND (user2id = 16)) OR ((userid = 16) AND (user2id = 1)))
    → BitmapOr (cost=8.86..8.86 rows=1 width=0) (actual time=0.025..0.026 rows=0 loops=1)
      → Bitmap Index Scan on idx_user_knows_user_userid (cost=0.00..4.43 rows=1 width=0) (actual time=0.020..0.020 rows=0 loops=1)
        Index Cond: ((userid = 1) AND (user2id = 16))
      → Bitmap Index Scan on idx_user_knows_user_userid (cost=0.00..4.43 rows=1 width=0) (actual time=0.004..0.004 rows=0 loops=1)
        Index Cond: ((userid = 16) AND (user2id = 1))
Planning Time: 0.147 ms
Execution Time: 0.071 ms
(10 rows)
```


- To retrieve all the recipe posted by a specific user

```
project45=# EXPLAIN ANALYSE SELECT * FROM Recipe WHERE CreatorUserId = 283;
              QUERY PLAN
-----
Index Scan using idx_recipe_creatoruserid on recipe (cost=0.43..170.46 rows=177 width=579) (actual time=0.027..0.142 rows=329 loops=1)
  Index Cond: (creatoruserid = 283)
Planning Time: 0.110 ms
Execution Time: 0.182 ms
(4 rows)
```

- Query to update a user's information by their username or id

```
project45=# EXPLAIN ANALYSE UPDATE users SET email = 'rohit@gamil.com',
firstName = 'Rohit', lastName = 'Kumar' WHERE username = 'rohit23';
              QUERY PLAN
-----
Update on users (cost=0.29..8.30 rows=1 width=319) (actual time=0.067..0.067 rows=0 loops=1)
  → Index Scan using idx_users_username on users (cost=0.29..8.30 rows=1 width=319) (actual time=0.036..0.038 rows=1 loops=1)
    Index Cond: ((username)::text = 'rohit23'::text)
Planning Time: 0.116 ms
Execution Time: 0.109 ms
(5 rows)
```

- To insert new recipe

```
project45=# EXPLAIN ANALYSE INSERT INTO recipe (creationDate, creationTime, title, locationIP, browserUsed, instruction, length, CreatorUserId, locationCountryId)
VALUES (CURRENT_DATE, CURRENT_TIME, 'Spaghetti Carbonara', '0.0.0.0', 'Chrome', 'Cook the pasta, fry the bacon, beat the eggs, mix everything together.', 60, 2, 0);
              QUERY PLAN
-----
Insert on recipe (cost=0.00..0.02 rows=1 width=232) (actual time=0.254..0.255 rows=0 loops=1)
  → Result (cost=0.00..0.02 rows=1 width=232) (actual time=0.102..0.102 rows=1 loops=1)
Planning Time: 0.047 ms
Execution Time: 0.288 ms
(4 rows)
```