

# Project -FocusTube

\*\*\*\*\*  
\*\*\*\*\*

## Project Name: FocusTube

**Version 2.0.1**

**Updated Date:** [ 09/07/2025 ]

### 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 aimed at helping users watch and manage educational playlists with minimal interruptions. This update includes advanced form handling, validation, video playback enhancements, and improved state management.

### 1.2 Scope

FocusTube provides a clean, ad-free interface with the following features:

- Paste YouTube playlist link/ID and watch videos distraction-free.
- Save, favorite, and track playlists.
- Use easy-peasy for managing app state.
- Forms handled with react-hook-form and validated using yup.
- Videos played using react-youtube-player with minimal branding and distractions.

### 1.3 Definitions

- **easy-peasy**: Simplified state management for React.
  - **react-hook-form**: Lightweight form handling library.
  - **yup**: JavaScript schema builder for value parsing and validation.
  - **react-youtube-player**: Component to embed and control YouTube players.
  - **localStorage**: Client-side key-value storage.
- 

## 2. Overall Description

### 2.1 Product Perspective

FocusTube is a modern React SPA built with Vite and enhanced with state management, smooth forms, and minimalistic YouTube integration.

### 2.2 User Characteristics

Same as before: students, professionals, and focused content consumers.

## 2.3 Constraints

- No backend; everything is client-side.
- Relies on YouTube's API and user internet connection.
- Video playback controlled via react-youtube-player.
- Cannot block ads (limited by YouTube's TOS).

## 2.4 Assumptions

- Users will have playlist links.
  - Modern browsers required (ES6+, React support).
- 

## 3. Specific Requirements

### 3.1 Functional Requirements

#### 3.1.1 Playlist Management

- Paste playlist link/ID.
- Save and favorite playlists.
- Track last-watched playlists.
- Alert on missing input.
- Cache check before API request.
- Refresh cached data (once/24 hrs).

#### 3.1.2 Video Player

- Play videos using `react-youtube-player`.
- Hide branding, related videos, and show clean player UI.
- Player settings:
  - `modestbranding: 1`
  - `rel: 0`
  - `controls: 1` or custom as needed.
- Control playback (play, pause, next, previous).

#### 3.1.3 Form Handling

- Use `react-hook-form` for user input.
- Validate all form input using `yup`.

#### 3.1.4 State Management

- Use `easy-peasy` for managing global state:
  - Saved playlists
  - Favorites

- Last watched
- UI preferences (e.g. dark mode)

### 3.1.5 Data Storage

- Use `localStorage` for:
  - Saved playlists
  - Cache
  - Last watched video

### 3.1.6 Navigation

- Home (playlist input)
- Playlist viewer
- Saved & favorite playlists

---

## 4. Non-Functional Requirements

- **Performance:** Optimized via Vite; fast loading.
- **Usability:** Clean MUI + Tailwind UI.
- **Security:** No third-party data sync (yet).
- **Scalability:** Efficient cache & state management.
- **Reliability:** Error handling for all user actions and API calls.

