Media Streaming with IBM Cloud Video Streaming

TEAM MEMBER

312621243023:Shalini.J

Phase-1 Document submission

The Objective

project involves creating a virtual cinema platform using IBM Cloud Video Streaming. The objective is to build a platform where users can upload and stream movies and videos on-demand. This project encompasses defining the virtual cinema platform, designing the user interface, integrating IBM Cloud Video Streaming services, enabling on-demand video playback, and ensuring a seamless and immersive cinematic experience.

Introduction

Creating a virtual cinema platform using IBM Cloud Video Streaming is an exciting project that can bring movie nights to life for friends and family, no matter where they are.

Here's a step-by-step guide to set up your virtual cinema platform:

Step 1: Sign up for IBM Cloud Video Streaming

• Go to the IBM Cloud website (https://cloud.ibm.com/) and sign up for an IBM Cloud account if you don't already have one.

• Once you're logged in, navigate to the IBM Cloud Video Streaming service. You can find it in the catalogue of services.

• Create an instance of the IBM Cloud Video Streaming service. Follow the setup instructions provided by IBM.

Step 2: Prepare Your Movies and Videos

• Ensure that you have the digital copies of your favourite movies and videos ready for upload. You may need to convert them into compatible formats if necessary.

Step 3: Upload Content to IBM Cloud Video Streaming

• Access your IBM Cloud Video Streaming dashboard.

• Create a new channel or event for your movie night. Give it a name, description, and set the access permissions (public or private).

• Upload your movies and videos to the channel or event. IBM Cloud Video Streaming provides options for uploading and managing your content within the platform.

Step 4: Set Up Streaming

• Configure your streaming settings. You can choose the video quality, bitrates, and adaptive streaming options to ensure the best possible viewing experience.

• Generate stream keys and URLs for your content. These will be used to access and play the movies and videos.

Step 5: Share the Virtual Cinema Experience

• Share the stream links with your friends and family. They can access the movies and videos from anywhere with an internet connection.

Step 6: Enhance the Experience

• Consider adding additional features to enhance the virtual cinema experience, such as live chat for viewers to discuss the movie in real-time or integrating a schedule for upcoming movie nights.

Step 7: Monitor and Manage

• Use the monitoring and analytics tools provided by IBM Cloud Video Streaming to track the performance of your virtual cinema platform. You can see how many people are watching, viewer engagement, and other useful metrics.

• Manage access permissions, security, and any user-generated content as needed to maintain a safe and enjoyable environment for all viewers.

Step 8: Enjoy Movie Nights

• Gather your friends and family virtually and start enjoying movie nights together. The

platform should provide a seamless streaming experience with high-quality video playback for an immersive cinematic experience.

Design thinking:

1.Platform Definition:

- **User Registration:**

• Users can create accounts with their email addresses or social media profiles.

• Registration includes basic profile information and password setup.

• Users can choose to receive notifications and updates.

- **User Profiles:**

• Users can customize their profiles with avatars and personal information.

• Profiles display user's uploaded videos, watch history, and favorite movies.

- **Video Upload:**

• Users can easily upload movies and videos from their devices.

• Supported video formats include MP4, AVI, MKV, and more.

• Users can provide movie details such as title, description, genre, and release year.

- **Content Management:**

• Users have access to a dashboard for managing their uploaded content.

• Options to edit video details, delete videos, or set video privacy (public or private).

• Bulk upload and video scheduling capabilities for content creators.

- **Search and Discovery:**

• A search bar allows users to find movies and videos quickly.

• Content recommendations based on user preferences and viewing history.

• Filter options by genre, release date, and popularity.

• **On-Demand Streaming:**

• Users can select movies from the catalog for on-demand streaming.

• Seamless playback with adaptive streaming for varying internet speeds.

• Options to pause, rewind, fast forward, and adjust video quality.

• **Live Events:**

• Support for hosting live virtual movie screenings with scheduled start times.

• Chat and comment features for real-time interaction during live events.

• Record and make live events available for on-demand viewing.

• **Social Integration:**

• Share movies and events with friends via social media links.

• Social login and sharing options for increased user engagement.

2.User Interface Design:

• **Homepage:**

• Featured movies and live events.

• Quick access to user profiles and settings.

• **Movie Details Page:**

• Movie poster, title, description, and genre.

• Video player with playback controls.

• Related movies and user ratings/reviews.

• **User Profile:**

• User's uploaded videos and playlists.

• Watch history and favorite movies.

• Profile customization options.

• **Search and Browse:**

• User-friendly search bar.

• Filter options and sorting by various criteria.

• Clear movie thumbnails and intuitive navigation.

• **Video Upload Page:**

• Step-by-step upload process with progress indicators.

• Options for adding video details and privacy settings.

• Drag-and-drop or file selection for uploading.

3.Video Upload:

• **Upload Wizard:**

• Guided step-by-step process for uploading videos.

• Allows users to add multiple videos in a single session.

• Progress bars and confirmation upon successful upload.

• **Video Management:**

• Dashboard for managing uploaded videos.

• Thumbnail customization and video title/description editing.

• Privacy settings for controlling who can view the content.

Streaming Integration:

• IBM Cloud Video Streaming:

• Integration of IBM Cloud Video Streaming services for hosting and streaming content.

• Utilize adaptive streaming for various internet speeds.

• Implement secure streaming protocols and content protection.

4.User Experience:

• High-Quality Playback:

• Ensure smooth and buffer-free playback.

• Support for HD and 4K video resolutions.

• Adaptive streaming for optimal quality based on user's internet connection.

• Interactive Features:

• Real-time chat during live events.

• User ratings and reviews for movies.

• Personalized recommendations based on viewing history.

• Responsive Design:

• User interface should be accessible on various devices (PCs, tablets, smartphones).

• Ensure responsive design for a consistent experience.

• Notifications:

• Send notifications for live events, new uploads, and user interactions.

• Allow users to customize notification preferences.

• **Security and Privacy:**

• Implement robust security measures to protect user data and content.

• Enable content owners to set privacy levels for their videos.

• **Feedback and Support:**

• Provide a feedback mechanism for users to report issues or suggest improvements.

• Offer customer support channels for user inquiries and assistance.

• **Accessibility:**

• Ensure the platform is accessible to users with disabilities.

• Implement features like closed captions and screen reader support.

Conclusion

Virtual cinema platform can provide an engaging and immersive movie-watching experience for users while leveraging IBM Cloud Video Streaming for reliable content delivery.

Code :

```
#Replace with your IBM Cloud video streaming credentials
api_key ="YOUR_API_KEY"
channel_id="YOUR_CHANNEL_ID"


#Replace with the URL of the media you want to stream
media_url="https://example.com/media.mp4"


#Create a live event
response=requests.post(f"https://api.video.ibm.com/channels/{channel_id}/events",
   Headers={"Authorization": api_key},
   json={"event_type":"live","start",:true},
)
event_id=response.json()["event"]["id"]


#upload media file
response=request.post(f"https://api.video.ibm.com/channels /{channel_id}/events/
{event_id}/uploads",)
headers={"Authorization": api_key},


json={"url": media_url},
)
upload_id = response.json()["upload"]


["id"]


# Start the live stream
response = requests.post(
f"https://api.video.ibm.com/channels /{channel_id}/events/
{event_id} /broadcasts",
 headers={"Authorization": api_key},
 json={"upload_id": upload_id},
)
broadcast_id = response.json()
["broadcast"]["id"]


# Get the stream URL
```

```
response = requests.get(
 f"https://api.video.ibm.com/channels /{channel_id}/events/{event_id}/
broadcasts
/{broadcast_id}/stream_url",
headers={"Authorization": api_key},
)
stream_url = response.json()["stream_url"]



# Print the stream URL
 print("Stream URL:", stream_url)
```

Make sure to replace "YOUR_API_KEY", "YOUR_
CHANNEL_ID", and "https:// example.com/media.mp4" with your own credentials and media URL