

Application Development for Mobile Devices

CSC 0780

Fall 2022

Florian Cartozo - Antoine Rosselli

06/12/2022

History Table

Version	Submission Date
M1	06/12/2022

Table of Contents

1. Project Description	3
2. Use Cases	4
3. Database Requirements	7
4. Detailed List of Main Entities, Attributes and Keys	9
5. Entity Relationship Diagram (ERD)	12

1. Milestone 1

Tavu is an application on IOS where you can daily share a video that you recently watched on Youtube (history of your 10 last video) with your friends. You can like these videos and your friends can do the same.

Must have features

- Create an Account
- Link your Account with Youtube
- Join Group
- Get liked video from youtube account
- Shared 1 liked video from youtube account

Nice have features

- Leave Group
- Create Group

2. Milestone 2

Estimate how much work each feature will take to implement (in hours)

In order to facilitate our work and because we are used to working together on many projects, we have separated the tasks into 2. The design part and the back-end part.

Antoine takes care of the design part of the application, i.e. creates all the views that we will need.

And Florian takes care of the functional part, i.e. making the application dynamic with real data and not fake data.

Views

View Controller : 5h (Antoine)

Here we need to create the view that will allow the google oauth

Create 1 button to add Youtube scope on google account

Create 1 button to call Google oauth

We need to create structure user

HomeView Controller : 5h (Antoine)

Here we need to create the view that will handle all video that was shared this day

We need to display them

Disable add video button if user already posted

We need to create structure posted-video

VideoSelect Controller : 5h (Antoine)

Here we need to handle all the liked video of the current user

We need to display them

We need to create structure video

GroupView Controller : 5h (Antoine)

Here we need to display all the information about the group

Display group name

Display number of members

Display list of member

Add button to change group

We need to create structure group

GroupChange Controller : 5h (Antoine)

Here we need to ask new group that we want to change

Automatically leave previous group if there is one

Change current group

Features

Create an Account : 5h (Florian)

Here we need to create a back-end in order to create an account.

Design account model

Create endpoint

Link iOS app to the back-end

Link your Account with Youtube: 5h (Florian)

Here we need to implement google oauth 2.0 on our app.

We need to add Youtube scope

We need to send accessToken to the back-end

Join Group: 3h (Florian)

We need to create an endpoint on the back-end to join group

We need to call the endpoint in our app

Get liked video from youtube account: 5h (Florian)

Here we need to get accessToken

Send it to back-end

Back-end call Youtube API and get liked videos

Back-end return a list with liked videos

Display liked videos on select on our app when you want to share a video

Shared 1 liked video from youtube account: 5h (Florian)

Here we need to call back-end to know if user has already share a video today

If not, get liked video

User select which video he wants to share

Video is sent to the back-end to store which video user shared today.

Leave Group: 2h (Florian)

Here we need a endpoint on the back to leave a group

Link this endpoint with our app

3. Milestone 3

Each member contributed to this project equally

Total hours for Antoine : 25h

Total hours for Florian : 25h

This project was a little hard because we used the Youtube API and a Back-end.
The app isn't perfect, so there are some bugs but it was fun !

Know bugs:

- When we want to log for the first time, we need to grant youtube scope with the add youtube scope button, but sometimes we need to do this manipulation multiple times before it works. Maybe it is a problem with google oauth and we didn't find any solution.
- App have difficulties to fetch data from back-end, we need to change view and come back to refresh