

FINAL REPORT : METUBE

A MULTIMEDIA DATABASE PROJECT

TEAM NUMBER : G3

TEAM MEMBERS:

PRASHANTH REDDY KADIRE

PAVANI TATIKONDA

JAHNAVI NELLEPALLI PARTHASARATHI

TABLE OF CONTENTS

INTRODUCTION

SYSTEM DESIGN

E-R DIAGRAM

DATABASE SCHEMA

FUNCTION DESIGN

FUNCTIONAL SPECIFICATION & TESTING

1.USER ACCOUNT

2. DATA SHARING

3. MEDIA ORGANISATION

4. USER INTERACTION

5. SEARCH

IMPLEMENTATION DETAIL

USER MANUAL

CONCLUSION

INTRODUCTION

MeTube is a tweaked version of the well-known YouTube website. It includes films, audio, photos, and animation snippets, among other things. Users are divided into two categories: registered and unregistered. Users who are not registered can only watch or download the media. After signing in to their account, registered users can submit video, access material, change their profile, and so on. Registered users can also subscribe to channels, create playlists, and send and receive messages from one another using the built-in messaging system.

SYSTEM DESIGN

The MeTube functions are implemented using the MVC architecture. As seen below, the MVC design pattern splits an application into three primary aspects: Model, View, and Controller.

The term "model" refers to the data that must be shown, as well as all the data-related logic that the user interacts with. This can be the data being transmitted between the View and Controller components or any other data related to business logic. A User object, for example, will retrieve user information from a database, change it, and either update or render the data back to the database.

All of the application's UI functionality is handled by the View component. The Customer view, for example, will comprise all of the UI components that the final user interacts with, such as text boxes, dropdowns, and so on.

Controllers serve as a link between the Model and View components, processing all business logic and incoming requests, manipulating data using the Model, and interacting with Views to produce the final output. The User controller, for example, will handle all interactions and inputs from the User View and use the User Model to update the database. The User data will be seen using the same controller.

MVC is popular because it separates the application logic and user interface layer, allowing for separation of responsibilities. The Controller accepts all

application requests and then works with the Model to prepare any data that the View requires. The View then generates a final presentable response using the data prepared by the Controller. The MVC abstraction can be represented graphically as follows.

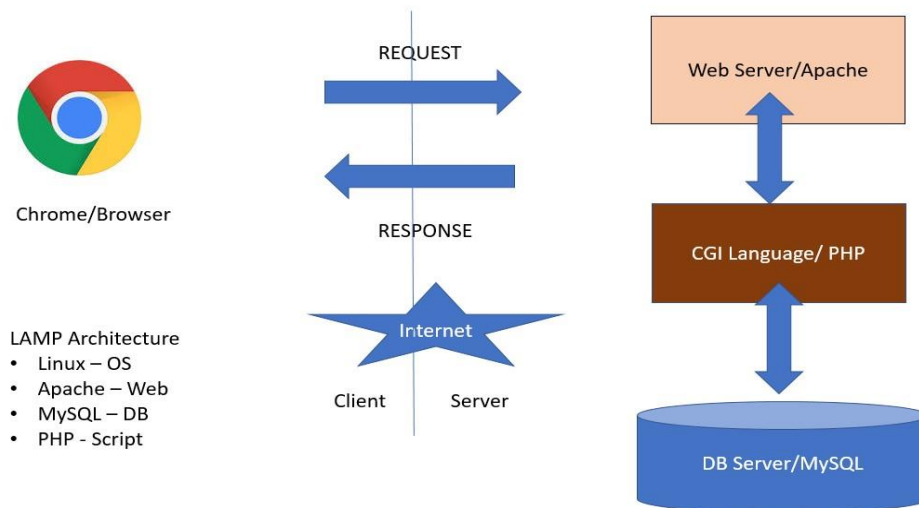


FIG 1 System Design

E-R DIAGRAM

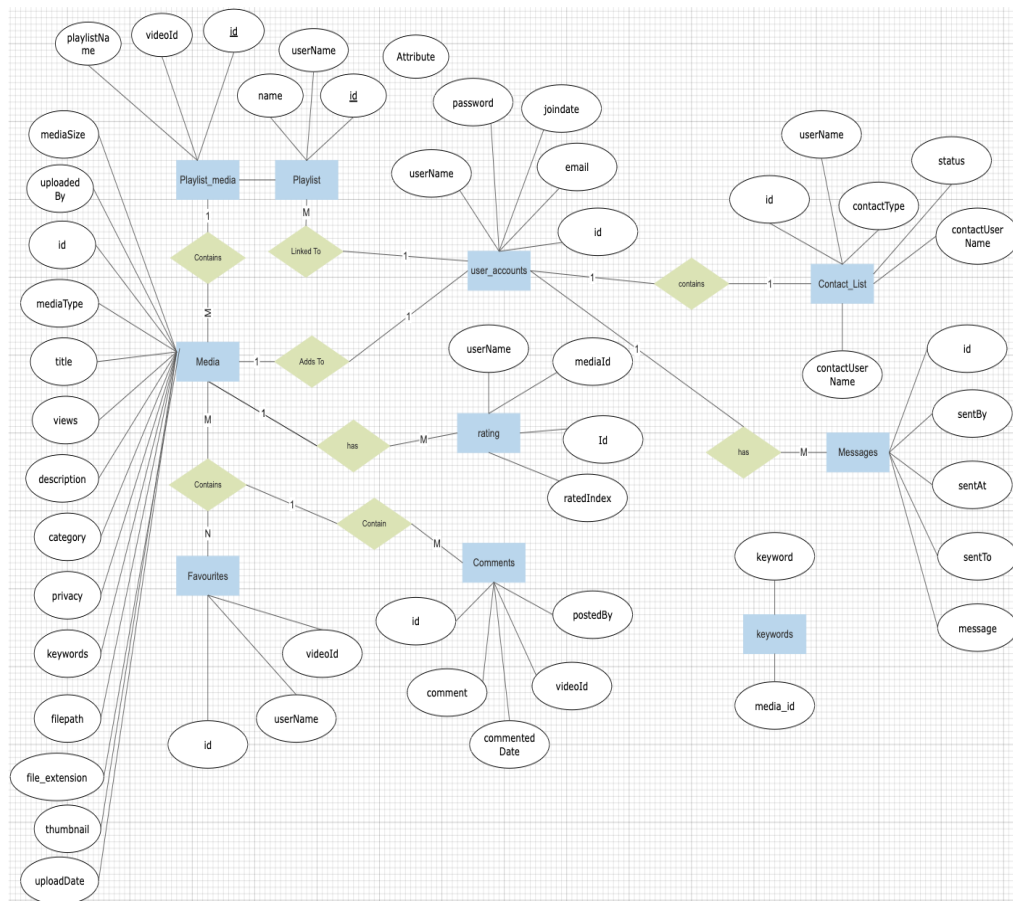


FIG 2. ER DESIGN

DATABASE SCHEMA

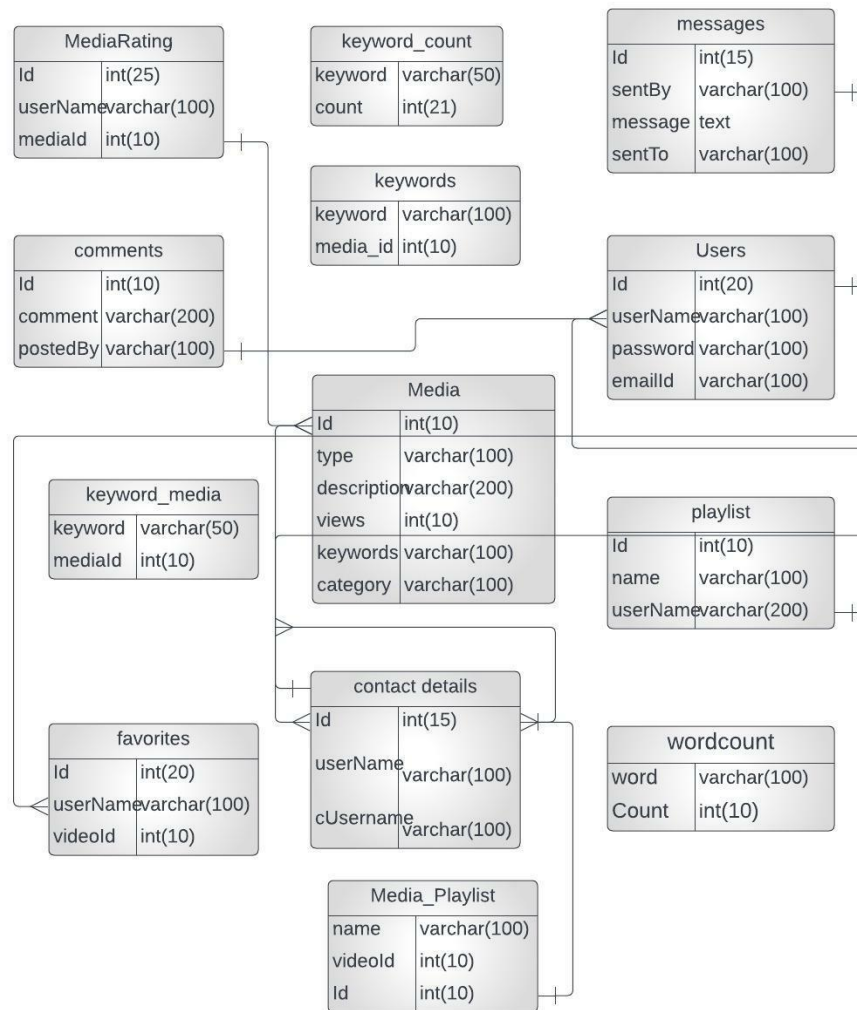


FIG 3. Database Schema

Functional Design

REGISTRATION :

- Please fill out all of the HTML form's input fields.
- The username must be checked to see if it already exists in the database.
- Check to see if the email address is correct.
- Check to see that the password is at least six characters long and matches the confirmed password.

- If all of the above tests pass, insert the data into the database and provide a success message.
- Otherwise, show the user an error message.
- End

SIGN IN

- Should accept the username and password from the login HTML form as input.
- If the user does not exist in the database, a user not found message will be displayed.
- Check to see if the password is the same as the username in the database.
- If the password is correct, create a session for the user and log them in.
- End

UPDATE PROFILE

- Using the HTML form, enter the email address or password to be updated.
- In the event of an email, verify that the entered value is a valid Email and update the row in the database table/
- If the request is for a password change, make sure the current password is the same as the one entered, i.e. the previous password is the same as the new password.
- Validate the new password is six characters long and identical to the confirm password.
- Update the database to reflect the change.
- End

UPLOAD

- Select a file from the system and fill up the HTML form with meta data
- Create a folder under/uploads/\$username if the user has never uploaded a media file.
- If the uploaded file already exists in /uploads/\$username, a prompt will appear stating that duplicate files exist and that you should submit it again.
- If there are no duplicate files, add new items to the media & keywords table.

- The user can edit media information once it has been uploaded by clicking the edit media info button.
- End

PLAYLIST

- Fill in the form's playlist name and the session variable's username.
- Check if the user's playlist already exists; if not, enter the user's username and the playlist's name into the playlist's username and name fields.
- Remove the playlist from the database by choosing it from the dropdown menu.
- Check if media already exists for the logged in user in the playlist where it is to be added when adding media to a playlist.
- If no media is present, add it to the playlist; else, show the media that is already there.
- When the user selects a playlist from the menu and clicks on view playlist, the database is queried for media in the playlist.
- End

DOWNLOAD

- Use the video object's getID() parameter to enter the video Id.
- Using the file extension listed in the database, download the file.
- The download button will save the file to the user's downloads folder.

FAVORITE

- Using the favorites button, you can add or remove media from the playlist.
- Input the current media's id and the username from the session data.
- Check to see if that user's media id exists in the database.
- If the media id already exists, set the button value to remove from playlist, and the media will be removed from the database.
- If the media does not exist in the database, when the user clicks on it, the database will be updated with the media id.

SEARCH

- In the search bar, the user types keywords and presses the search button.

- Show the results (a list of media based on a sql query on the terms "media join")
- You can narrow down your search by size (all, 0-100K, 100-100K, >1000K).
- Different features (title, recently uploaded, size, etc.) are used to sort the media results.

MESSAGE

- Fill in the recipient ID from the menu and the message content with the username from the session.
- Add the user message, username, and recipient name to the database.
- View inbox messages uses a select query to show all messages where the receiver is the currently logged in username.
- View sent messages displays all messages where sent user = logged in username using a select query.

CONTACTS

- Use the session's username and the menu to select a friend ID.
- Check to see whether there is already a link between the users.
- If the relationship already exists in the database, make the necessary changes.
- Otherwise, create a database relationship between the contacts.
- To ban or unblock a user, select the user from the dropdown menu.
- Check the state of the users, and based on the user input, enter or update the row.

FUNCTIONAL SPECIFICATION AND TESTING

1. USER ACCOUNT:

a. User registration:

The user can register as a new user by clicking the "Register" button if all fields are met.

