

# CS411 Team 2: PrimeTime

## Data Documentation:

In this project, we are using MySQL as our database. We have only one scheme which is name by PrimeTime. Currently we having following tables:

### 1. **prime\_time\_members:**

- a) uid { INT(10) , Primary Key }
- b) username { VARCHAR(30) }
- c) password { VARCHAR(45) }
- d) email { TEXT }
- e) salt { VARCHAR(6) }

This table is mainly for storing user login credittials. Passwords are hashed using MD5 along with their salt. It is not super secure, we may start using Bcrypt. uid is the primary key which connect this table with all other tables.

### 2. **prime\_time\_member\_profile:**

- a) uid { INT(10), Primary Key }
- b) username { VARCHAR(45) }
- c) ins\_username { VARCHAR(45) }
- d) ins\_password { TEXT }
- e) ins\_cached\_time { TIMESTAMP }
- f) avatar { VARCHAR(45) }

This table saves up user profile information which stands for their Instagram Credential Info. Also we store cached\_time in here to determine if we need to refetch their Instagram Media using our InstagramScraper program.

### 3. **prime\_time\_session:**

- a) uid {INT(10), Primary Key}
- b) sid { VARCHAR(45) } (SessionId)
- c) token { TEXT }
- d) lastlogin { TIMESTAMP }
- e) expiretime { TIMESTAMP }

This table is used to store user authentication information. When user login they are assigned to a SessionId and Token value (Created by hashed value of sessionId , uid, and current timestamp). Each time user make a request, they will need to validate their token value before the request gets proceed.

### 4. **prime\_time\_user\_media:**

- a) mid { VARCHAR(30) , Primary Key }

- b) uid { INT(10) }
- c) created { Timestamp }

This table is storing connect between user and their cached medias. Here Mediald is the primary key for avoiding duplicate medias.

#### **5~14. Instagram\_cache\_0~9:**

- a) id { VARCHAR(30), Primary Key }
- b) shortcode { VARCHAR(45) }
- c) created { TIMESTAMP }
- d) num\_comment { INT(10) }
- e) num\_like { INT(10) }
- f) image\_link { TEXT }
- g) media\_type { VARCHAR(10) }
- h) locationId { INT(10) }
- i) expect\_like { INT(10) }

There are 10 tables, they all used for caching media posts from users. All posts are distributed into groups by their last digit of their mediald.

## **How We Utilizing Caching:**

Since it is no way we know if user has post a new media on Instagram unless we make a API call and check. So for caching medias for our model training and also for releasing the pressure of our Scraper (It has a limitation of about 3000 requests per 10 minutes. ) , we record the last cached time for each user. If the time is beyond a day. We fetch medias directly from our database. Also user can request a refresh on their own.

## Caching Sequence Diagram:

