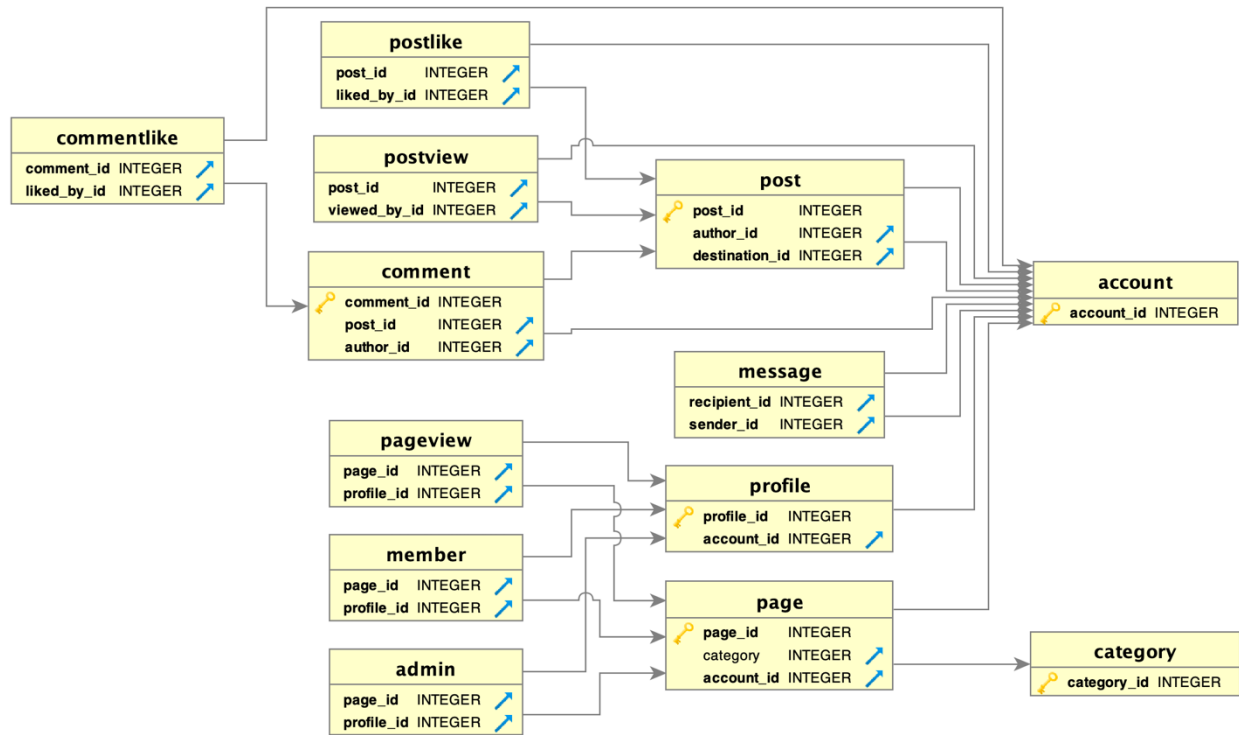


Assumptions:

- A Page and a Profile are accounts and therefore have an account id which is an integer that increments by 1 each time a new account is created starting from 0.
- An account can create a post, and each post will have a unique int id which starts from 3000 and is incremented by 1 every time a new post is created.
- A profile will also have a unique int id which starts from 1000 and is incremented by 1 every time a new profile is created in addition to having a unique account id.
- A post receives views and within each view the account id of the viewer and the time it is viewed is recorded.
- A post which can be either a text, audio, video, or image contains comments and each comment has a unique int id which starts from 4000 and increments every time a comment in a post it is contained by is added. Only the account can create their post.
- A comment can receive a like. When that happens the comment id, the account id of the person who liked it and the time are recorded.
- An account can send and receive messages. The time sent, the recipient account id, and the sender account id act as the primary keys.
- A Page can be categorized and each category has a unique int id and has a string that corresponds to that id to decrease redundancy. A page must have a category
- I used PostgreSQL on the Google Cloud Platform as my DBMS
- All queries are made from the backend sever which runs on nodejs and is deployed on Heroku
- Web app: <https://cse3330-221718.firebaseio.com>
- Code Repo: <https://github.com/ondikumana/CSE3330>