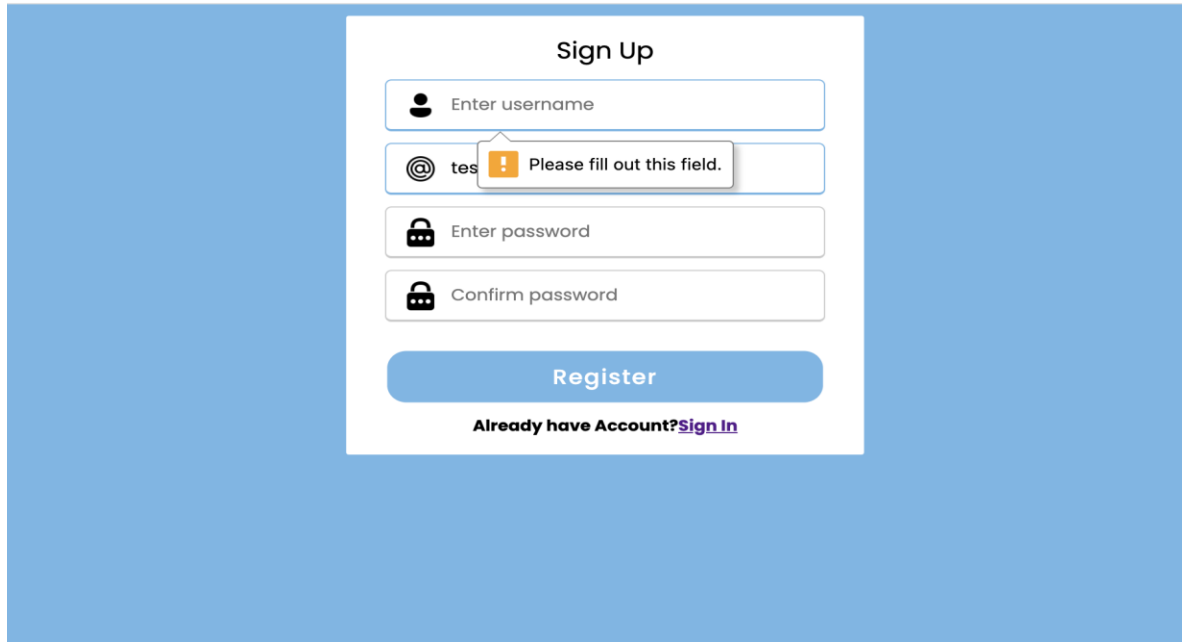
The image shows a 'Sign Up' form on a blue background. The form is white and contains the following elements: a title 'Sign Up', four input fields with icons (person, email, and two padlocks) and labels 'Enter username', 'Enter email', 'Enter password', and 'Confirm password', a blue 'Register' button, and a link 'Already have Account? Sign In'.

When user entered email is in valid and when passwords length doesn't match the validation logic of 6 characters or above.

Test case 1: while Registering if you enter invalid email, it displays mail is invalid

The image shows the same 'Sign Up' form as before, but with validation errors. Below the 'Enter email' field, the text 'Email is not valid' is displayed. Below the 'Enter password' field, the text 'Passwords Must be 6 characters or above' is displayed. The 'Register' button and the 'Sign In' link remain visible.

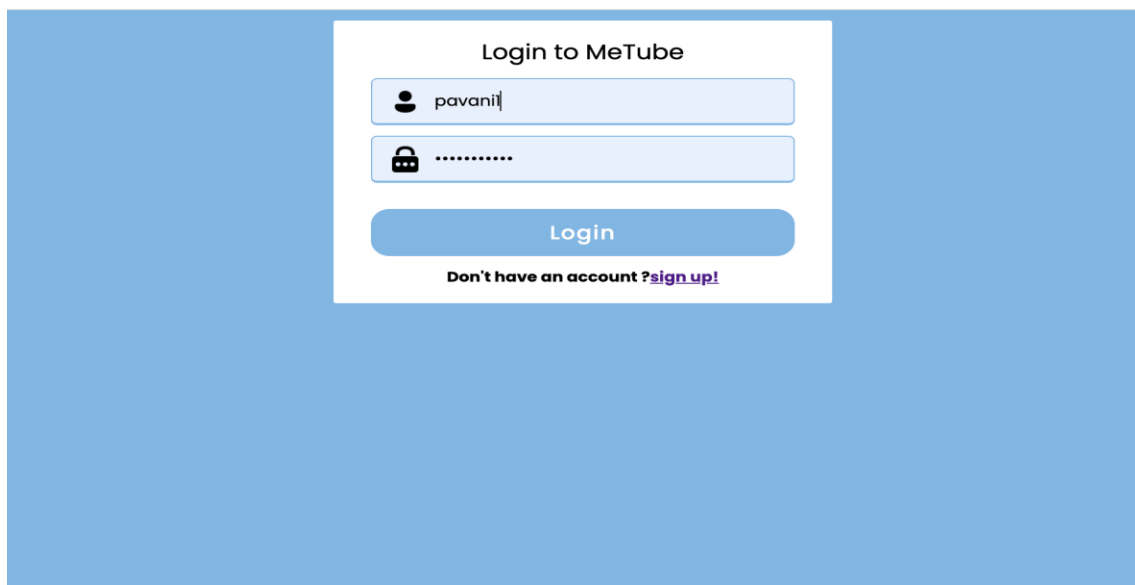
Test case 2: when you try to Register without filling in the username field.



The image shows a 'Sign Up' form on a light blue background. The form is white with a title 'Sign Up' at the top. It contains four input fields: 'Enter username' (with a person icon), 'Enter email' (with an '@' icon and the text 'tes' visible), 'Enter password' (with a lock icon), and 'Confirm password' (with a lock icon). A blue 'Register' button is at the bottom. Below the button is a link: 'Already have Account? [Sign In](#)'. A yellow error message box with an exclamation mark icon is positioned over the email field, displaying the text 'Please fill out this field.'

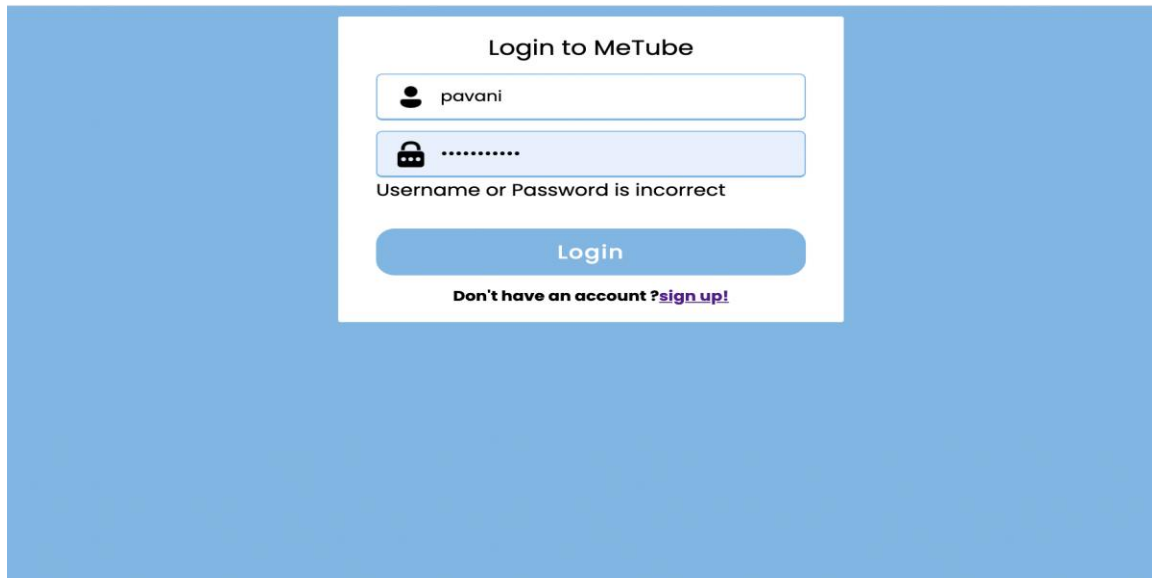
Sign-In:

User can login into the MeTube app using SignIn page and when they enter right user name and password and click “Login” they can able to login to the MeTube Application.

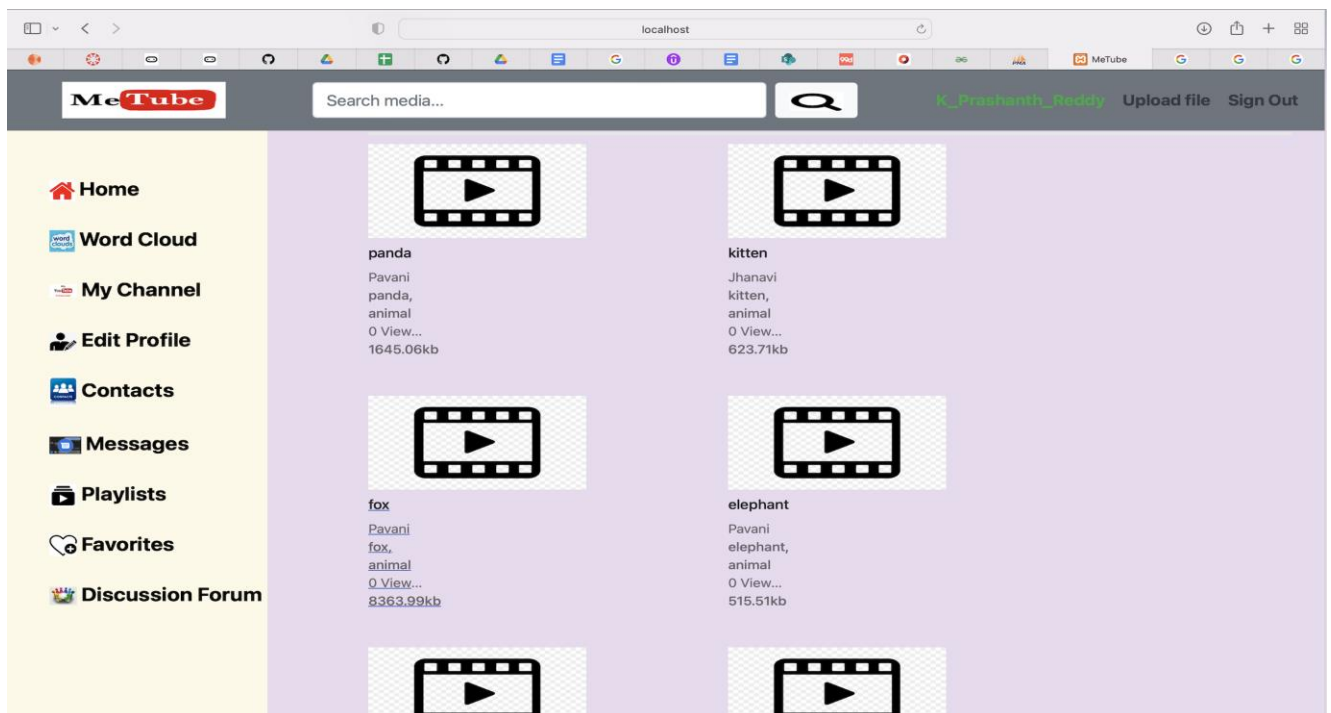


The image shows a 'Login to MeTube' form on a light blue background. The form is white with a title 'Login to MeTube' at the top. It contains two input fields: 'Enter username' (with a person icon and the text 'pavanit' visible) and 'Enter password' (with a lock icon and masked text '.....'). A blue 'Login' button is at the bottom. Below the button is a link: 'Don't have an account ? [sign up!](#)'.

Throws an error message when username and password is incorrect in SignIn page.

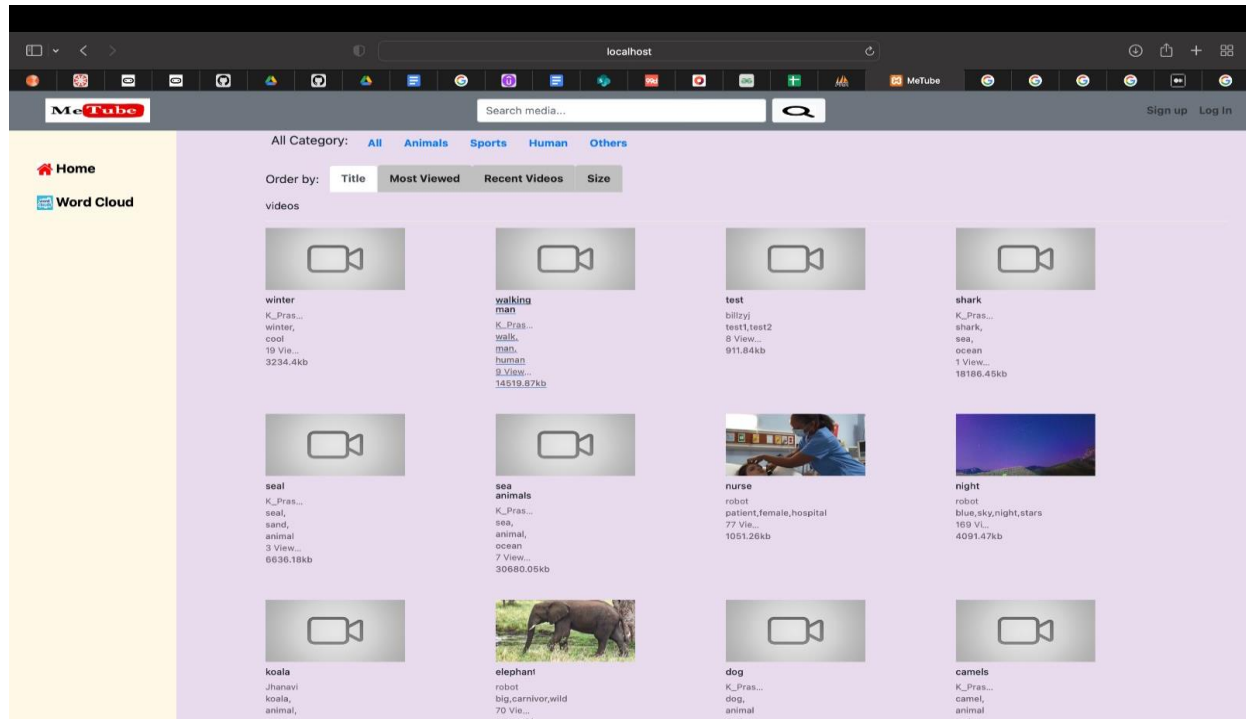


When User successfully able to login to the Metube application with the username “K-Prashath_Reddy” and the username is display at the top right of the screen.



b. Sign-Out

Application redirects to the MeTube homepage upon Signing out and shows the Home page of the app with Sign Up and Login at the top right of the page.



c. Profile Update

Profile Update allows user to change the basic information of the user such as updating email address and updating password.

