Project -FocusTube

****

****

Project Name: FocusTube

Version 1.0

Date: [05/03/2025]

Prepared by:

Md. Kamrul Hasan

Roll : 79

Section : A

Lab Group : Q

Reviewed by:

Mr. Md. Monir Hossain

Assistant Professor

Dhaka City College

****

****

Requirements:

- Users can paste a YouTube playlist link/ID to load a playlist.
- Users can save playlists for future viewing.
- Users can mark playlists as favorites.
- Users can track the last-watched playlist.
- Show an alert prompting users to enter a playlist ID.
- Check if the playlist ID is already in cache before making an API call.
- Fetch data from the YouTube API and store it in cache if not already available.

- Add a refresh button to update the cached playlist data (limited to once every 24 hours).
- Embedded YouTube player will play videos in order.
- No recommended videos or YouTube distractions.
- Users can control playback (play, pause, next, previous).

Software Requirements Specification (SRS) for FocusTube

1. Introduction

1.1 Purpose

FocusTube is a distraction-free YouTube course viewer designed to help users watch educational playlists without interruptions. It allows users to save, manage, and track their learning progress efficiently.

1.2 Scope

FocusTube provides an ad-free, distraction-free interface where users can:

- Paste a YouTube playlist link/ID and watch videos without recommendations or unrelated content.
- Save playlists for later access.
- Mark playlists as favorites.
- Track the last-watched playlist.
- Cache fetched playlists to reduce unnecessary API calls.
- Refresh cached playlists with a time-limited refresh button.

1.3 Definitions, Acronyms, and Abbreviations

- **YouTube API:** Interface for fetching video and playlist data from YouTube.
- **MUI (Material UI):** A React component library for UI design.
- **React-Router:** Library for managing navigation within the React application.
- **localStorage:** Browser storage for saving user data.
- **Cache:** Temporary storage of fetched API data to minimize redundant requests.

1.4 References

- YouTube API Documentation: https://developers.google.com/youtube/v3
- React Documentation: https://reactjs.org
- MUI Documentation: https://mui.com

---

2. Overall Description

2.1 Product Perspective

FocusTube is a single-page web application (SPA) built using React. It enhances the YouTube learning experience by removing distractions and providing a structured way to track progress.

2.2 User Characteristics

- **Students & Learners:** Those who want to focus on educational content.
- **Professionals:** Individuals following professional development courses.
- **Content Consumers:** Users who prefer a clean and distraction-free experience while watching playlists.

2.3 Constraints

- Requires an active internet connection.
- Limited to YouTube playlists (no standalone video support initially).
- API rate limits from YouTube may affect usage.
- Cached data should be refreshed only within a controlled time frame (e.g., once every 24 hours).

2.4 Assumptions and Dependencies

- The YouTube API will remain available and free to use within its quota limits.
- Users will have valid YouTube playlist links.
- The application will be accessed via modern browsers supporting React and localStorage.

---

3. Specific Requirements

3.1 Functional Requirements

3.1.1 Playlist Management

- Users can paste a YouTube playlist link/ID to load a playlist.
- Users can save playlists for future viewing.
- Users can mark playlists as favorites.
- Users can track the last-watched playlist.
- Show an alert prompting users to enter a playlist ID.
- Check if the playlist ID is already in cache before making an API call.
- Fetch data from the YouTube API and store it in cache if not already available.
- Add a refresh button to update the cached playlist data (limited to once every 24 hours).

3.1.2 Video Player

- Embedded YouTube player will play videos in order.
- No recommended videos or YouTube distractions.
- Users can control playback (play, pause, next, previous).

3.1.3 Data Storage

- Playlists and favorites will be stored in **localStorage**.
- Last-watched playlist will be automatically updated.
- Cached playlists will be stored locally to minimize redundant API requests.

3.1.4 Navigation

- Home page for pasting playlist links.
- Playlist viewer page for watching videos.
- Saved playlists page for managing saved content.

3.2 Non-Functional Requirements

- **Performance:** The app should load playlists quickly with minimal delay.
- **Usability:** Simple and intuitive UI using MUI components.
- **Scalability:** Should handle multiple playlists without performance issues.
- **Reliability:** App should function without crashing even with multiple saved playlists.
- **Security:** User data should be stored locally without third-party access.

---

4. System Design & Technology Stack

4.1 Technology Stack

- **Frontend:** React.js
- **UI Library:** MUI
- **State Management:** `useState`, `useEffect`
- **Routing:** React-Router
- **Data Storage:** localStorage
- **External API:** YouTube API

4.2 Architecture Diagram (To be added)

---

5. Future Enhancements

- **Progress Tracking:** Users can see how much of a playlist they have completed.
- **Custom Tags:** Users can categorize their saved playlists.
- **Dark Mode:** UI theme customization.
- **Multi-Device Sync:** Sync playlists using a backend (Firebase or other DB).

---

6. Conclusion

FocusTube aims to improve the learning experience by removing distractions from YouTube. By leveraging React, YouTube API, and localStorage, it provides a seamless way for users to focus on

their courses. Future improvements will enhance user engagement and usability.

7. References

- YouTube API Docs: https://developers.google.com/youtube/v3
- React Documentation: https://reactjs.org
- MUI Documentation: https://mui.com