CLOUD APPLICATION DEVELOPMENT(CAD)

**Project:** Media Streaming with IBM Cloud video Streaming

**Phase 2:** Innovation

Define Requirements:

Clearly outline the features, such as user profiles, playlists, real-time chat, and notifications.

Determine the technology stack you'll be using, such as front-end frameworks (React, Angular), back-end frameworks (Node.js, Django), and database systems (MongoDB, PostgreSQL).

Design the Database:

Create a database schema to store user information, playlists, movies/shows data, chat messages, and other relevant details.

Establish relationships between different entities, such as users and playlists, movies/shows and playlists, etc.

Develop User Authentication and Profiles:

Implement user registration and login functionality with secure password storage and authentication methods.

Create user profiles where users can add avatars, bios, favorite genres, and other personal details.

Implement User-Generated Playlists:

Allow users to create, edit, and delete playlists.

Provide options to add movies/shows to playlists, and ensure these associations are stored in the database.

Integrate Real-Time Chat:

Choose a real-time communication technology like WebSockets to enable instant messaging between users.

Implement public chat rooms for discussions related to movies/shows and private messaging for one-on-one conversations.

Implement message history, emoji reactions, and basic moderation tools.

Develop Movie/Show Pages:

Design individual pages for movies/shows displaying details, play options, discussion sections, and related content suggestions.

Implement features like adding to playlist, real-time chat, user ratings, and reviews.

Set Up Notifications:

Implement a notification system to alert users about new messages, playlist updates, and friend activities.

Use push notifications or WebSocket notifications to ensure real-time updates.

Implement User Interactions:

Enable users to follow/unfollow others and manage their friend list.

Implement actions like liking playlists, movies/shows, and comments.

Implement Security Measures:

Ensure data security by using encryption and secure communication protocols.

Implement user authorization to control access to various features and data.

Test Thoroughly:

Perform extensive testing to identify and fix bugs.

Test the platform's usability and responsiveness on different devices and browsers.

Launch and Gather Feedback:

Deploy the platform to a web server or a cloud service.

Gather user feedback and make necessary improvements based on the feedback received.

**Homepage:**

Featured Movies/Shows:

Display a selection of popular movies and shows.

User-Generated Playlists:

Showcase curated playlists created by users based on genres, moods, or themes.

Trending Discussions:

Highlight ongoing discussions about trending movies or shows.

Personalized Recommendations:

Provide movie/show suggestions based on users' watch history and preferences.

**User Profiles:**

Create Profile:

Users can create profiles with avatars, bio, and favorite genres.

My Playlists:

Users can create, edit, and share their playlists. They can add movies/shows from the platform to their playlists.

Watch History:

Show a list of movies/shows recently watched by the user.

**Movie/Show Pages:**

Play Button:

Starts the movie/show.

Add to Playlist:

Users can add the movie/show to their playlists.

User Ratings and Reviews:

Allow users to rate and review movies/shows.

Related Content:

Suggest similar movies/shows based on the current selection.

**Real-Time Chat:**

Public and Private Chats:

Users can engage in public discussions related to specific movies/shows or initiate private chats with friends.

Emoji Reactions:

Users can react to messages with emojis to express their emotions quickly.

Moderation Tools:

Implement tools to moderate chats, ensuring a positive and respectful environment.

Notification System:

Notify users when they receive new messages or when someone replies to their comments.

**Notifications:**

New Playlist Updates:

Notify users when someone adds their movie/show to a playlist.

Discussion Alerts:

Notify users about ongoing discussions related to movies/shows they've watched or are interested in.

Friend Activity:

Notify users about their friends' activities, such as creating playlists or starting discussions.

**Settings:**

Privacy Controls:

Allow users to control who can message them and who can see their activities.

Notification Preferences:

Enable users to customize their notification settings based on their preferences.

Account Security:

Implement two-factor authentication and account recovery options.

By incorporating these features, MovieHub can create an engaging, social, and personalized movie-watching experience, enhancing user interaction and satisfaction on the platform.