The screenshot shows a web browser window with the URL 'localhost'. The page has a dark header with the 'MeTube' logo, a search bar, and links for 'Upload file' and 'Sign Out'. A left sidebar contains navigation links: Home, Word Cloud, My Channel, Edit Profile, Contacts, Messages, Playlists, Favorites, and Discussion Forum. The main content area is titled 'Update your profile' and contains two sections. The first section, 'Enter an updated email address:', has a text input field labeled 'New email address' and an 'Update Email' button. The second section, 'Enter Old Password', has a text input field labeled 'Old password' and an 'Update Password' button. Below this, there are fields for 'Enter New Password' (labeled 'New password') and 'Enter Confirm New Password' (labeled 'Confirm new password').

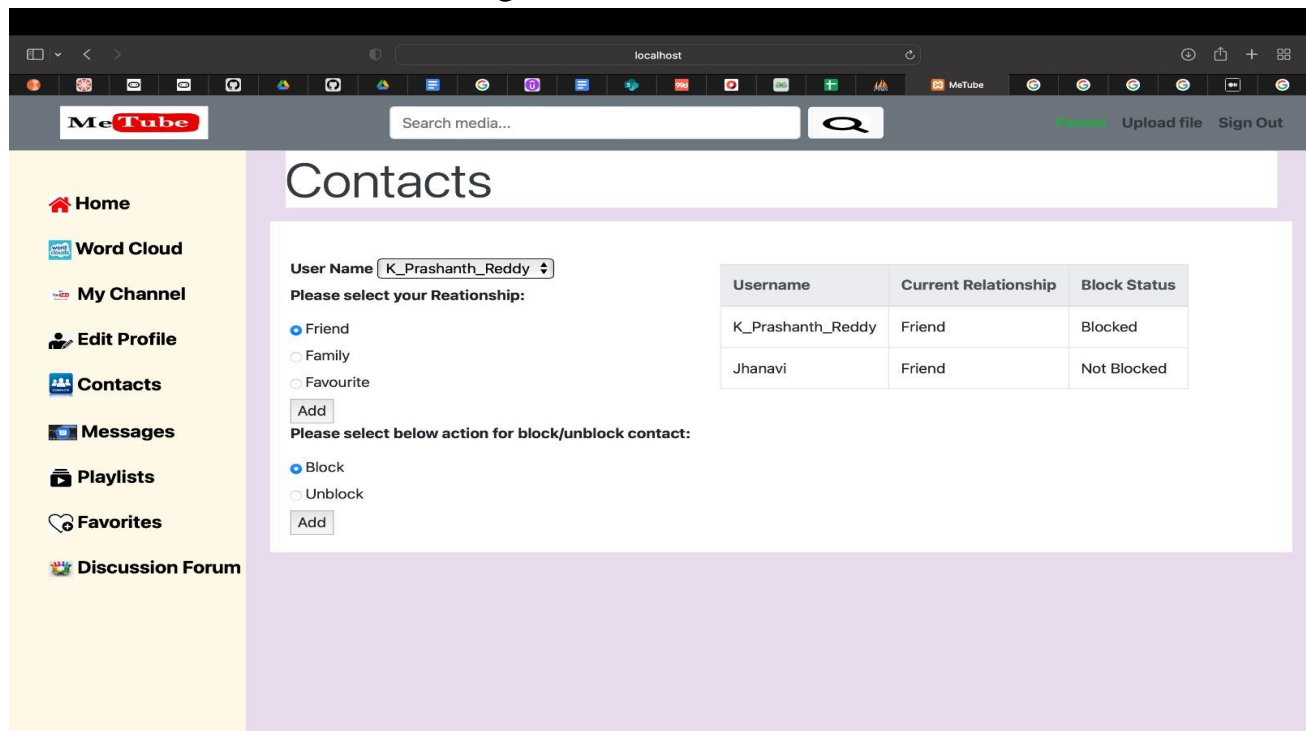
When the Entered Email address is not valid then it throws an error message saying “Email is not valid”

Testcase 3: When you try to update the email by providing invalid email it shows email is invalid.

This screenshot shows the same 'Update your profile' form as the previous one, but with an error message. The 'New email address' input field now contains the text 'Email is not valid'. The 'Update Email' button remains visible below the input field. The rest of the form, including the password fields and the 'Update Password' button, remains unchanged.

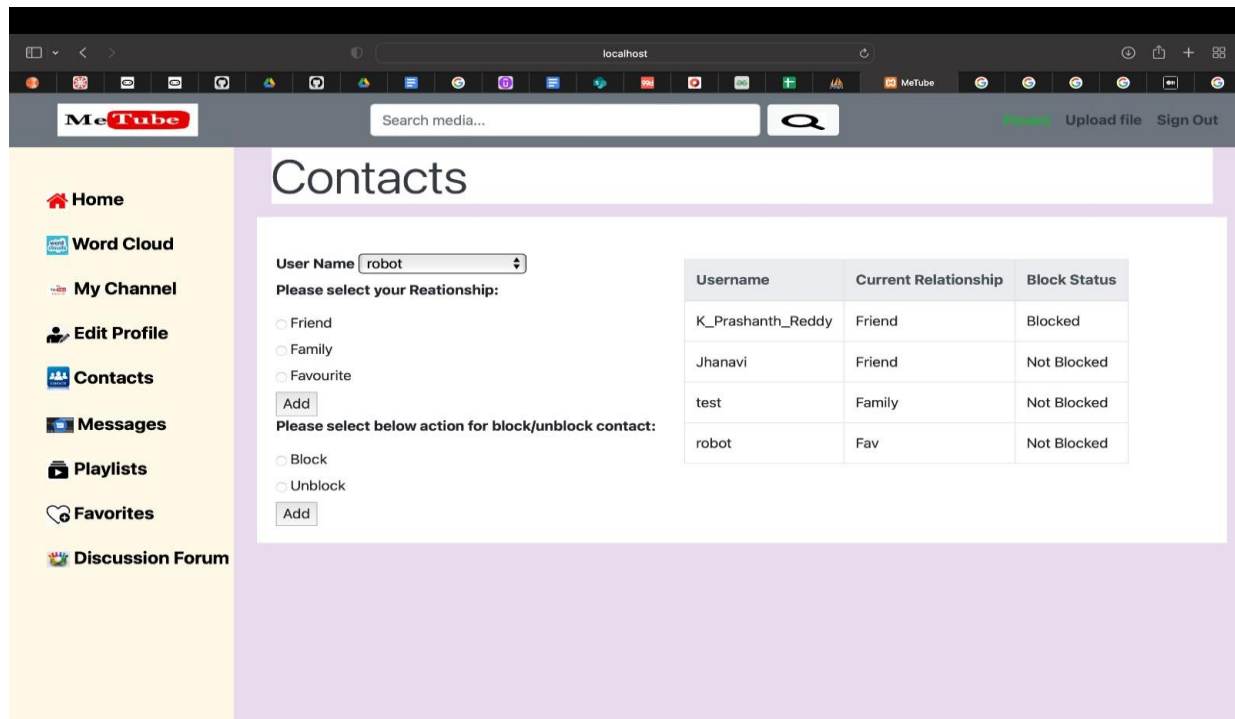
e. Contact List:

Contact List allows users to organize their contacts into a contact list



f. Contact List Organization and Blocking a user:

Contact Page also adds a relationship status from the list of radio buttons(Friend, Family, Favorites) and also able to block/unblock certain contact by selecting the radio button(Block/Unblock).



2. DATA SHARING

a. Upload multimedia files :

To upload multimedia files, a user must be logged in to the system. The user is led to their home page after logging in, which has a "Upload File" button at the top. They can go to the "Upload a File" page by clicking on that.

The fields that must be filled out while uploading a file are shown in the screenshot below. It contains all the information such as the choose file, Media title, Keywords, Select a category, visibility, and Media Description.

Test case 4 : When you forgot to fill one of the field it will show fill out this field.

The screenshot shows the MeTube web application interface. On the left is a navigation sidebar with links: Home, Word Cloud, My Channel, Edit Profile, Contacts, Messages, Playlists, Favorites, Discussion, and Forum. The main content area is titled "Select a file to Upload*" and contains the following fields:

- Choose File:** A text input field containing the placeholder text "teacher students+shool+classroom+child.mp4".
- Media Title*:** A text input field with a tooltip that says "Fill out this field". The field contains the placeholder text "Enter the title name...".
- Keywords:** A text input field containing the placeholder text "Enter the keywords separated by comma(,)"
- Select a category:** A group of radio buttons with options: Animal, Sports, Human, and Other. The "Other" option is selected.
- Visibility:** A group of radio buttons with options: Public, Family, Friends, and Favorite. The "Public" option is selected.
- Media Description:** A large text area containing the placeholder text "300 characters maximum..."

At the bottom of the form are two buttons: "Upload" and "Cancel".

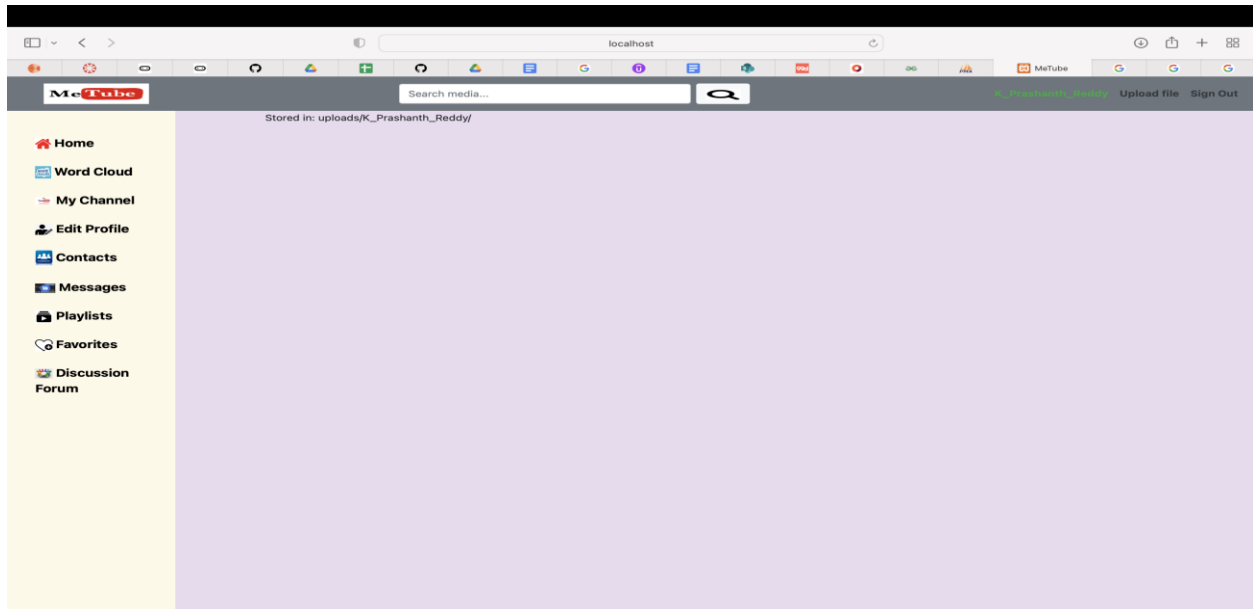
After filling in all the fields click on the Upload button.

This screenshot shows the same MeTube upload form, but now it is filled out with specific information:

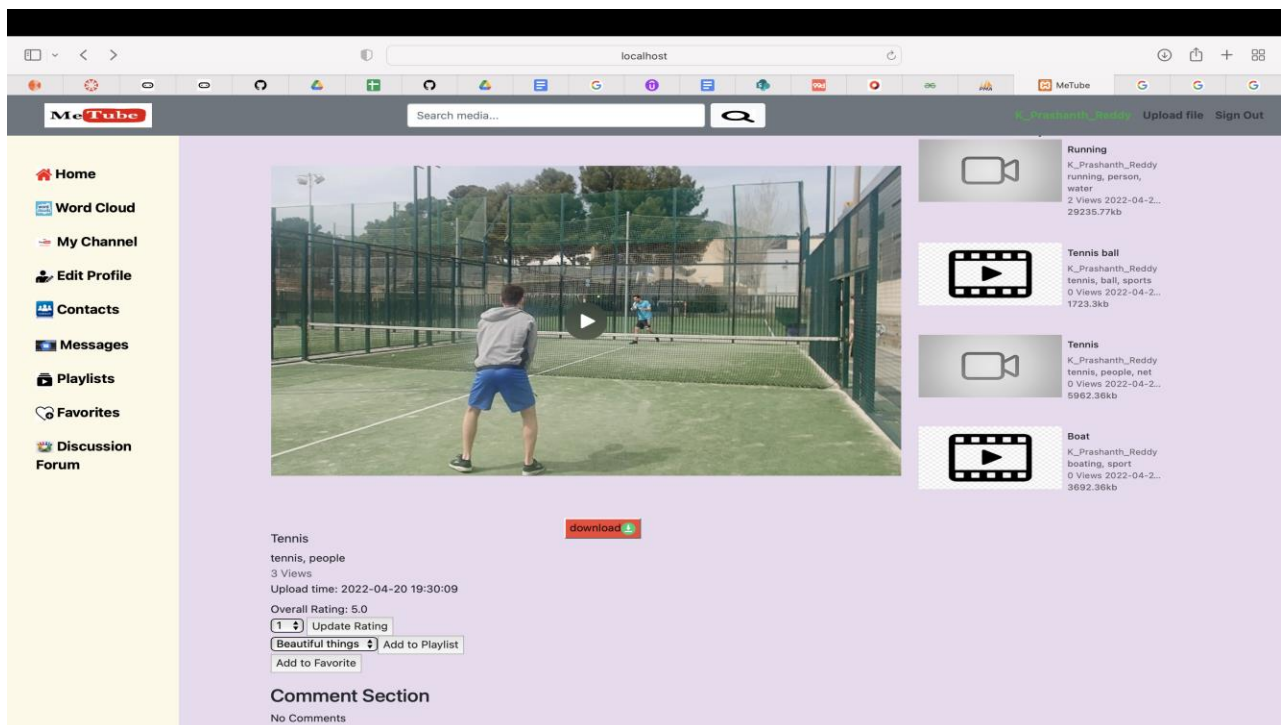
- Choose File:** The text input field now contains "tennis ball.jpg".
- Media Title*:** The text input field now contains "Tennis ball".
- Keywords:** The text input field now contains "tennis, ball, sports".
- Select a category:** The radio button for "Sports" is now selected, while "Other" is unselected.
- Visibility:** The radio button for "Public" remains selected.
- Media Description:** The text area now contains the text "A Tennis ball in the court".