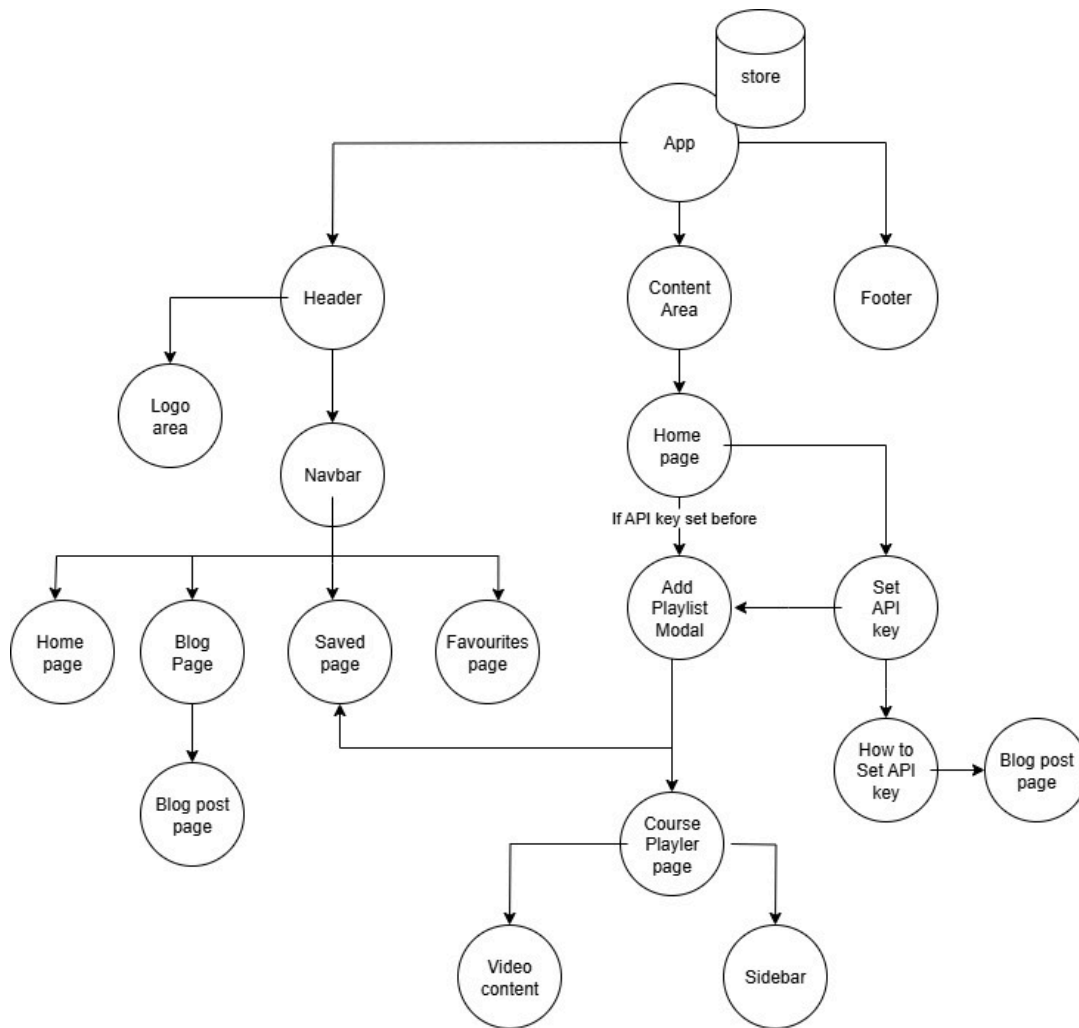
---

## 5. Technology Stack

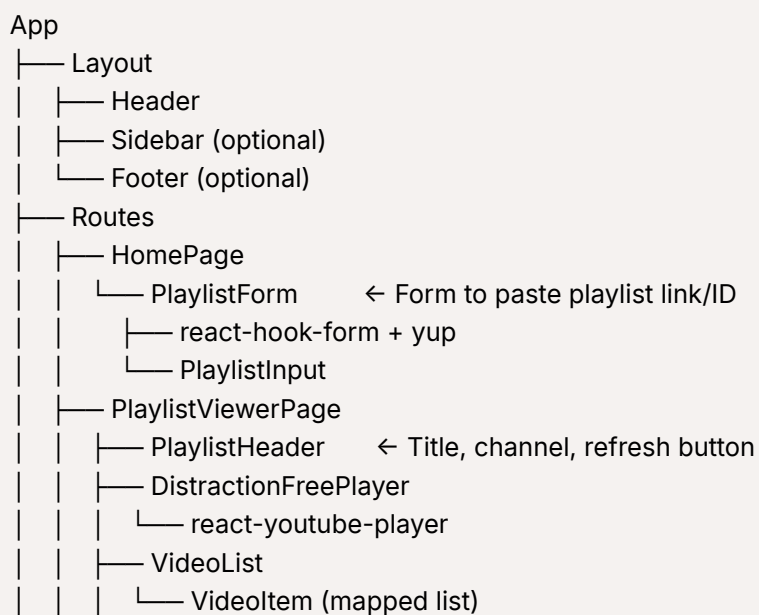
Component	Tool
Frontend	React + Vite
UI	MUI + Tailwind CSS
State Management	easy-peasy
Form Handling	react-hook-form
Validation	yup
Routing	react-router
Video Player	react-youtube-player
Storage	localStorage
API	YouTube API

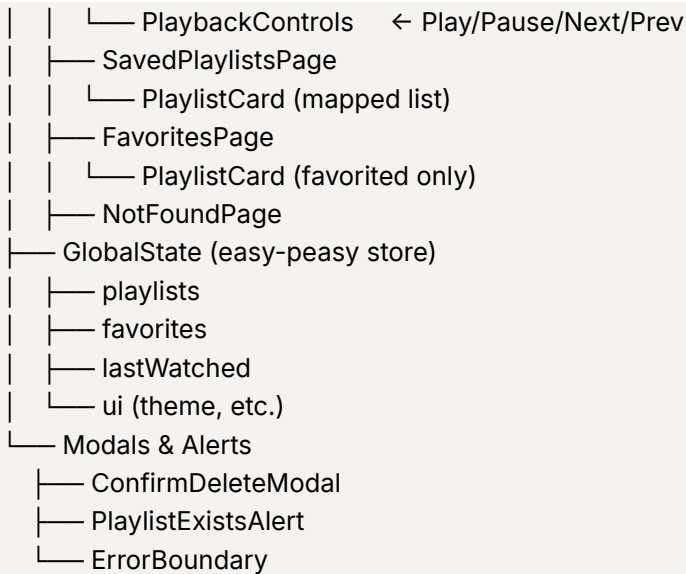
---

## 6 Architecture Diagram



## FocusTube Component Tree





## Notes:

- **DistractionFreePlayer** wraps **react-youtube-player** with minimal branding and clean layout.
- **State** is managed globally via **easy-peasy**, including:
  - Saved playlists
  - Favorites
  - Last watched video
- **Forms** are handled via **react-hook-form**, validated by **yup**.
- Pages are routed via **React Router**.
- Styling uses **MUI + Tailwind CSS**.

## 7. Future Enhancements

- Playlist progress tracking.
- Custom tags and categories.
- UI themes / dark mode toggle.
- Multi-device sync (using Firebase or other DB).
- Notification system (reminders before upcoming videos).

## 7. References

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

## 9. Conclusion

FocusTube continues to evolve into a focused, structured video learning tool. With improved tech stack integration and distraction-reduction features, it serves as a productivity-boosting companion for serious learners.

---