Part 1 Introduction

0. Members

Team Titicaca	
Xuejing Wang	xw2668
Chucheng Lin	cl3915
Guangchen Lan	gl2713
Jiaqi Liu	jl5518

1. What will your project do? (xw)

We will implement an Influencer recommendation platform. Currently, we have various social media platforms, providing more opportunities for normal people including ourselves to have an influence on the internet. Meanwhile, we are also exposed to more information shared by other people. We will implement a platform for both influencers and audiences to categorize, rate and comment on the existing social accounts. The purpose is to help the audience filter out content they are not interested in and help them find their accounts of interest more efficiently. We also want to help the influencers set up a standard of good quality and help them identify their role and public recognition.

2. Who or what will be its users? (xw)

Our users can be anyone active on social media. This includes anyone who wants to share on social media platforms and anyone who wants to see other people's sharing, as well as those trying to find advertisement channels.

3. What to show in the demo? (gl)

After a user selected her/his favorite categories in GUI, related influencers will be shown in order of popularity.

4. What kind of data do you plan to store? (JL)

The data we plan to store consists of multiple databases, account information and user feedback for any social media account, and user information for any user.
Account information is stored in the format as:

- Channel id (primary key): The ID for a YouTube/Instagram channel.
- Social media type (sort key): The type of the social media account. Valid values are YOUTUBE and INSTAGRAM.
- Country code: A two-letter ISO-3166-1 country code, such as US, CN(China), or FR (France). The country code ZZ is used to report metrics for which YouTube could not identify the associated country.
- Age group: This dimension identifies the age group of the logged-in users associated with the report data. Valid values are 1_17, 18_24, 25_34, 35_44, 45_54, 55_64 and 65_.
- Gender: This dimension identifies the gender of the logged-in users associated with the report data. Valid values are FEMALE, MALE and GENDER_OTHER.
- Subscribers Count: The number of subscribers of this channel.
- Average Video/Post views: The average number of video/post views among certain periods of time duration (for example, 3 months).
- Tags: The most frequent k (k=5 by default) tags associated with this account (for example, PET, FOOD,TRAVEL, EDUCATION, DEEP LEARNING, tec).
- Link: The link to this account.
- Rating: A 10-scale score generated from the platform users' feedback, and the number of users have rated this account.

User Feedback is stored in the format as:

- User id (primary key): The ID for the platform user.
- Like: Whether the user is interested in this account. Valid values are TRUE and FALSE.
- Tags: The tags the user can manually add to this account.
- Comments: The comments the user writes about this account.
- Rating: A 10-scale score the user rates for the account.

User Information is stored in the format as:

- User id (primary key): The ID for the platform user.
- Email: The email address the user used to register.
- Phone: The phone number the user used to register.
- Gender: The gender of the user. Valid values are FEMALE, MALE and GENDER_OTHER.
- Age: The age of the user.
- Country: The country the user is associated with.
- Interests (sort key): The tages the user manually selects as interested.

5. What API do you plan to use and what will you use it for? (cl)

1. `GET https://www.googleapis.com/youtube/v3/videos`
   `Usage:`
   - `Get popular videos on youtube. Get corresponding channels as influencers`
   - `Get corresponding categories from channels, which can be used to facilitate recommendations`
2. `GET https://www.googleapis.com/youtube/v3/search`
   `Usage: Search related channels as recommendations to users`
3. `GET https://www.googleapis.com/youtube/v3/members`
   `Usage: Get the number of subscribers for channels, which can be used to decide whether a channel is influential or not.`
4. `Similar APIs as above from other social media such as Instagram and Tiktok.`

Part 2 User Stories

1. As a user, I want some popular youtube channel recommendations so that I could kill time.
   a. When I login to my account, I should be able to see a recommendation list.
   b. If I like some channel before, when I login to my account, I should be able to see a list of recommended channels that are somehow related to the channel I like before.
2. As a popular influencer, I want my channel to have fair scores so that I could have more subscriptions.
   a. If I get less subscriptions, my scores should be lower than before.
   b. If I get more subscriptions, my scores should be higher than before.
3. As an audience, I would like to know about popular channels that's related to my interest. My conditions of satisfaction are:
   a. If I'm interested in traveling, I should be able to select a category named "travel" and see all the recommended traveling channels.
   b. When the channels are shown, I should be able to see each channel's introduction, rating, and popular comments. I should also be able to go to that channel through a link easily.

4. As an advertiser, I want to look for business partnerships of the most popular channels so I could optimize my advertisement. My conditions of satisfaction are

a. When I log in my account, I should be able to select some tags related to the product I would like to advertise about, and see all associated channels.
b. When all associated channels are shown, I should be able to filter the search results by specifying the social media type, gender, age range, and country.
c. When all channels are filtered by my requirements, I should be able to sort the channels by rating, subscribers, average video/post views among a certain duration, so I can pick some most popular channels to collaborate with.
d. After I picked a channel, I should be able to get the link so I can be redirected to the channel.

Part 3 Acceptance Testing

1. Given a user's subscriptions, when the user login to his or her account, the system should recommend some similar and popular channels to the user.
    a. Common case: when the user logs in, he or she should be able to see recommended channels shown in the UI.
    b. Special case: The user logs in and likes a channel. Then the user logs out. When he or she logs in again, a list of recommended channels which are related to the channel the user liked before should be shown in the UI.
2.
    a. Common case: When an influencer has lots of subscribers, she/he/it should get high scores on the recommendation platform.
    b. Special case: When an unpopular influencer cannot find her/him/it on the platform, she/he/it will receive a link to upload her/his/its information from other social media.
3.
    a. Common case: when the user logs into his or her account, she/he should be able to choose the category named "travel" and get the list of popular travelling influencers with a brief introduction, the link to their account, their rates and popular comments.
    b. Special case: when the user wants to search for a topic that's not a built-in category we've prepared, we will prompt the user with the available topics we have and express that we will continue developing to satisfy his/her needs
4. a. Common cases: When the user logs into his or her account, she or he should be able to select or type some tags, filter the resulting list of accounts by restriction (country, gender, age range, etc.) and sort them by popularity.
   b. Special Case:
   (i). When the user type in some unprepared tags, the application should prompt the user with available topic lists for selection. The tags the user attempts to use will be recorded and possibly becomes a new tag due to its frequency.
   (ii). When the user is cannot sort the account list by rating due to the lack of information, for example, there are not enough people rating the accounts. The account list is presented due to its relatedness to the tags, and a prompt that explains

the lack of rating information and suggests the user to sort by other popularity standards, for example subscribers count or average video/post views, should pop up.

Part 4 Facilities (xw)

1. We will build the front end using js, css and the bootstrap framework.
2. We will store and update our influencer data in an AWS DynamoDB.
3. We will implement the logic in Java, built with Maven, in dropWizard framework, and intelliJ IDE. We'll use the google coding style for style check and the built in bug finder of intelliJ. JUnit testing will be adopted. The built in coverage of intelliJ will be used for coverage testing.
4. We will perform data scraping in Python, from the APIs mentioned above.