The "Upload" and "Cancel" buttons remain at the bottom of the form.

After successfully Uploading the media file it will show the path where the video has been uploaded



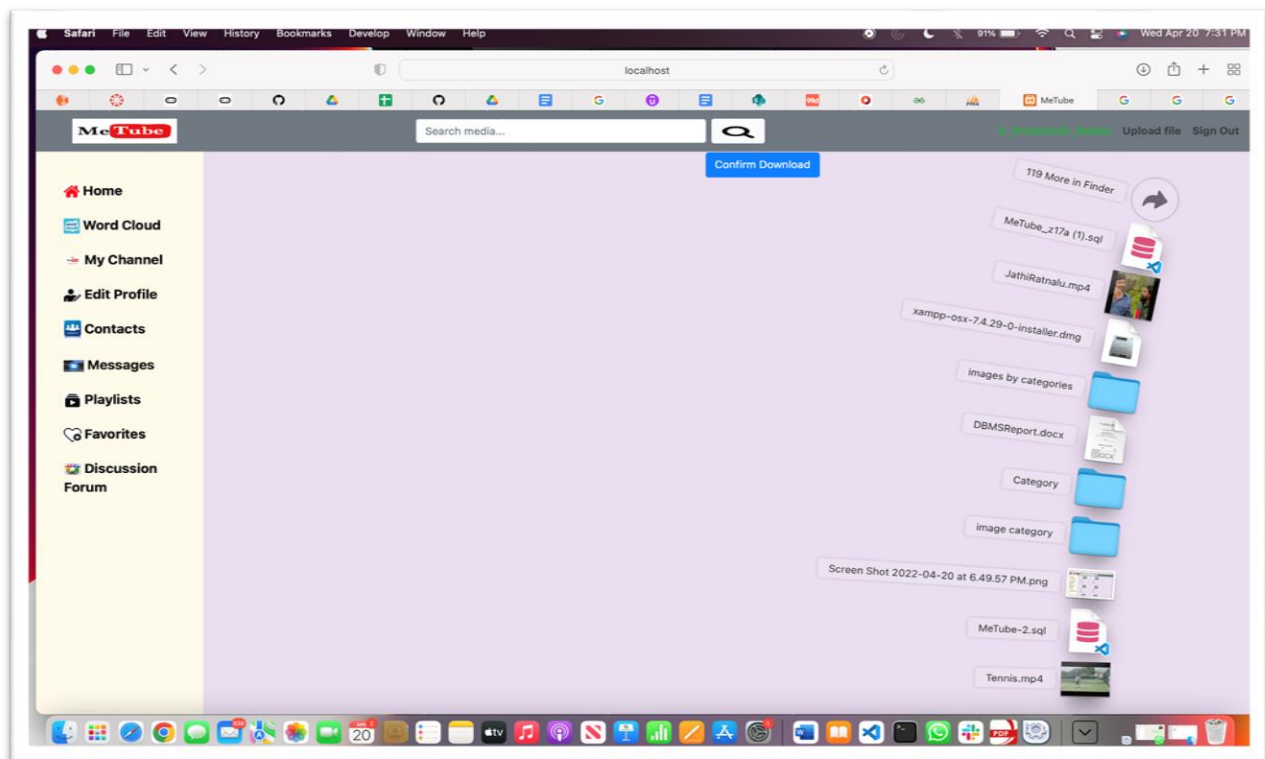
Viewing the Uploaded video by the user and can download the video by clicking on the download button.



b. Download :

The multimedia files in the MeTube system can be downloaded and viewed by any user with a web browser.

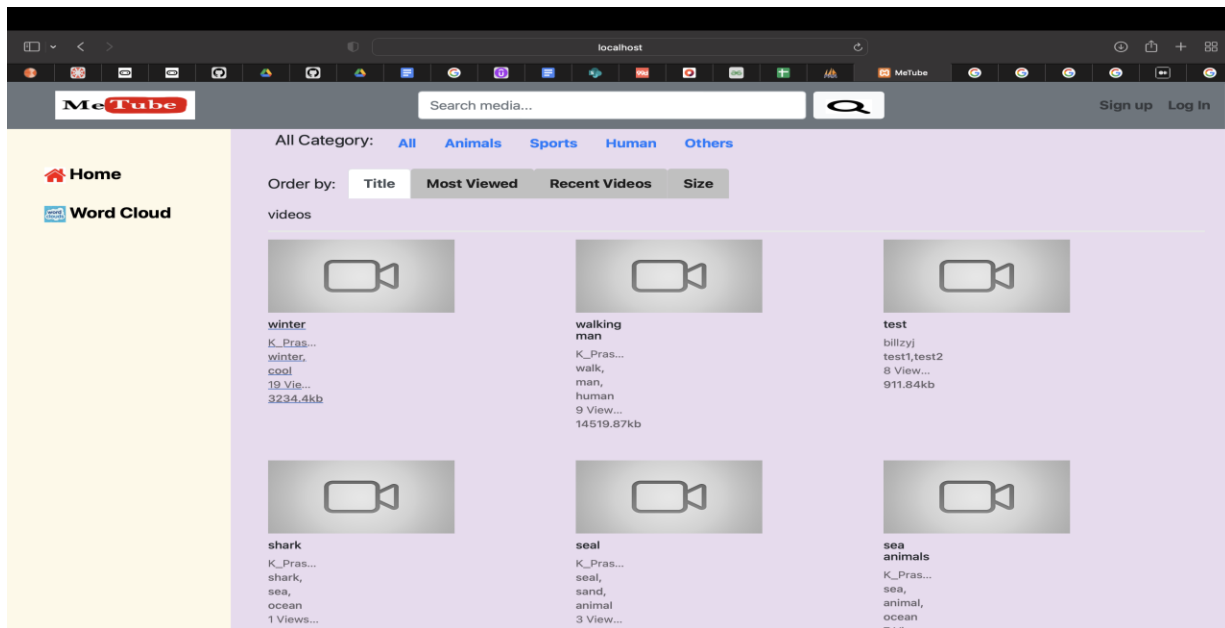
Downloaded the Teninis.mp4 video which was uploaded by User



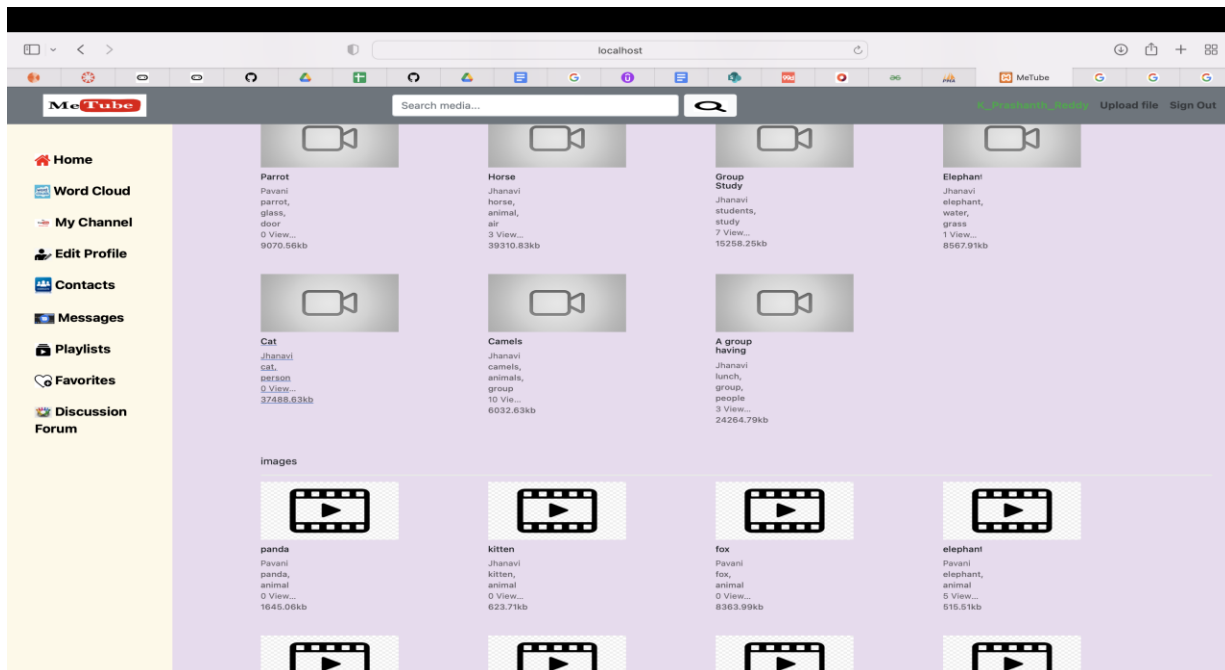
c. View media files :

Regardless of whether they are signed in or not, a user can browse all of the media files in the Metube system by clicking the " All Categoruy" button or searching for it in the search bar MeTube site. When clicked on the video, we'll be taken to a web interface where we can watch the video.

Viewing the Media before login



After Login in to the MeTube website



d. Sharing mode :

Setting the sharing ways for multimedia files that a user uploads (public, family, friends, favorite)

MeTube Search media... Upload file Sign Out

Edit Media information:

Media Title:
winter

Keywords:
winter, cool

Select a category:

- ☐ Sports
- ☐ Animal
- ☐ Human
- ☒ Other

Visibility:

- ☐ Public
- ☒ Family
- ☐ Friends
- ☐ Favorite

Description
winter came

e. Blocking :

Block certain users from viewing or downloading media files that were uploaded by this user.

MeTube Search media... Upload file Sign Out

Contacts

User Name: Pavani

Please select your Relationship:

- ☐ Friend
- ☐ Family
- ☐ Favourite

Add

Please select below action for block/unblock contact:

