67-262: Database Design and Development

Phase 2 Full Write-up

T15

by Xavier Xia (boxia) and Gabi Augustin (gaugusti)

December 12th, 2021

Professor Raja Sooriamurthi, Professor Xiaoying Tu

User Definitions

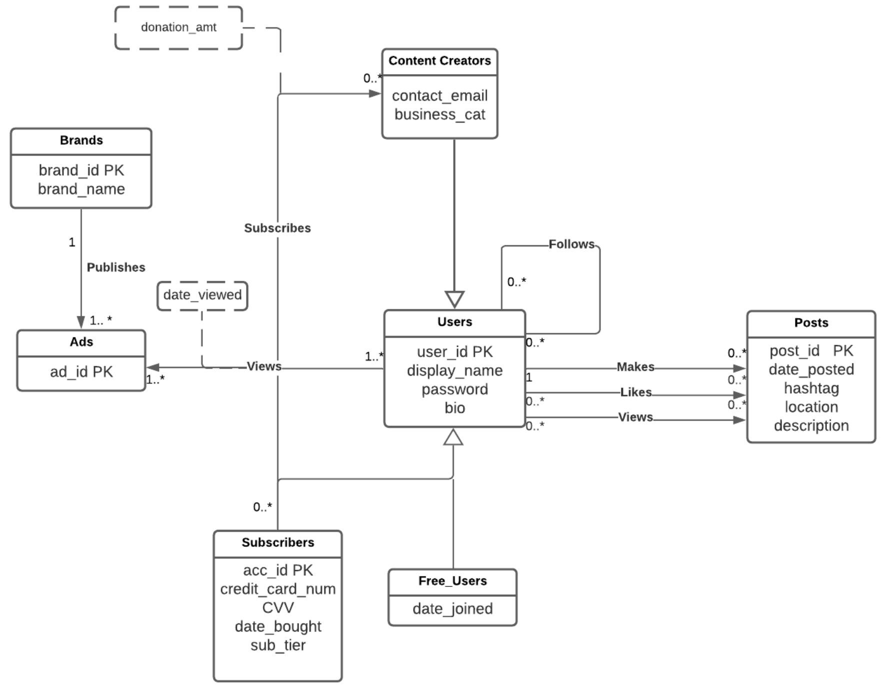
- **(Individual) User** → A person who uses the Instagram app to interact with or post their own content such as photos, stories/highlights, and/or reels. They can create their own profile, follow others (such as influencers or those within their social circle)
- **Content Creator** → A person who creates content and uses Instagram as a place to promote their work. The content can range from art, fashion, videos/film, dance, and more, so long as it can be captured visually.
- **Brands** → An outside company that wants to promote their products to users. Their goal is to get more traffic and attention on their ads. They do not have associated instagram accounts.

User Stories

ID	type	As a <role>	I want <goal>	so that <reason>
US1	simple	As a user	I want to view posts from my followers	so that I can see how my followers are doing and interact with their posted content.
US2	analytical	As a content creator	I want to see my most viewed post	so that I can see what type of content is most popular with my followers
US3	simple	As a content creator	I want to see my post’s total likes	so that I can see the popularity of one post.
US4	simple	As a user	I want to see all posts in a specific hashtag	so that I can find content similar to what I’ve enjoyed previously.

US5	complex	As a content creator	I want to see which post of mine under a certain hashtag has the most likes.	so that I can track interactions across content types.
US6	complex	As a user	I want to know what influencers the people I follow are following	so that I can keep updated on their interests and possibly find new influencers and artists to follow.
US7	complex	As a brand	I want to know how many users view my ad every month	so that I can check on the success of my products and how many people it reaches.
US8	complex	As a content creator	I want to which posts users are seeing and when	so that I can track the popularity of my content and keep updated on good times to post content.
US9	analytical	As a brand	I want to know the rankings of engagement rate (num_views) of my advertisements	so that I can see which advertisements are the most successful and model future ones after them.
US10 (New)	analytical	As a user,	I want to increase my appreciation for a content creator by increasing my donation (shown through sub_tier)	so that I can support the artist actively through different subscription tiers.

Conceptual Model:



**Clear arrows denote inheritance*

Assumptions:

a) “Following” is denoted by mutual following between two users.

b) A post can only be created by one user.

c) A user cannot see the same ad more than once in a single day.

d) All brands have posted at least one ad.

e) A user can have zero posts, followers, and follow nobody.

f) A content creator can have no followers, follow nobody, and have no subscribers.

g) A subscriber has to subscribe to at least one content creator.

h) A user can view a post multiple times, but it is only counted once per day.

i) All premium posts are created by a sole content creator.

j) Each credit card can only be used by one user and one acc_id.

k) Each subscriber must have a positive (non-zero) donation amount to begin with.

l) Each post can only have one hashtag denoting the type of content

Relational Model

bold: PK’s

underlines: FK’s

- Subsume Reasons

Brands (**brand_id**, brand_name)

Ads (**ad_id**, brand_id)

- Ads subsumes brand_id because a specific brand publishes each ad to Instagram.

Users (**user_id,** display_name, password, bio)

Subscribers (**user_id, acc_id**, credit_card_num, CVV, date_bought, sub_tier)

Free_Users (**user_id,** date_joined)

Content_Creators (**user_id**, contact_email, business_cat)

Posts (**post_id, user_id,** date_posted, hashtag, location, description)

- Post subsumes user_id to indicate that a sole user has created a certain post.