Relations



Tables

```
CREATE SEQUENCE account_id_seq start 38 increment 1; --Usually starts from 1
```

```
CREATE TABLE Account (  
  account_id INT NOT NULL DEFAULT nextval('account_id_seq') PRIMARY KEY,  
  creation_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  account_type VARCHAR(7) NOT NULL  
);
```

```
CREATE SEQUENCE category_id_seq start 25 increment 1; --Usually starts from 20
```

```
CREATE TABLE Category (  
  category_id INT NOT NULL DEFAULT nextval('category_id_seq') PRIMARY KEY,  
  category_description VARCHAR(100) UNIQUE NOT NULL  
);
```

```
CREATE SEQUENCE page_id_seq start 2004 increment 1;--Usually starts from 2000
```

```
CREATE TABLE Page (  
  page_id INT NOT NULL DEFAULT nextval('page_id_seq') PRIMARY KEY,  
  page_name VARCHAR(100) NOT NULL,  
  logo_url VARCHAR(300),  
  header_image_url VARCHAR(300),  
  description VARCHAR(2000),  
  category INT,  
  account_id INT NOT NULL  
);
```

```
ALTER TABLE Page ADD FOREIGN KEY (category) REFERENCES Category(category_id) ON DELETE  
SET NULL;
```

```
ALTER TABLE Page ADD FOREIGN KEY (account_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;
```

```
CREATE SEQUENCE profile_id_seq start 1021 increment 1; --Usually starts from 1000
```

```
CREATE TABLE Profile (  
  profile_id INT NOT NULL DEFAULT nextval('profile_id_seq') PRIMARY KEY,  
  fname VARCHAR(100) NOT NULL,  
  lname VARCHAR(100) NOT NULL,  
  phone VARCHAR(12) NOT NULL,  
  email VARCHAR(100) NOT NULL UNIQUE,  
  username VARCHAR(12) NOT NULL UNIQUE,  
  password VARCHAR(12) NOT NULL,  
  account_id INT NOT NULL  
);
```

```
ALTER TABLE Profile ADD FOREIGN KEY (account_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;
```

```
CREATE TABLE Member (  
page_id INT NOT NULL,  
profile_id INT NOT NULL ,  
join_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
ALTER TABLE Member ADD CONSTRAINT unique_member UNIQUE(page_id, profile_id);
```

```
ALTER TABLE Member ADD FOREIGN KEY (page_id) REFERENCES Page(page_id) ON DELETE  
CASCADE;
```

```
ALTER TABLE Member ADD FOREIGN KEY (profile_id) REFERENCES Profile(profile_id) ON DELETE  
CASCADE;
```

```
CREATE TABLE PageView (  
page_id INT NOT NULL,  
profile_id INT NOT NULL,  
time TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
ALTER TABLE PageView ADD CONSTRAINT unique_pageview UNIQUE(page_id, profile_id);
```

```
ALTER TABLE PageView ADD FOREIGN KEY (page_id) REFERENCES Page(page_id) ON DELETE  
CASCADE;
```

```
ALTER TABLE PageView ADD FOREIGN KEY (profile_id) REFERENCES Profile(profile_id) ON  
DELETE CASCADE;
```

```
CREATE TABLE Admin (  
page_id INT NOT NULL,  
profile_id INT NOT NULL,  
admin_since_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
ALTER TABLE Admin ADD CONSTRAINT unique_admin UNIQUE(page_id, profile_id);
```

```
ALTER TABLE Admin ADD FOREIGN KEY (page_id) REFERENCES Page(page_id) ON DELETE  
CASCADE;
```

```
ALTER TABLE Admin ADD FOREIGN KEY (profile_id) REFERENCES Profile(profile_id) ON DELETE  
CASCADE;
```

```
CREATE TABLE Message (  
recipient_id INT NOT NULL,  
sender_id INT NOT NULL,  
sent_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
was_sent BOOLEAN,
```

```
was_delivered BOOLEAN,  
was_read BOOLEAN,  
body VARCHAR(2000) NOT NULL  
);
```

```
ALTER TABLE Message ADD FOREIGN KEY (recipient_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;  
ALTER TABLE Message ADD FOREIGN KEY (sender_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;
```

```
CREATE SEQUENCE post_id_seq start 3013 increment 1; --Usually starts from 3000
```

```
CREATE TABLE Post (  
post_id INT NOT NULL DEFAULT nextval('post_id_seq') PRIMARY KEY,  
time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
author_id INT NOT NULL,  
destination_id INT NOT NULL,  
attachment_url VARCHAR(200),  
body VARCHAR(2000) NOT NULL  
);
```

```
ALTER TABLE Post ADD FOREIGN KEY (author_id) REFERENCES Account(account_id) ON DELETE  
CASCADE;  
ALTER TABLE Post ADD FOREIGN KEY (destination_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;
```

```
CREATE TABLE PostLike (  
post_id INT NOT NULL,  
liked_by_id INT NOT NULL,  
time TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
ALTER TABLE PostLike ADD FOREIGN KEY (post_id) REFERENCES Post(post_id) ON DELETE  
CASCADE;  
ALTER TABLE PostLike ADD FOREIGN KEY (liked_by_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;
```

```
ALTER TABLE PostLike ADD CONSTRAINT unique_postlike UNIQUE(post_id, liked_by_id);
```

```
CREATE TABLE PostView (  
post_id INT NOT NULL,  
viewed_by_id INT NOT NULL,  
time TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
ALTER TABLE PostView ADD FOREIGN KEY (post_id) REFERENCES Post(post_id) ON DELETE  
CASCADE;
```

```
ALTER TABLE PostView ADD FOREIGN KEY (viewed_by_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;
```

```
ALTER TABLE PostView ADD CONSTRAINT unique_postview UNIQUE(post_id, viewed_by_id);
```

```
CREATE SEQUENCE comment_id_seq start 4014 increment 1; --Usually starts from 4000
```

```
CREATE TABLE Comment (  
comment_id INT NOT NULL DEFAULT nextval('comment_id_seq') PRIMARY KEY,  
post_id INT NOT NULL,  
time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
author_id INT NOT NULL,  
attachment_url VARCHAR(200),  
body VARCHAR(2000) NOT NULL  
);
```

```
ALTER TABLE Comment ADD FOREIGN KEY (post_id) REFERENCES Post(post_id) ON DELETE  
CASCADE;
```

```
ALTER TABLE Comment ADD FOREIGN KEY (author_id) REFERENCES Account(account_id) ON  
DELETE CASCADE;
```

```
CREATE TABLE CommentLike (  
comment_id INT NOT NULL,  
liked_by_id INT NOT NULL,  
time TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
ALTER TABLE CommentLike ADD FOREIGN KEY (comment_id) REFERENCES  
Comment(comment_id) ON DELETE CASCADE;
```

```
ALTER TABLE CommentLike ADD FOREIGN KEY (liked_by_id) REFERENCES Account(account_id)  
ON DELETE CASCADE;
```

```
ALTER TABLE CommentLike ADD CONSTRAINT unique_commentlike UNIQUE(comment_id,  
liked_by_id);
```

