

Simplified **YouTube** *Database Model*

BY

CS19B017

D. HAKESH

CS19B047

VINEET K.D.

CS19B078

SUNEET

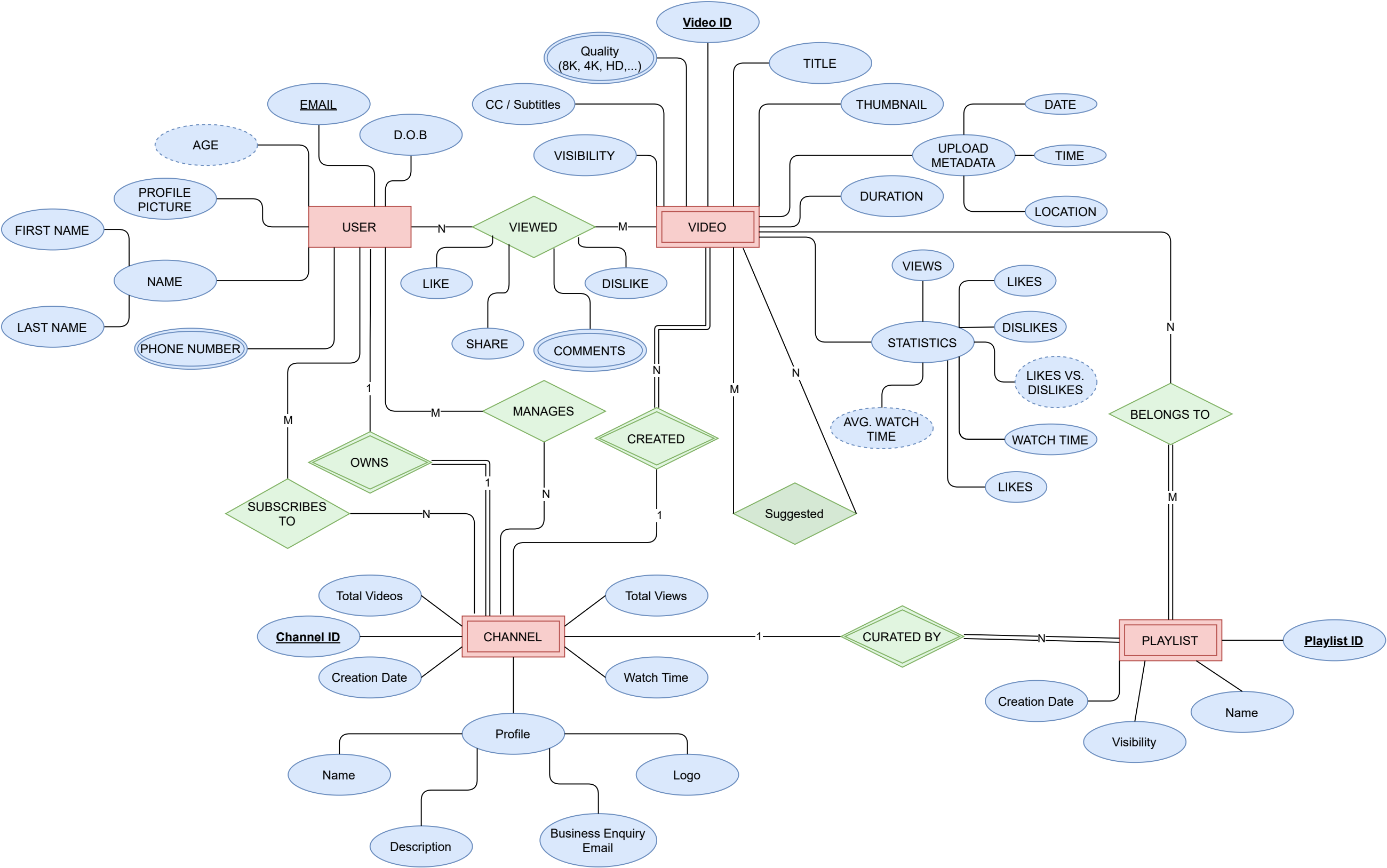
CS19B080

K.S.V. VINEETH

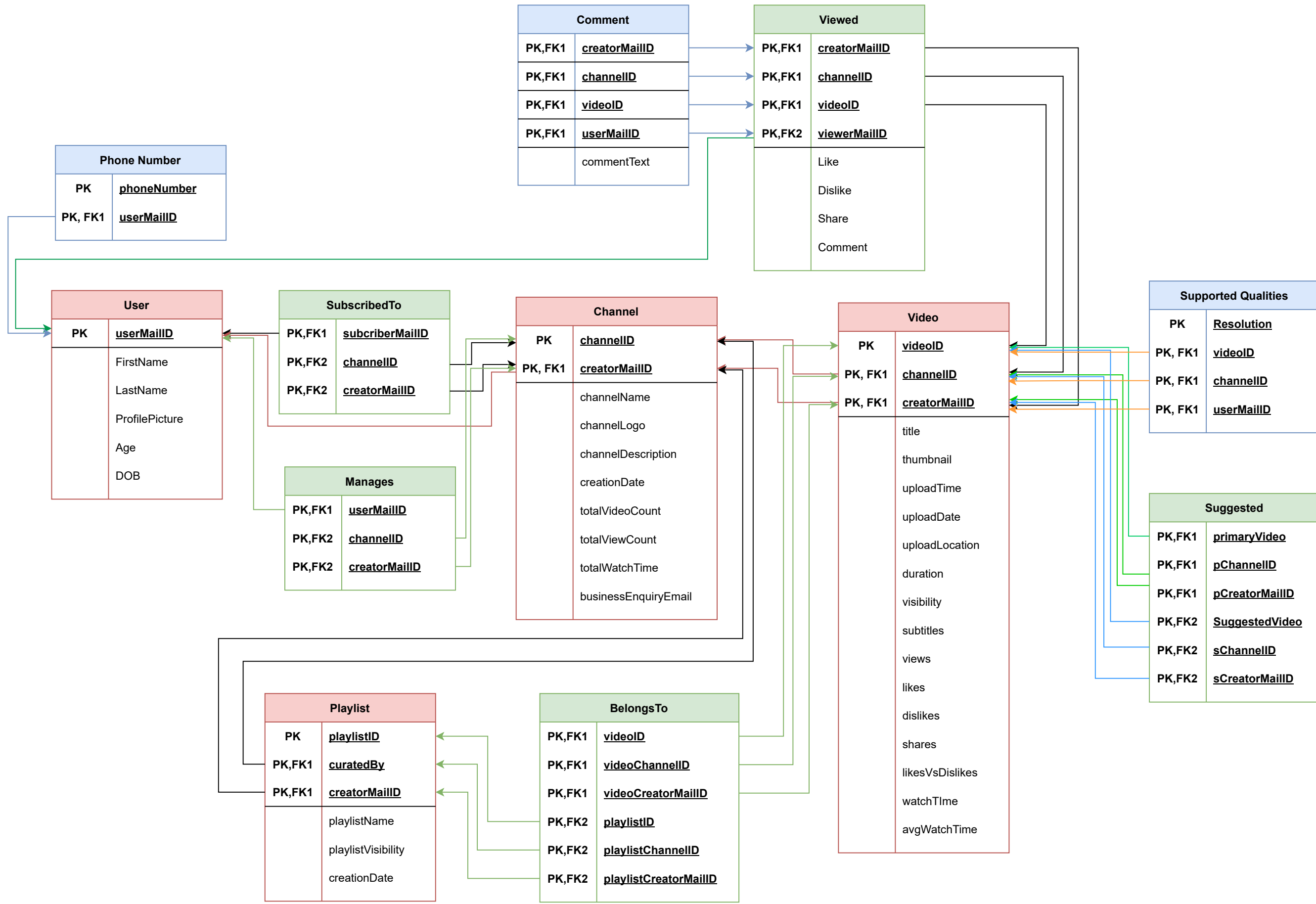
CS19B081

RAVISRI V.

Group - 6



Relational Schema Diagram



Domain Description:

In the youtube video platform, there are several users and channels. Each user has his unique email id. We also keep track of the user profile details. A channel contains streams of videos as playlists. A channel also can have multiple playlists. Every channel can be uniquely identified by the owner's ID and its corresponding channel ID. Similarly, playlist and video can be uniquely identified by their playlist/Video ID along with corresponding channel ID. Playlists have a name and visibility restrictions. Without a channel, it is not possible to have either a playlist or a video. We also keep track of profile, total videos, total views, joining date, watch time of every channel.

A user can create new videos in his channel (we assume in this data model that each user as an owner can create at most one channel) and can view videos of other's channels. A User can see and contribute to statistics details of videos like no. of views, likes/dislikes, comments. It can be possible that some users called managers who are not owners of the channel but can add videos to that channel. Viewer can subscribe to any particular channel so that he will get notifications when new video is added to that channel. Every video has follow-up video as suggestion. A video also has many details like when a video was added, duration of the video, quality, visibility, subtitles etc.

Explanation of the ER Diagram:

The ER Model describes the entities and attributes of the data that the platform will be storing. The entities involved in the model are - User Account, Channel (strong) and Playlists, Video (weak). Videos constitute a weak entity set which are related to the Channel uploading the video, allowing only videos uploaded by a Channel to exist. Playlists constitute a weak entity set where each playlist consists of a non-empty collection of videos.