- ☒ Block
- ☐ Unblock

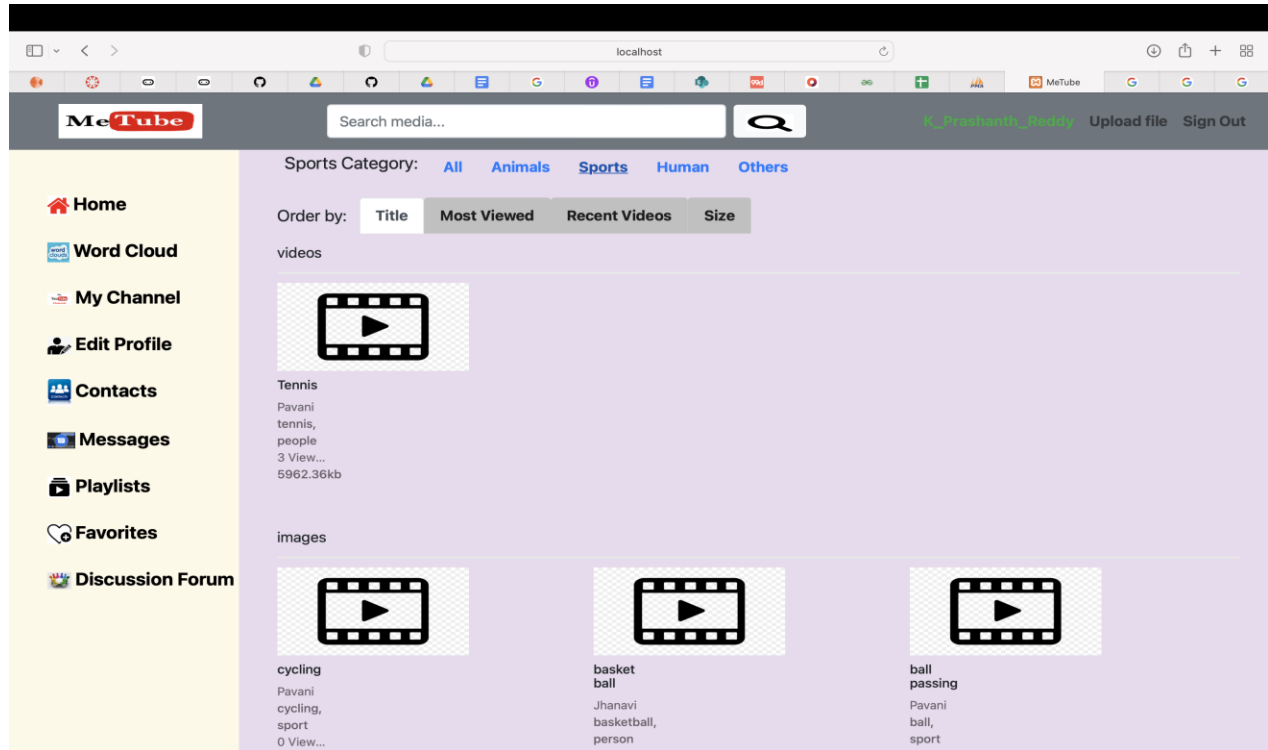
Add

Username	Current Relationship	Block Status
Pavani	Friend	Blocked

3. MEDIA ORGANISATION

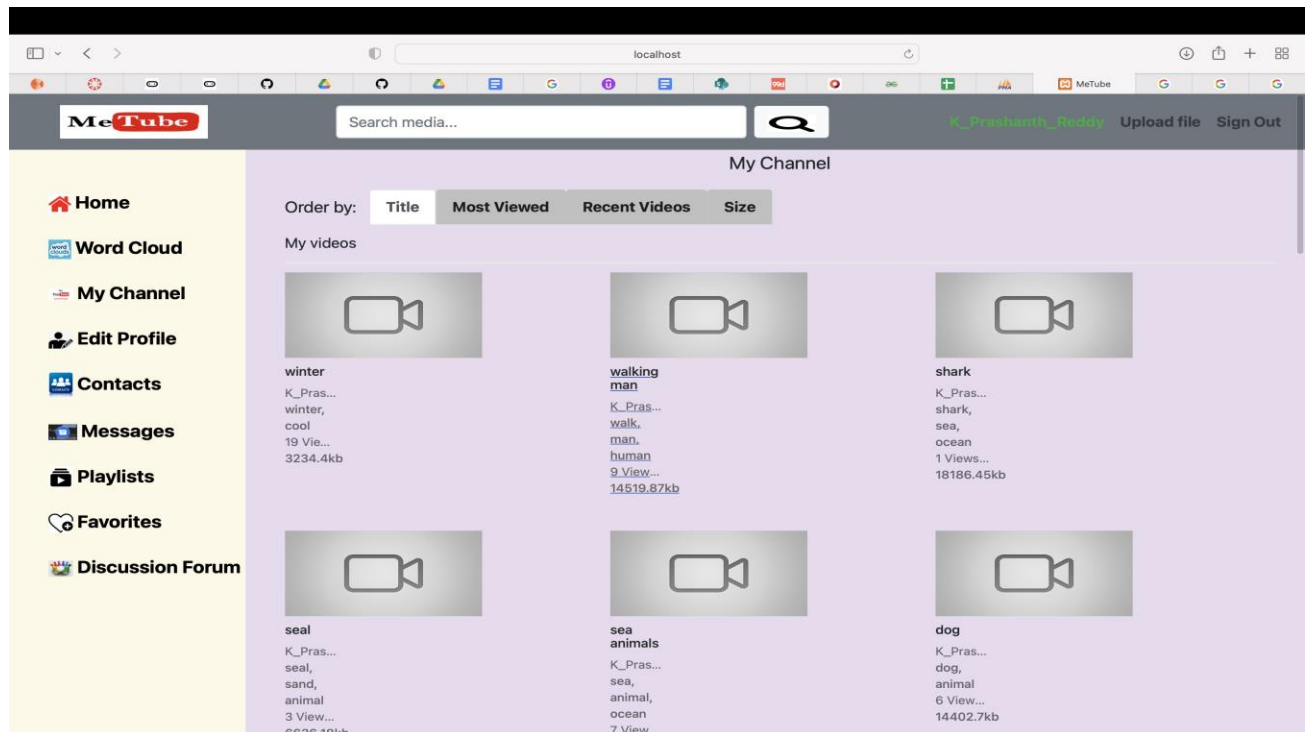
a. Browsing by category :

Users can search for media files by category. Below the search bar, there are different type of categories to choose from, as well as a "all" category.



b. Channels :

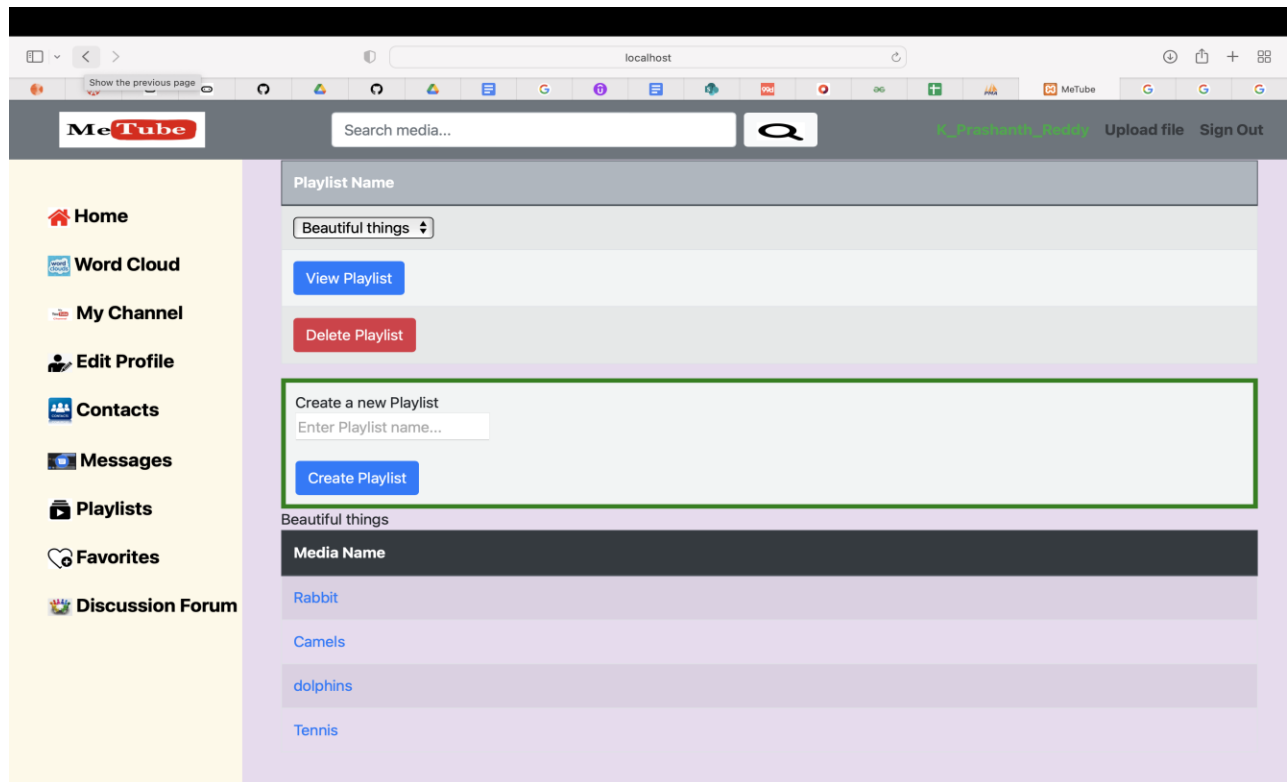
Each user's media files are organized into a channel. At any time, he can view what are the media files uploaded by the user and he can also edit the media info.



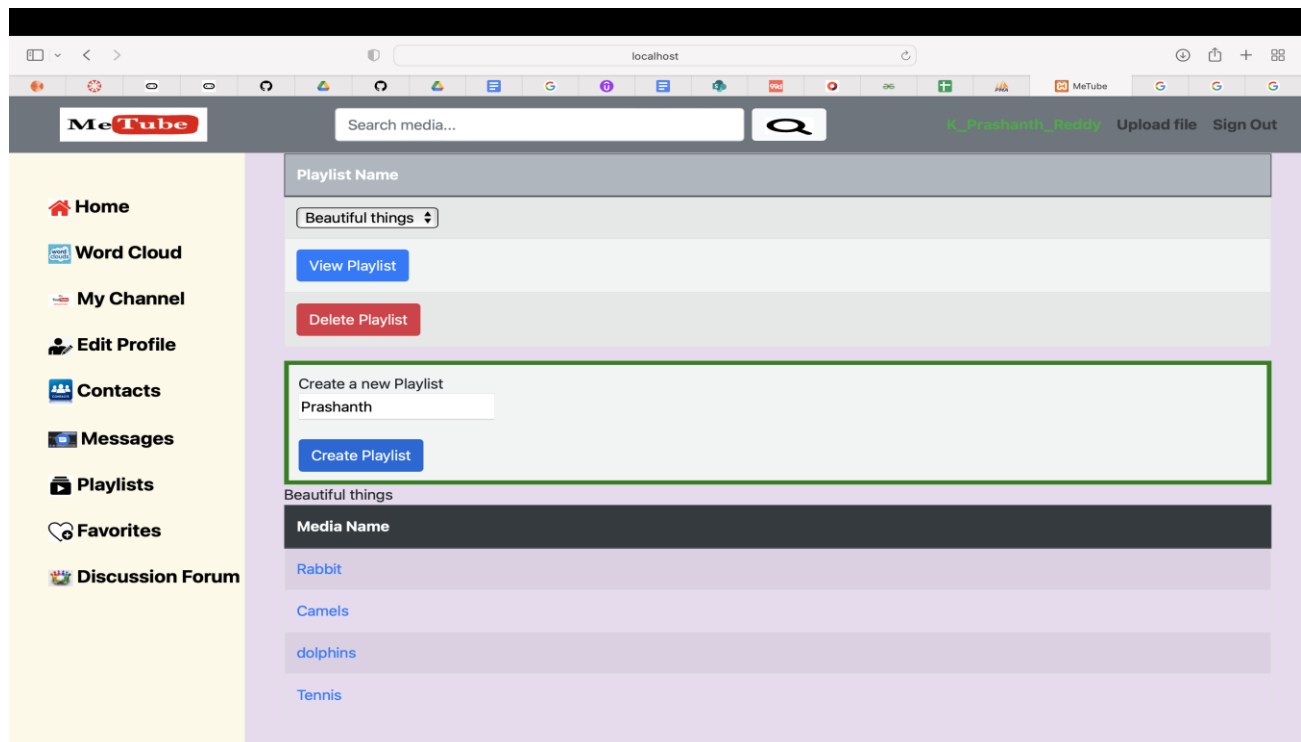
c. Playlists :

In the side navigation on clicking the "Playlists" tab, the user can see all of their playlists. The "Create Playlist" button takes us to make a playlist. Where we have to enter the Playlist name.

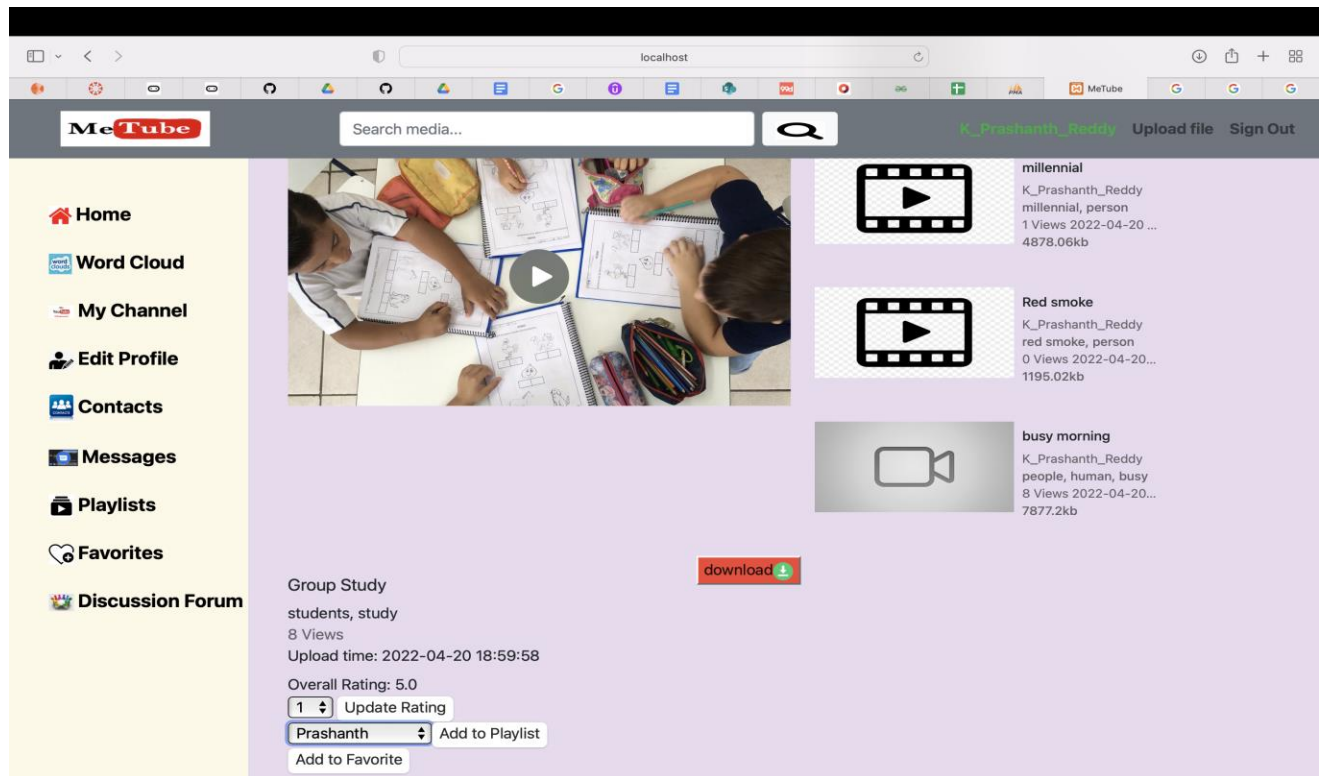
Initially there is only Beautiful things playlist as you can see in the image below.



