Tables description of assignment 7\_01

Table of Contents

[Functions (at a high level) that are supported 2](#_Toc19556136)

[ER diagram for stored data 2](#_Toc19556137)

[Table design for stored data 2](#_Toc19556138)

[The sequence diagram of third-party login 12](#_Toc19556139)

**Functions (at a high level) that are supported**

On logging in user will see newsfeed which contains posts from groups and users'

wall whom you follow sorted by time.

You can also see walls of people whom you follow.

You can also see wall of a group which you follow.

You can like a post and also comment on it.

You can also reply to someone's comment on a post.

Message url to see all the chats of which you are a participant and all earlier

messages in them. You can send messages to them.

Noti\_cations come to user when someone likes his post or comments on it. No-

ti\_cation is also sent when someone replies to his/her comment. They are also

sent when someone posts on his/her wall. User is also noti\_ed when a new

message comes to any of the chats of which he/she is a member.

Groups have admin. They can add new admins. They can also accept requests

from other users to follow the group.

**ER diagram for stored data**

ER diagram is as below:

**A close up of a map

Description automatically generated**

**Table design for stored data**

1. **Person**

Attributes: Person ID, First name, Last name, Date of birth, gender, User-

name, Password, Profile picture, Address, Other details

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | The auto increment ID of Person, primary key with comments ‘personId PK’ |
| First\_name | varchar(50) | The first name of the person |
| Last\_name | varchar(50) | The last name of the person |
| Date\_of\_birth | timestamp | The birth date of the person |
| Gender | numeric(2,0) NOT NULL | The gender of the person |
| Mobile\_no | numeric(11,0) NOT NULL | The mobile number of the person |
| Email | varchar(256) NOT NULL | The email address of the person |
| Username | varchar(256) NOT NULL | The register username of the person |
| Password | varchar(256) NOT NULL | The register password of the person |
| Profile\_picture | varchar(256) | The picture of the person profile |
| Address | varchar(256) | The address of the person |
| Other\_details | varchar(256) | The other details of the person |
| Status | numeric(1,0) |  |

1. **Person\_auths**

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Auths\_ID | int | The auto increment ID of auths, primary key with comments ‘authsIdPK’ |
| Person\_ID | int | foreign key (Person\_ID) references Person on delete Cascade |
| Credential | varchar(256) NOT NULL | The credential of the third-party login, e.g. password\_hash, access token, etc. |
| Identity\_type | varchar(50) NOT NULL | The identity type of the third-party login, {email, phone, username, facebook, youtube, …} |
| Identifier | varchar(50) NOT NULL | The identifier for current login, E.g. email address, phone number, username, etc. |

1. **Post**

Attributes: Post ID, Text, Image, Video, Link, privacy, time

All are of type varchar

Image, Video contain link of the image/video

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Post\_ID | int | The auto increment ID of auths, primary key with comments ‘postIdPK’ |
| Text1 | varchar(256) | The text content of the post |
| Image | varchar(256) | The image of the post |
| Video | varchar(256) | The video of the post |
| Link | varchar(256) | The link of the post |
| Privacy | numeric(1,0) | Set privacy of the post |
| Time1 | timestamp | The creation time of the post |

1. **Posted\_By**

Attributes: Person ID, Post ID

A user can post many posts but a post can be posted by only one user.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Post\_ID | int | Primary key, foreign key (Post\_ID) references Post on delete Cascade |
| Person\_ID | int | foreign key (Person\_ID) references Person on delete Cascade |

1. **Group\_or\_Wall**

Attributes: group or wall id, name, type, Group description, Group photo

A place which contains the posts.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Group\_or\_wall\_id | int | The auto increment ID of group/wall, primary key with comments ‘groupOrWallIdPK |
| Name | varchar(50) | The name of group/wall |
| Type | numeric(1,0) | The type of group/wall |
| Group\_description | varchar(256) | The description of group/wall |
| Group\_photo | varchar(100) | The photo of group/wall |

1. **Posted\_on**

Attributes: Post ID, group or wall id

A post is posted to only one wall or group but a wall or group can contain

many posts.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Post\_ID | int | Primary key, foreign key (Post\_ID) references Post on delete Cascade |
| Group\_or\_wall\_id | int | Foreign key (Group\_or\_wall\_id) references Group\_or\_Wall on delete Cascade |

1. **Follower**

Attributes: Person ID, group or wall id

Maps users with the groups they follow.

A user can follow many groups or walls. A group or wall can be followed by

many users.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | Primary key, foreign key (Person\_ID) references Person on delete Cascade |
| Group\_or\_wall\_id | int | Primary key, foreign key (Group\_or\_wall\_id) references Group\_or\_Wall on delete Cascade |

1. **Admin**

Attributes: Person ID, group or wall id

Contains admins of group or wall.