The associated attributes of the various entities are as follows -

- User Account
 - Email address - a simple, single valued, key attribute - Email id in the format "id@xyz.com"
 - Profile picture - a simple, single valued attribute - an image with a resolution of 800x800 px
 - User name - a composite attribute consisting of
 - First Name - string of characters

- Last Name - string of characters
 - Date of Birth - a simple, single valued attribute - date of birth in DD-MM-YYYY format
 - Age - a derived, single valued attribute - unsigned integer
 - Phone Number - a simple, multi-valued attribute - a 10 digit number
- Channel
 - ID - a simple, single valued, key attribute - a positive integer ID (1, 2, 3 so on)
 - Profile - a composite attribute consisting of
 - Name - a string of characters
 - Logo - an image with a resolution of 800x800 px
 - Description - channel description in plain English text (upto 2000 characters)
 - Business Enquiry Email - Email id in the format "id@xyz.com"
 - Total Videos - a simple, single valued attribute - a positive integer
 - Total Views - a simple, single valued attribute - a positive integer
 - Watch Time - a simple, single valued attribute - a positive integer
 - Creation Date - a simple, single valued attribute - date in DD-MM-YYYY format
- Video
 - ID - a simple, single valued, partial key attribute - a positive integer ID (1, 2, 3 so on); the video ID in combination with the channel ID uniquely identifies a video
 - Title - a simple, single valued attribute - title in plain English text (upto 100 characters)
 - Thumbnail - a simple, single valued attribute - the image corresponding to the video displayed with a resolution of 1280x720 px
 - Upload Metadata - a composite attribute composed of
 - Date - date in DD-MM-YYYY format
 - Time - time of upload (IST)
 - Place - a valid location
 - Duration - a simple, single valued attribute - duration in hours, minutes and seconds

- Statistics - a composite attribute composed of
 - Number of Views - unsigned integer
 - Total Watch Time - Total time for which the video is viewed in seconds
 - Average View Duration - derived attribute computed as follows
 - $= (\text{Total Watch Time}) / (\# \text{ of Views})$
 - Number of Likes - unsigned integer
 - Number of Dislikes - unsigned integer
 - Number of Shares - unsigned integer
 - Likes vs. Dislikes - derived attribute computed as follows
 - $= (\text{Likes}) / (\text{Likes} + \text{Dislikes}) * 100$
- Quality - a simple, multi-valued attribute (video playback quality 4k, 1080p, 720p etc.)
- Visibility - a simple, single valued attribute - one of the following
 - Public - Visible to everyone on YouTube
 - Unlisted - Anyone with link can view
 - Private - Only visible to the owner and managers
- Subtitles - a simple, single valued attribute; consists of plain English text
- Playlist
 - ID - a simple, single valued, partial key attribute - a positive integer ID (1, 2, 3 so on); the playlist ID in combination with the channel ID uniquely identifies the playlist.
 - Creation date - a simple, single valued attribute - date of creation in DD-MM-YYYY format
 - Name - a simple, single valued attribute - a string of characters
 - Visibility - a simple, single valued attribute - one of the following
 - Public - Visible to everyone on YouTube
 - Unlisted - Anyone with link can view
 - Private - Only visible to the owner and managers

Relationships:

The relations between them are - “is Manager of”, “is Owner of”, “Created by”, “subscribed to”, “has View”, “Created”, “is part of”. The relations would help organise the videos on the platform for the owner and would help a user to watch the videos he would want. There are the 4 main types of relations - One - One (“is Owner of”), One - Many (“Created”), Many - One (“Created by”), Many - Many (“is Manager of”).

Description of Relations:

1. **User** ← “**Manages**” → **Channel**: The relation indicates that the particular user is a manager of the given Channel. Since a channel can have multiple managers and the same user can be a manager of multiple channels, this relation is Many - Many.
2. **User** ← “**Owns**” → **Channel**: This relation between user and channel indicates that the particular user owns that channel and all its monetary benefits. Since the owner of a channel would be unique and one user can only create only one channel, this relation is “One - One”.
3. **Channel** ← “**Created**” → **Video**: This is a relation between Video and Channel indicating which channel had uploaded that video. Since the same channel may upload multiple videos, this relation is One - Many.
4. **User** ← “**Subscribes To**” → **Channel**: This is again a relation between a user and a channel indicating that, that used has subscribed to the given channel. Since a user can subscribe to multiple channels and multiple users may subscribe to the same channel the relation is Many - Many.
5. **User** ← “**Viewed**” → **Video**: The given relation between user and video indicates that the particular user has viewed the given video. Since the user may watch multiple videos and the same video can be watched by multiple users, the given relation is Many - Many. The attributes associated with the relationship are -

- a. Liked - a simple, single valued attribute indicating whether the viewer liked the video
 - b. Disliked - a simple, single valued attribute indicating whether the viewer disliked the video
 - c. Shared - a simple, single valued attribute indicating whether the viewer shared the video
 - d. Comments - a simple, multi-valued attribute; consisting of text containing the viewer's comments on the video - only one comment per user is allowed
6. **Playlist** ← “**Curated By**” → **Channel**: The given relation between Channel and Playlist indicates that the particular playlist was created on the given channel. Since a channel can create multiple playlists but a playlist will belong to only one channel, the given relation is Many - One.
7. **Video** ← “**Belongs To**” → **Playlist**: The given relation between videos and playlist indicates that the particular video is a part of the given playlist. Since the same video can be a part of many playlists and a playlist can have many videos, the given relation is Many - Many.
8. **Video** ← “**Suggested**” → **Video**: The given relation between two videos of same channel indicates that the content of the videos are related and thus the user would be suggested to watch the second video once the first video ends. Since the second video can then have links to further videos the given relation is Recursive.