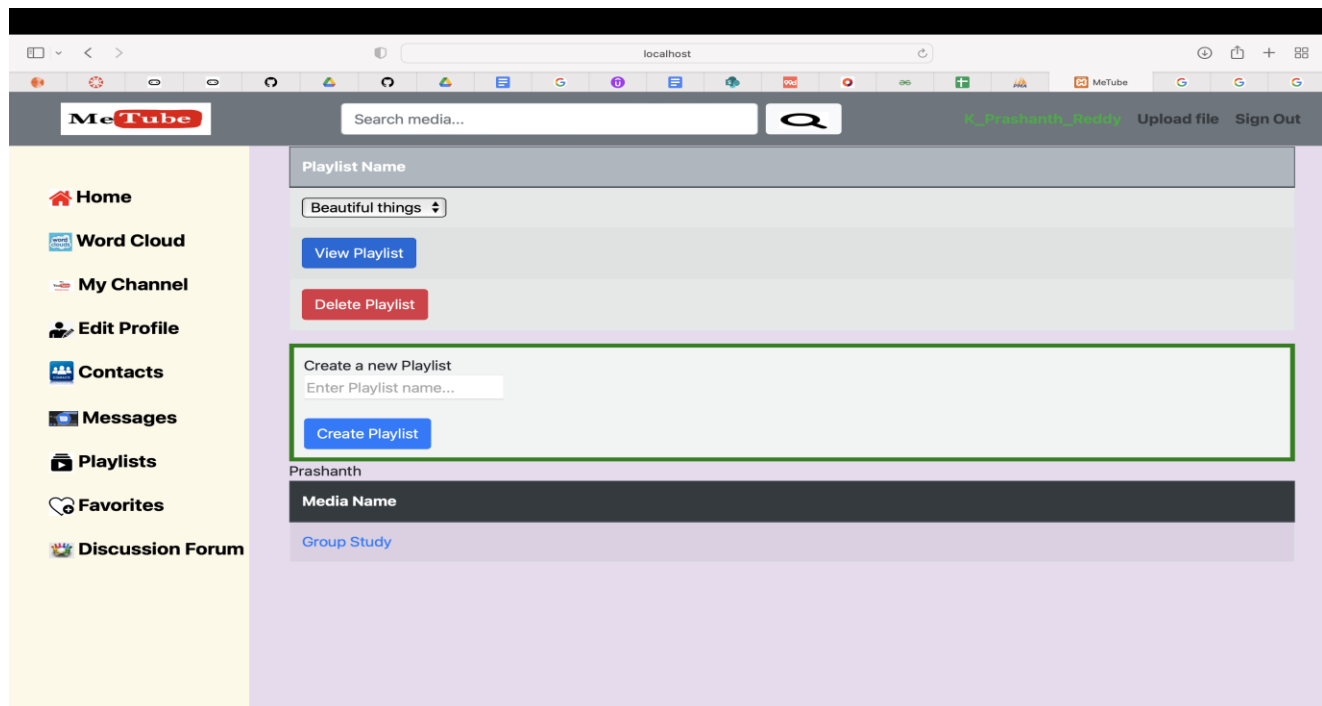
Creating Playlist called Prashanth



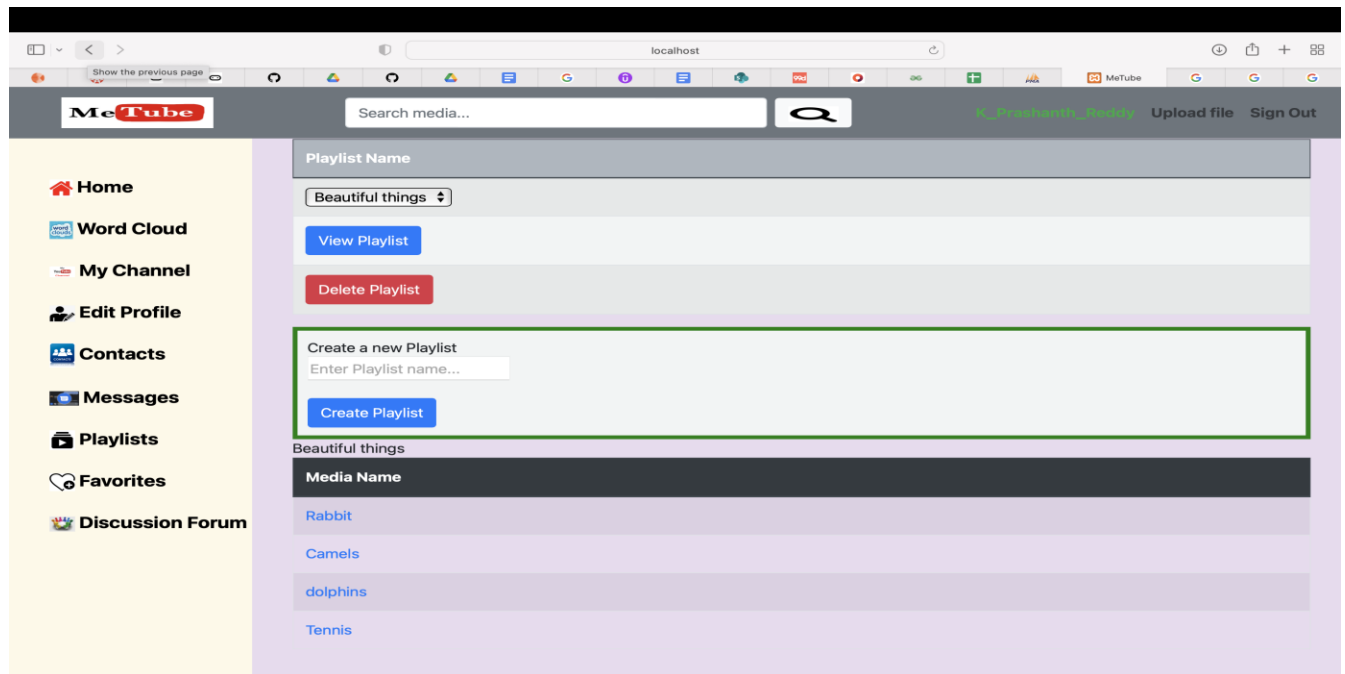
Adding Video to the Prashanth Playlist.



Now, we can see the Group study video in the Prashanth's Playlist



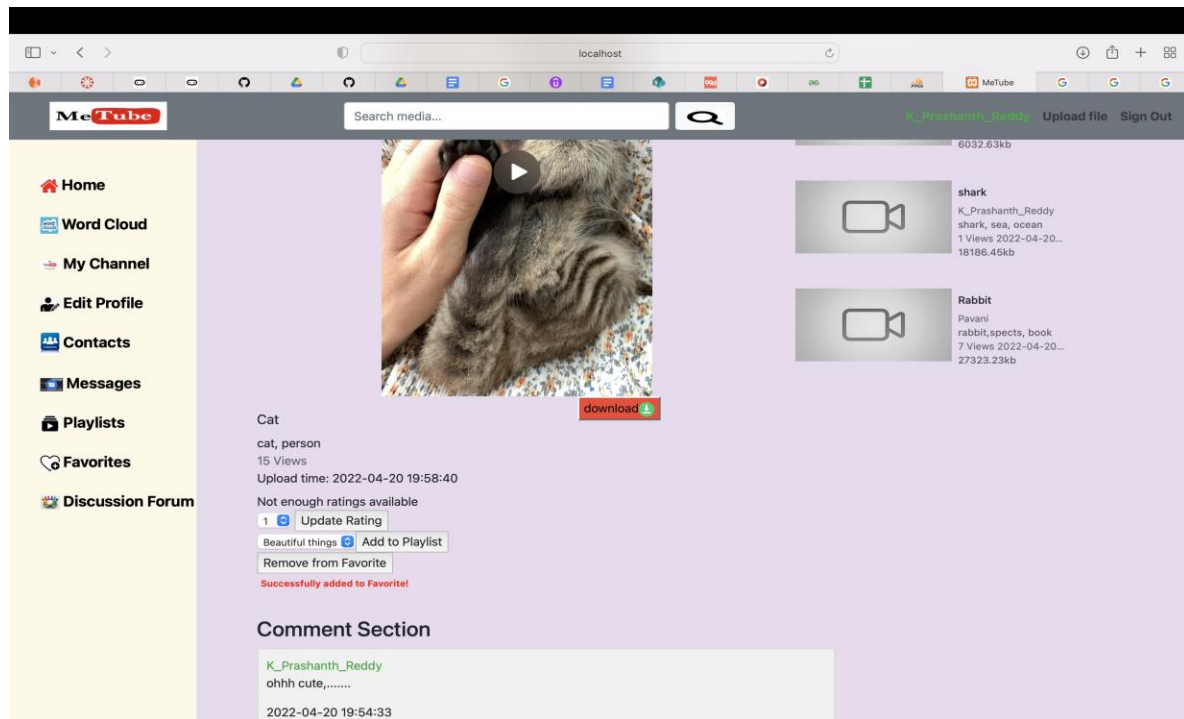
After deleting Prashanth Playlist, remaining Playlist's view.



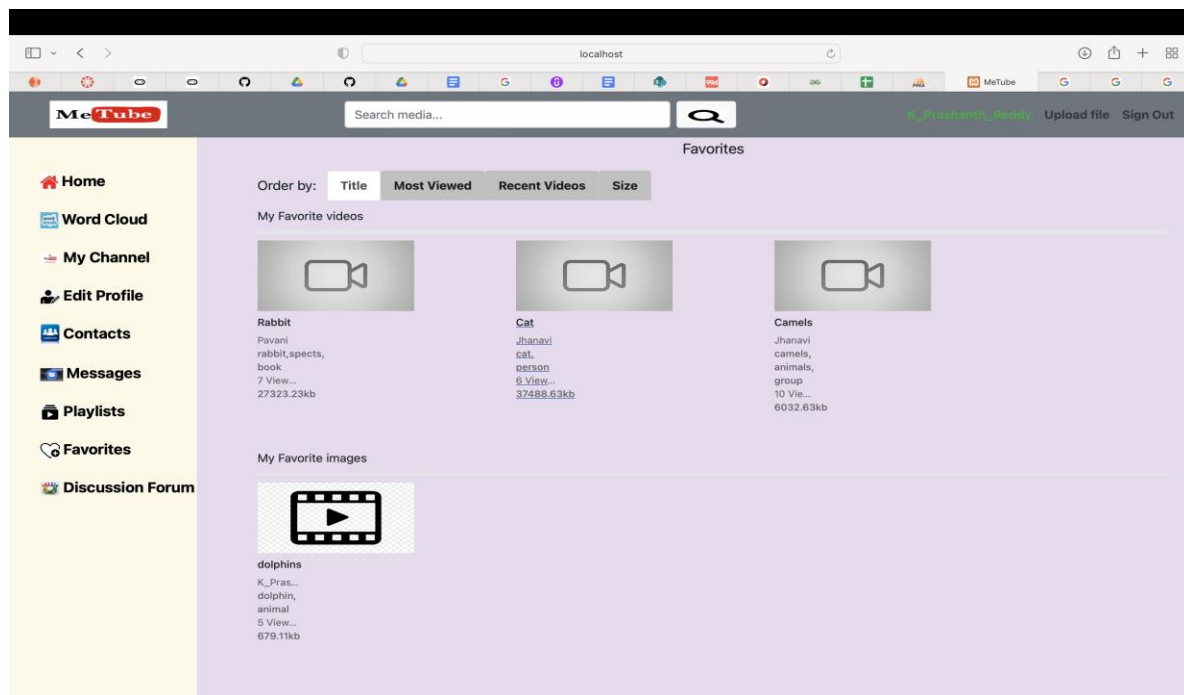
d. Favorite Lists:

In the side navigation on clicking the "Favorites" tab, the user can see all of user Favorite videos. By clicking on the add favorite button, the video will be added to favorites and displays successfully added to favorites in red color.

The user can also access their favorited videos by going to their profile page and clicking on the "Favorites" tab.

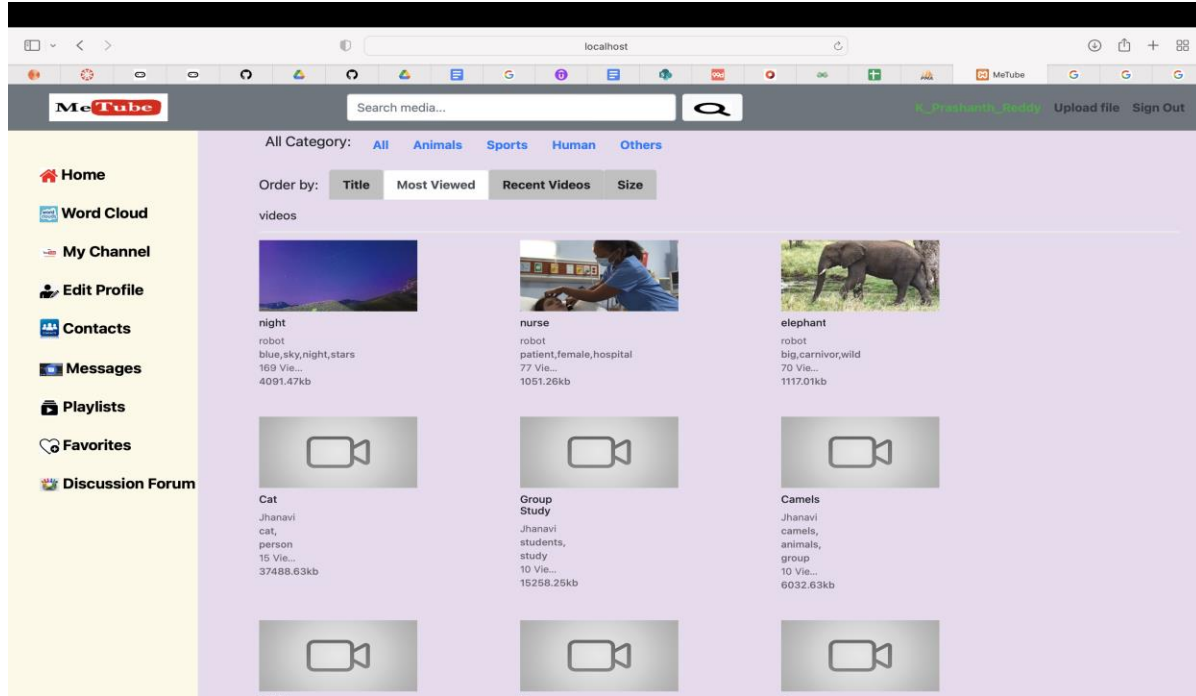


View the Favorites list by clicking on Favorites tab.