There is a wall for a user whose admin is user himself.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | Primary key, foreign key (Person\_ID) references Person on delete Cascade |
| Group\_or\_wall\_id | int | Primary key, foreign key (Group\_or\_wall\_id) references Group\_or\_Wall on delete Cascade |

1. **Request**

Attributes: Person ID, group or wall id

Contains the list of the people who wish to follow a group or wall but are

not a follower yet. Admins can accept the request and allow them to follow.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | Primary key, foreign key (Person\_ID) references Person on delete Cascade |
| Group\_or\_wall\_id | int | Primary key, foreign key (Group\_or\_wall\_id) references Group\_or\_Wall on delete Cascade |

1. **Tag**

Attributes: Person ID, Post ID

Defines the person who is tagged in a post.

A person can be tagged in many posts and many people can be tagged in a

post.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | Primary key, foreign key (Person\_ID) references Person on delete Cascade |
| Post\_ID | int | Primary key, foreign key (Post\_ID) references Post on delete Cascade |

1. **Like**

Attributes: Person ID, Post ID

Defines the people who have liked the post.

A user can like many posts and a post can be liked by many people.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | Primary key, foreign key (Person\_ID) references Person on delete Cascade |
| Post\_ID | int | Primary key, foreign key (Post\_ID) references Post on delete Cascade |

1. **Comment**

Attributes: Comment ID, Text, time

Comment contains a text and the time at which it was posted.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Comment\_ID | int | The auto increment id of comment, primary key with comments ‘commentIdPK’ |
| Text1 | varchar(256) | The content of comment |
| Time1 | timestamp | The creation time of comment |

1. **Comment\_on\_by**

Attributes: Person ID, Comment ID, Post ID

Contains the person who commented on a post.

A comment is posted by only one user and on only one post.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | Foreign key (Person\_ID) references Person on delete Cascade |
| Comment\_ID | int | Primary key, foreign key (Comment\_ID) references Comment on delete Cascade |
| Post\_ID | int | Foreign key (Post\_ID) references Post on delete Cascade |

1. **Reply**

Attributes: Reply ID, Text, time

Just like there is a comment on a post, there is a reply on a comment.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Reply\_ID | int | The auto increment id of reply, primary key with comments ‘replyIDPK’ |
| Text1 | varchar(256) | The content of reply |
| Time1 | timestamp | The creation time of reply |

1. **Comment\_Reply**

Attributes: Person ID, Reply ID, Comment ID

Just like Comment on by

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Person\_ID | int | Foreign key (Person\_ID) references Person on delete Cascade |
| Reply\_ID | int | Primary key, foreign key (Reply\_ID) references Reply on delete Cascade |
| Comment\_ID | int | Primary key, foreign key (Comment\_ID) references Comment on delete Cascade |

1. **Chatbox**

Attributes: chatbox ID, name

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Chatbox\_ID | int | The auto increment id of reply, primary key with comments ‘chatBoxIdPK’ |
| Name | varchar(256) | The name of chatbox |

1. **Chatters**

Attributes: Person ID, chatbox ID

A chatbox is shared among several people and a person can have several chat-

box. It provides interface for displaying messages only to the members of the

chatbox.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Chatbox\_ID | int | Primary key, foreign key (Chatbox\_ID) references Chatbox on delete Cascade |
| Person\_ID | int | Primary key, foreign key (Person\_ID) references Person on delete Cascade |

1. **Message**

Attributes: Person ID, chatbox ID, Message text, time

Person ID here represents the person who sent the message to the chatbox

whose ID is chatbox ID

It also has the time at which the message was sent.

A message is sent by only one person and to only one chatbox

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Chatbox\_ID | int | The chatbox id of message, primary key, foreign key (Chatbox\_ID) references Chatbox on delete Cascade |
| Person\_ID | int | The creator of message, primary key, foreign key (Person\_ID) references Person on delete Cascade |
| Time1 | timestamp | The creation time of message, primary key |
| Message\_text | varchar(256) | The content of message |

1. **Notification**

Attributes: Notification ID, text, seen unseen, time

Contains the notification message for some user. seen unseen denotes whether

the notification is seen or unseen. time denotes the time at which the notification

was generated.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Notification\_ID | int | The auto increment id of notification, primary key with comments ‘notificationId PK’ |
| Text1 | int | The content of notification |
| Seen\_unseen | numeric(1,0) | The seen / unseen of notification |
| Time1 | timestamp | The creation time of notification, primary key |

1. **Notification\_for**

Attributes: Notification ID, Person ID

Maps the notification to the person to whom it corresponds

A person can have many notifications but every notification is associated with

only one person.

| **Fields** | **Type** | **Description** |
| --- | --- | --- |
| Notification\_ID | int | Primary key, foreign key (Notification\_ID) references Notification on delete Cascade |
| Person\_ID | int | Foreign key (Person\_ID) references Person on delete Cascade |

**The sequence diagram of third-party login**

**A screenshot of a cell phone

Description automatically generated**