Relational Schema

Creators(channelId:VARCHAR(30) [PK], channelTitle:VARCHAR(30), categoryId:INT, video_id:VARCHAR(30) [FK to VideoInfo.video_id)
VideoInfo(video_id:VARCHAR(30) [PK], title:VARCHAR(100), publishedAt: DATETIME, tags:VARCHAR(100), description:VARCHAR(200), channelId:VARCHAR(30), categoryId:INT, video_id:VARCHAR(30) [FK to VideoMetrics.video_id])
VideoMetrics(video_id:VARCHAR(30) [PK], view_count: INT, likes: INT, dislikes: INT, comment_count: INT, video_id:VARCHAR(30) [FK to VideoInfo.video_id])
TrendingKeywords(keywords: VARCHAR(30) [PK], categoryId: INT, use_count: INT)
WebsiteUsers(user_id: INT [PK], username: VARCHAR(30), password: VARCHAR(30), email: VARCHAR(50), channelId: VARCHAR(30) [FK to Creators.channelId])

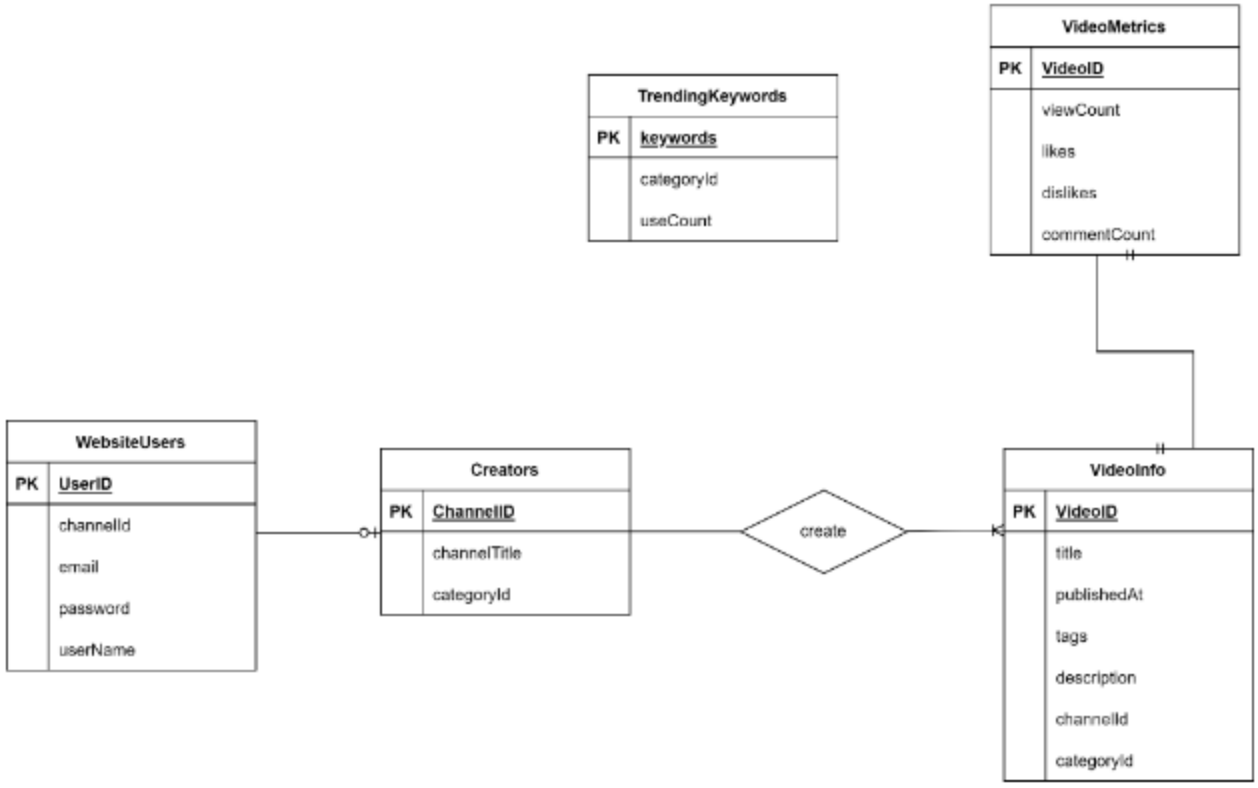
Assumptions made:

1. Creators to VideoInfo: We assume that for every one creator, there will be many videos created, with video information for each video made. As such, it is a one-to-many relationship.
2. VideoInfo to VideoMetrics: We assume that every video made will have unique video metrics. As such, it is a one-to-one relationship.
3. WebsiteUsers to Creators: Our web app allows creators to sign up as website users to track their own video metrics. In this case, we assume a one-to-one optional relationship to say that if there is one website user signed up, they can be linked to a content creator. Similarly, a content creator can be linked to one website user. We keep this optional because perhaps not all content creators will sign up to use the website.

Description of Relationships:

1. Creators: This table holds all the information related to the people who post videos. It contains a unique channel_id to identify each video as well as the channel title and overall category to which the channel's content is related.
2. VideoInfo: This table holds all the information related to the initial posting of a video. It contains a unique video_id to identify each video, and then holds the video's title, published date and time, tags and a description written when posting it, the YouTube channel it was posted to, and the category that the video falls under.
3. VideoMetrics: This table holds all the relevant information pertaining to the performance of a video. This would include metrics such as view count, likes, comment count, etc., which will help us dictate which keywords and categories result in the most successful youtube videos.
4. TrendingKeywords: This table was created to allow us easy and instant access to popular keywords based on video category. The table holds keywords, the category the keyword is associated with and tells the use_count which is the total number of times the keyword was used across all videos. This will help us rank the most popular keywords.
5. WesbiteUsers: This table includes all the information that is unique to each website user. It contains the user's login, password, channel id, and email. This table also allows us to link users with their individual content platforms.

ER Diagram


VideoMetrics
PK
VideoID
viewCount
likes
dislikes
commentCount
TrendingKeywords
PK
keywords
categoryId
useCount
WebsiteUsers
PK
UserID
channelId
email
password
userName
Creators
PK
ChannelID
channelTitle
categoryId
create
VideoInfo
PK
VideoID
title
publishedAt
tags
description
channelId
categoryId