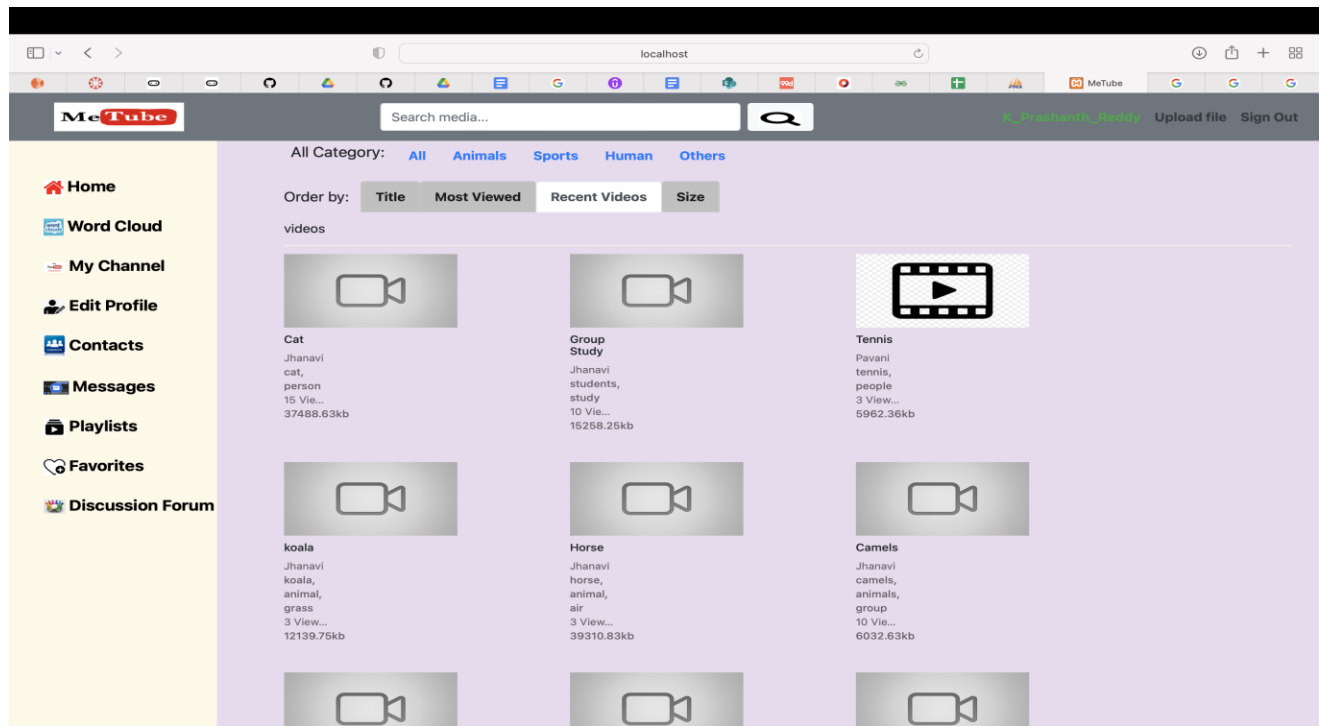
e. Most viewed:

The most popular media files must be displayed on the Metube system's homepage by clicking on the most viewed button .



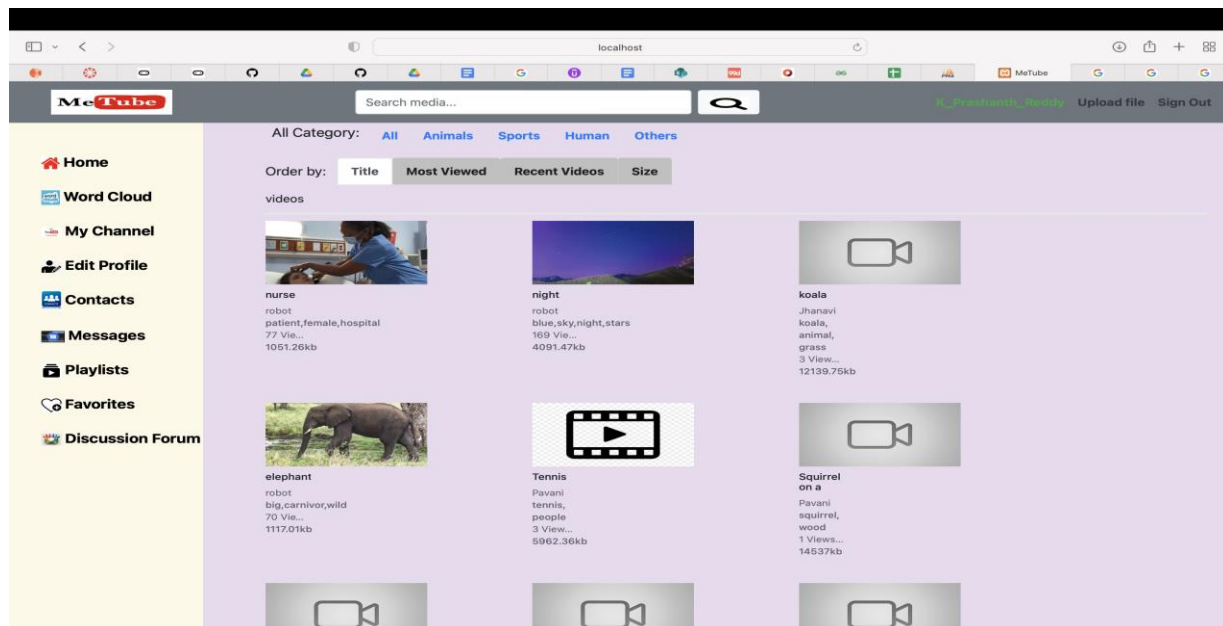
f. Most recently uploaded:

The most recently uploaded media files should be displayed on the Metube system's homepage by clicking on the most recent button.



g. Ordering media files:

We should be able to order the media file lists based on different features when we display the results of videos by Title, Most Viewed, Recent Videos, and size etc..



4. USER INTERACTION

a. Message Exchange between users :

This system allows users to communicate with one another by sending and receiving messages. This is organized in a "Inbox," and "Sent" which the user may access by clicking the Inbox button and sent button. The user will be able to view the messages sent and received.

The screenshot displays the 'Messages' section of the McTube application. On the left is a sidebar with navigation links: Home, Word Cloud, My Channel, Edit Profile, Contacts, Messages, Playlists, Favorites, and Discussion Forum. The main content area is titled 'Messages' and features a form for composing a message to 'K_Prashanth_Reddy'. Below the form are 'Inbox' and 'Sent' buttons. A section titled 'View Messages' shows a list of sent messages.

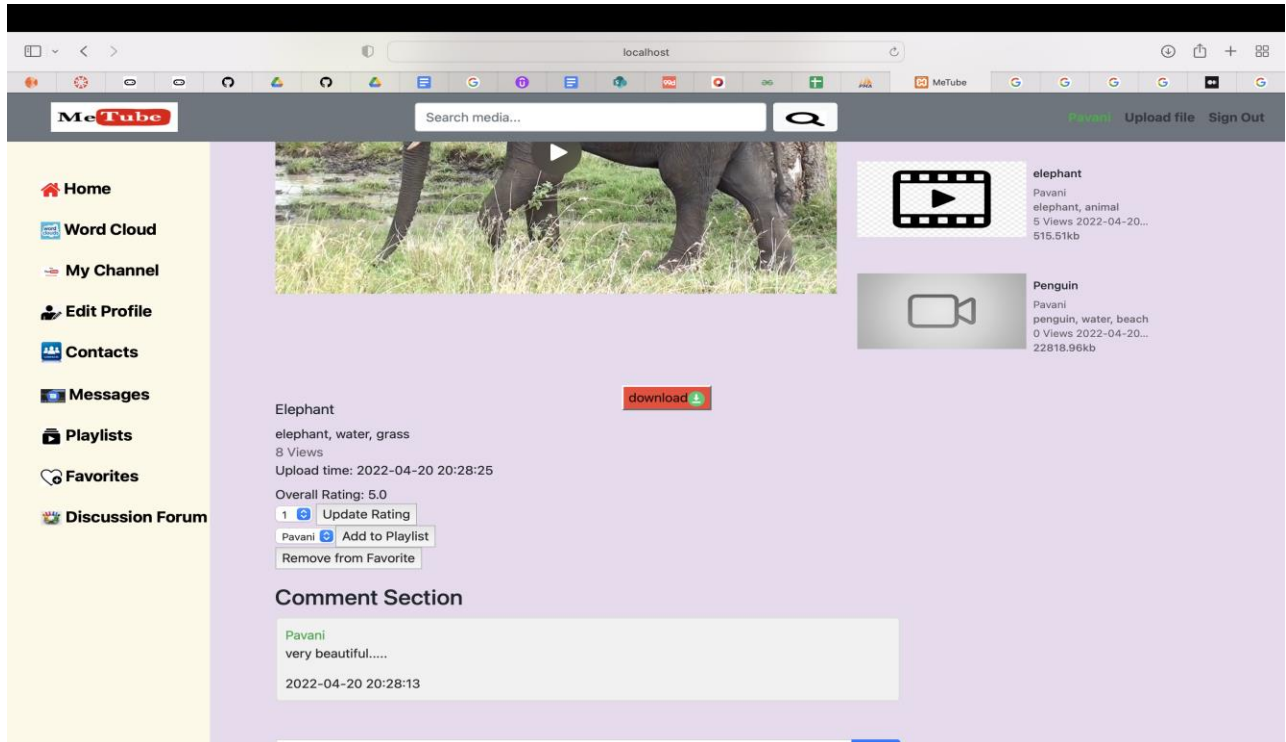
Messages	Sent To	Sent At
Hai Prashanth, how are you?	K_Prashanth_Reddy	2022-04-20 19:00:33
Hai Prashanth, what are the courses you are planning to take for this semester?	K_Prashanth_Reddy	2022-04-20 20:21:16

Each message has four buttons the user can choose from:

1. Select username and type message and click on Message button then it will deliver the message to the other user.
2. Inbox: By clicking on the Inbox button, we can view the messages received from the other users.
3. Sent : By clicking on the Sent button, we can view the messages replied to the other users

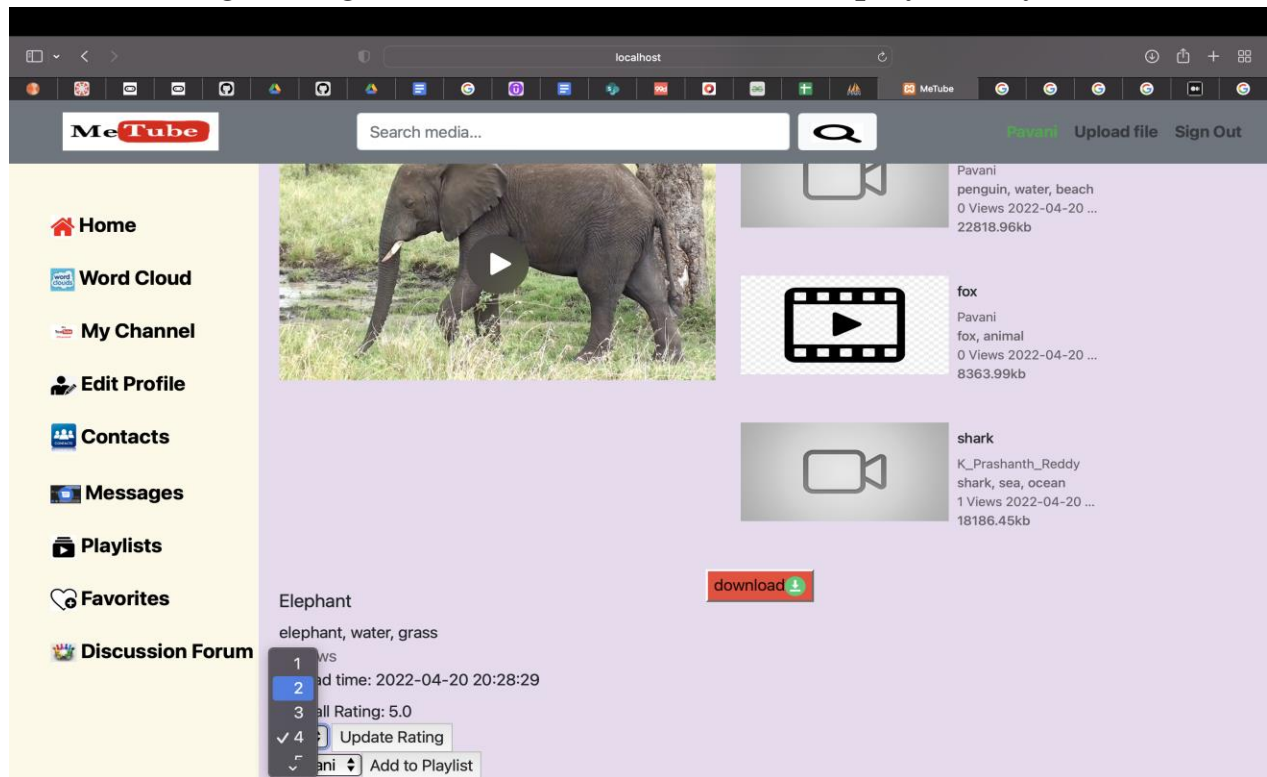
b. Commenting on media file :

This allows users to leave feedback on a video file. A user can leave a response to a specific comment. Under that comment,. Following that, everyone watching the video below it will see that user's comment.



c. Media rating:

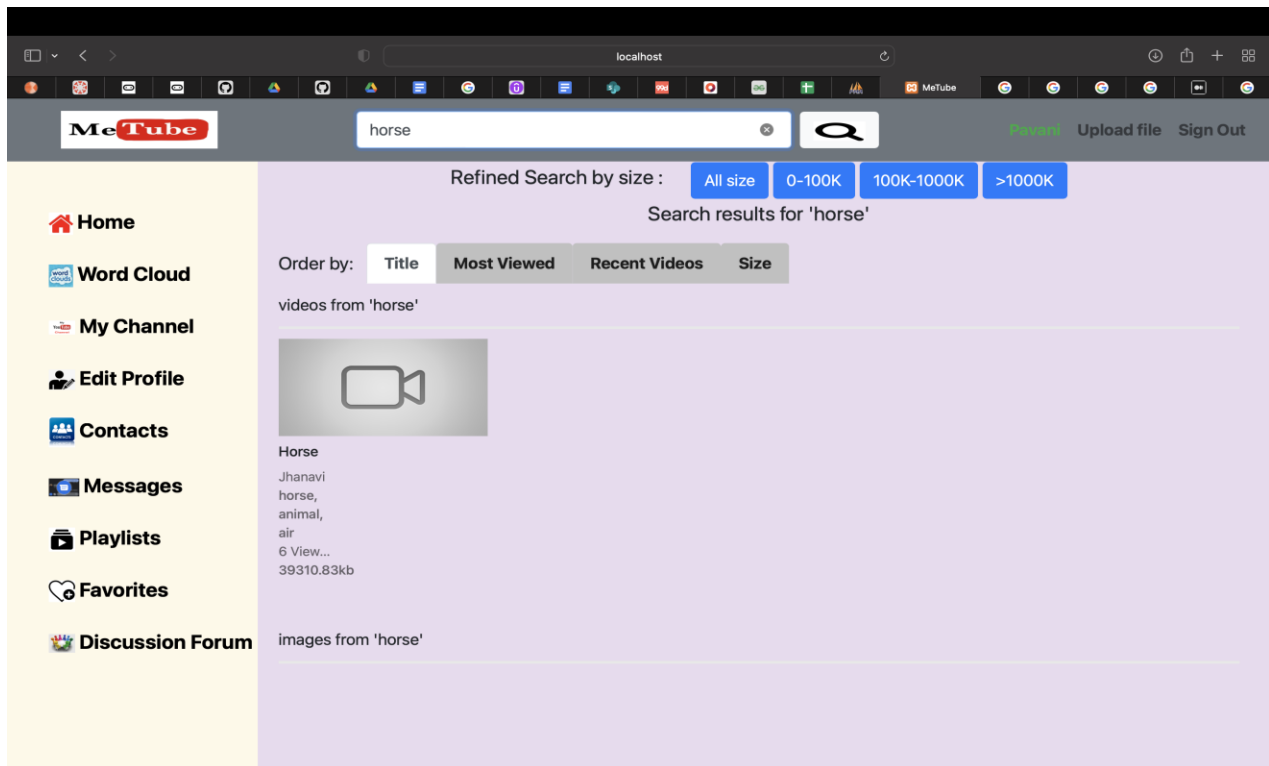
Provide a 5-star rating system with which users can rate the media files. The current average rating of the media file should be displayed on your interface.