Follows (**u1.user_id, u2.user_id**)

Subscribes (**u1.user_id, u2.user_id,** donation_amt)

Views_Ad (**user_id, ad_id, date_viewed**)

Views_Post (**post_id, user_id**, date_viewed **, view_by**)

Likes_Post (**post_id, user_id**, date_liked, **liked_by**)

Functional Dependencies

Relation	**Functional Dependencies**	**Form**
Brands (**brand_id**, brand_name)	brand_id → brand_name	BCNF
Ads (**ad_id**, brand_id)	ad_id → brand_id	BCNF
Users (**user_id**, display_name, password, bio)	user_id → display_name, password, bio, date_joined	BCNF
Subscribers (**user_id,** acc_id, credit_card_num, CVV, date_bought, sub_tier)	user_id → credit_card_num, date_bought, sub_tier acc_id → date_bought, sub_tier, credit_card_num credit_card_num → CVV	2NF
Free_Users (**user_id,** date_joined)	user_id → date_joined	BCNF
Posts (**post_id, user_id,** date_posted, hashtag, location, description)	post_id, user_id → date_posted, hashtag, location, description	BCNF
Content_Creators (**user_id**, contact_email, business_cat)	user_id → contact_email, business_cat	BCNF

No Functional Dependencies: The following dependencies do not have any non-key attributes and therefore have no functional dependencies. They are all in the form: BCNF.

- Follows (**u2.user_id, u2.user_id**)
- Subscribes (**u2.user_id, user_id,** donation_amt)
- Views_Ad (**user_id, ad_id, date_viewed**)
- Views_Post (**post_id, user_id**, **date_viewed**)
- Likes_Post (**post_id, user_id**, date_liked)

Notes:

- Post_ids are unique to individuals, but not to *every* individual. For example, Both User1 and User2 can have a post of post_id=1, despite having different user_ids. These IDs are generated through posting order and to gather information on a specific post, both the user_id and post_id must be known.
- While an Ad and Post can be viewed multiple times, it is only counted as being seen once per day.
- Posts can only be liked once.

Normalization

Brands (**brand_id**, brand_name)

- Functional Dependencies:

  brand_id → brand_name
- Form: BCNF
- Justification: A brand's company name can be found if its unique ID is known. There are no existing partial or transitive dependencies.

Ads (**ad_id**, brand_id)

- Functional Dependencies:

  ad_id → brand_id
- Form: BCNF
- Justification: Having the unique ID of an advertisement gives you the Brand that has created it given that each advertisement is made by a sole company. There are no existing partial or transitive dependencies.

Users (**user_id,** display_name, password, bio)

- Functional Dependencies:

  user_id → display_name, password, bio
- Form: BCNF
- Justification: Having a user's user_id/username gives you access to the information set inside of their profile.

Subscribers (**user_id,** acc_id, credit_card_num, CVV, date_bought, sub_tier)

- Functional Dependencies:

  user_id → credit_card_num, date_bought, sub_tier

  acc_id → date_bought, sub_tier, credit_card_num

  credit_card_num → CV
- Form: 3NF
- Justification: There is a transitive dependency between credit_card_num and CVV. There are partial dependencies. To move into BCNF, decompose as follows:
  - R1: Subscribers (**user_id,** acc_id)
  - R2: Credit_Cards(**credit_card_num**, CVV)

    - R3: Accounts(**acc_id**, credit_card_num, date_bought, sub_tier)

Free_Users (**user_id,** date_joined)

- Functional Dependencies:

  user_id → date_joined
- Form: BCNF
- Justification: Having a user's id gives you access to the date they joined Instagram as it was registered at sign-up.

Posts (**post_id, user_id,** date_posted, hashtag, location, description)

- Functional Dependencies:

  post_id, user_id → date_posted, hashtag, location, description
- Form: BCNF
- Justification: Post_ids are created incrementally as a user posts and correspond to the order in which they are posted. (i.e., User1 and User2 can both have posts of id=1,2,3,4, etc.,) When post_id is paired with user_id it then uniquely identifies all information contained with a post such as when it was posted, the hashtag associated, where it was location, and its description. There are no transitive or partial dependencies.

Content_Creators (**user_id**, contact_email, business_cat)

- Functional Dependencies:

  user_id → f_name, l_name
- Form: BCNF
- Justification: Having an individual Content Creator's user_id gives you their first and last name. It is impossible to find a creator's full name based on either just the first name or the last name and vice versa. There are no transitive or partial dependencies.

Follows (**u1.user_id, u2.user_id**)

- Form: BCNF
- Justification: This relation does not have any FDs so it cannot be further decomposed as none or bad. There are no non-key attributes.

Subscribes (**u1.user_id, u2.user_id**)

- Form: BCNF

- Justification: This relation does not have any FDs so it cannot be further decomposed as none or bad. There are no non-key attributes.

Views_Ad (**user_id, ad_id, date_viewed**)

- Form: BCNF
- Justification: This relation does not have any FDs so it cannot be further decomposed as none or bad. There are no non-key attributes.

Views_Post (**post_id, user_id**, **date_viewed ,** **view_by**)

- Form: BCNF
- Justification: This relation does not have any FDs so it cannot be further decomposed as none or bad. There are no non-key attributes.

Likes_Post (**post_id, user_id**, **date_liked**, **liked_by**)

- Form: BCNF
- Justification: This relation does not have any FDs so it cannot be further decomposed as none or bad. There are no non-key attributes.