5. SEARCH

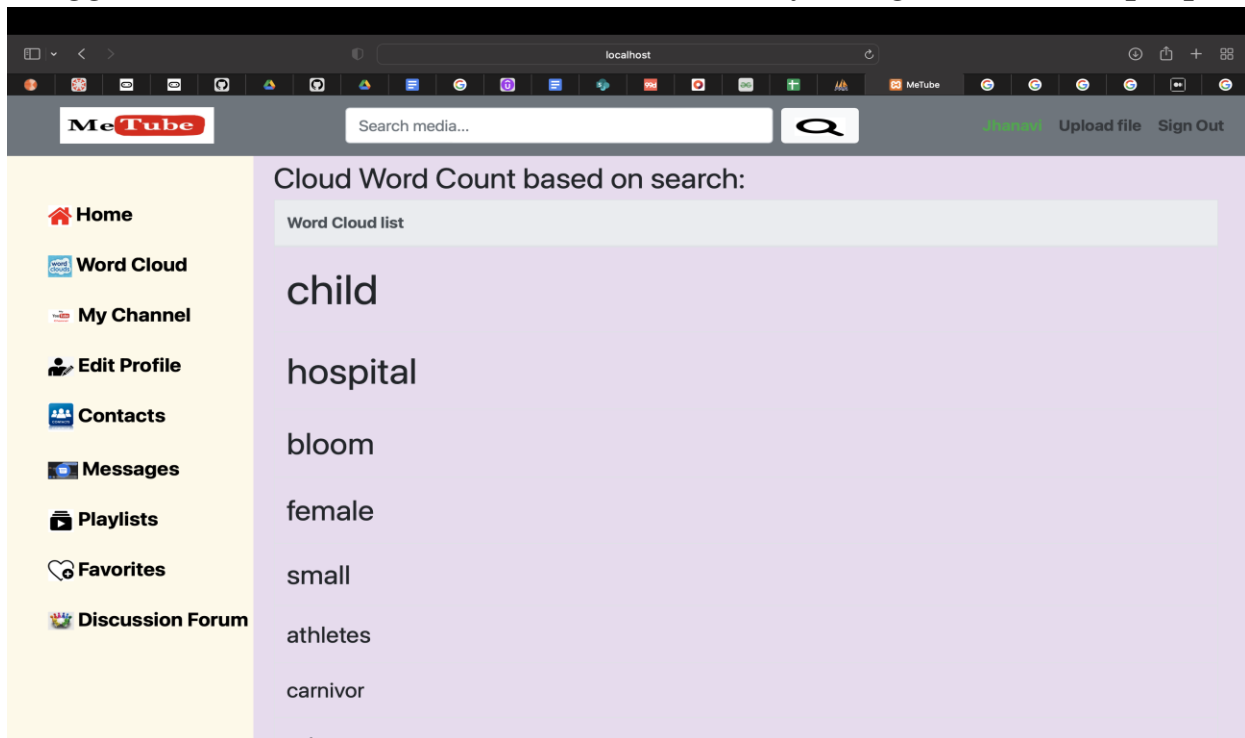
a. Keyword based search :

The user may search for videos using the search box, which is based on terms discovered in the title and key words.



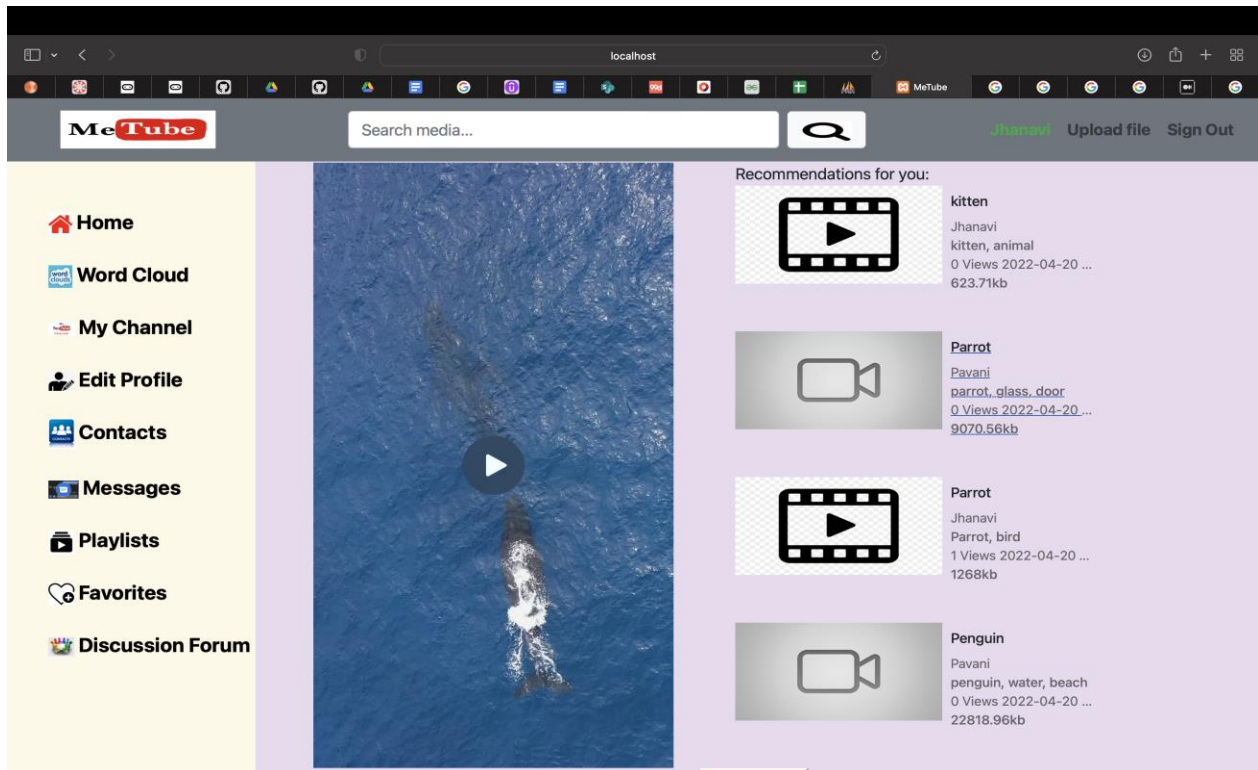
b. Word cloud :

We'll need to create a keyword cloud with various font sizes. A keyword with a bigger font size indicates that it was searched by a larger number of people.



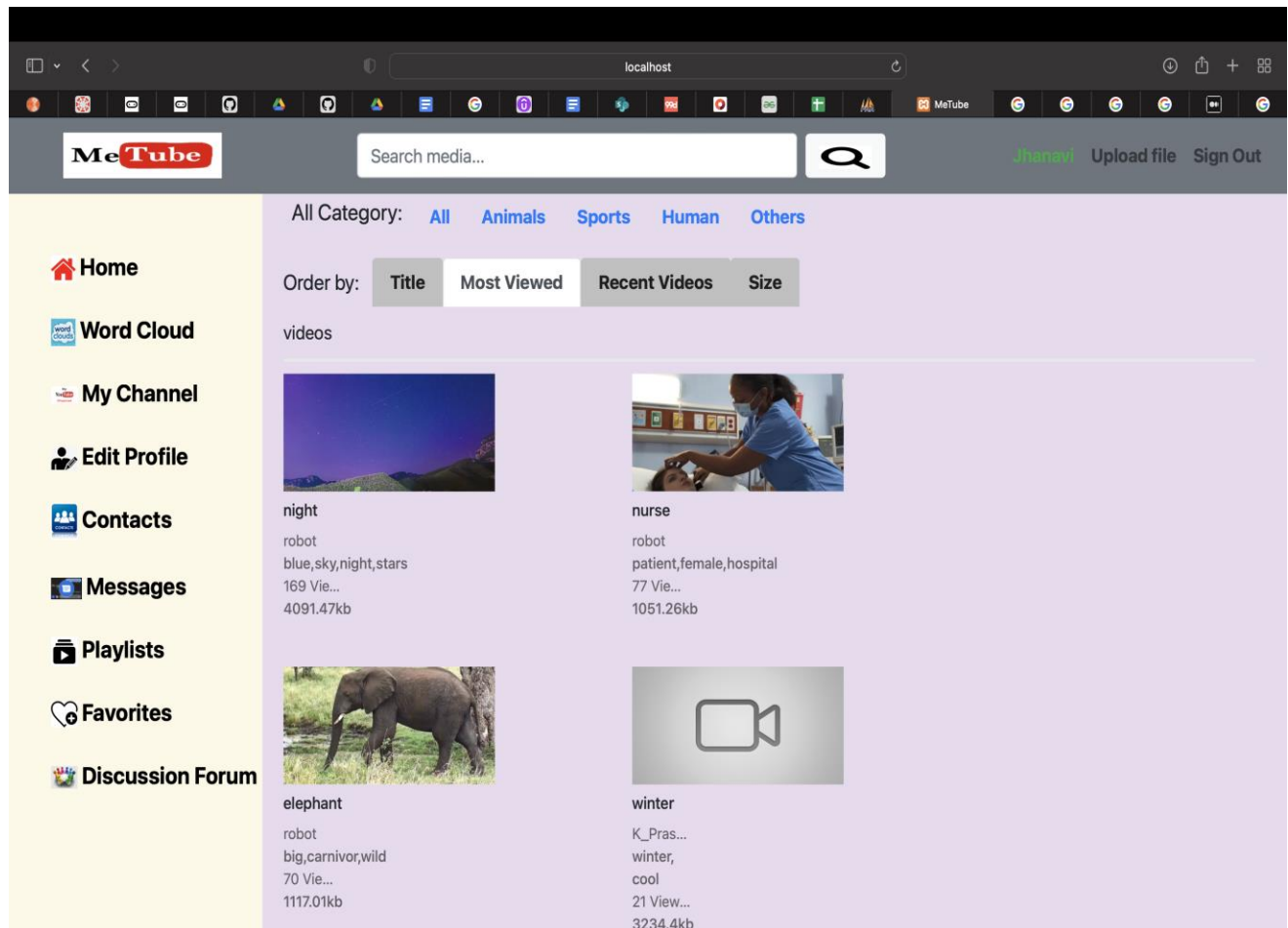
c. Media recommendation

Recommended media will appear beside the media you're presently watching, based on similarities to the current media as determined by various recommendation algorithms.



d. Feature based media search:

Search for media based on some low level features



IMPLEMENTATION DETAIL

We used the following to implement the front end of this project

1. HTML
2. CSS
3. JavaScript.

To implement the backend of the project we used:

1. PHP
2. MySQL

The main project file: MeTube_c6mb > MeTube_c6mb/index.php

The database: MeTube_c6mb

USER MANUAL

1. Users can register for an account on the website by clicking the "Sign Up" button and entering a username, email, password, as well as confirming the password.
2. If the user already has an account, he or she can sign in using the "Log In" option, which requires the user to provide the username and password that were used to register the account.
3. After logging in, the user can choose from the options on the main page to read his or her profile, explore all people, or browse all media.
4. When the user chooses to see the profile, the user will see the profile body, which he or she can alter to suit his or her personality and interests. On top of the profile body, this space also reveals when the profile was created.
5. From here, the user can browse to "Uploads," which shows all of the content that the user has already uploaded to this account. . "All," "Human," "Sports," "Animal," are among the categories accessible. They can also be found by using the search bar.
6. The user can also make their own playlists by going to the "Playlists" menu..
7. Users can click to the "Favorites" tab to see all of the stuff they've designated as favorites through this account.
8. The "Edit Profile" button in the "Edit Profile" tab in the side navbar allows the user to change their password and email.
9. When a user selects the "All Category" option on the main page, they will be able to see all of the available content on the website that has been posted by people with various accounts.
10. To return to the home page, the user can always click on the "MeTube" logo in the top-left corner of the web page or the button in the top-left corner with a home symbol.

11. If the user wishes to search for specific content, he or she can utilize the buttons on the top-right to select "All," "Human," "Sports," "Animals," and then type the keywords in the search box.
12. The user is redirected to a page where he or she can upload content of their choice by clicking on the "Choose File" button and filling in the "Media Title" field, choosing a category for the file, filling in a description for the file, setting up some tags (key words) for the file by entering them in the given field and separating them by a comma, and setting the comments to enabled or disabled for the file. The user can then upload the files by clicking the "Upload" button.
13. Select username and type message and click on Message button then it will deliver the message to the other user. By clicking on the Inbox button, we can view the messages received from the other users. By clicking on the Sent button, we can view the messages replied to the other users.

CONCLUSION

Users can now form an account, log in, post media files such as video, photos, and animation clips, as well as remark on these files, block comments on their own files, send and receive messages from other accounts, and create their own playlists, as explained in this article.