**Index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width,initial-scale=1" />

<title>MyTube - YouTube Clone</title>

<style>

:root{

--bg:#181818;

--card:#202020;

--text:#fff;

--muted:#aaa;

}

\*{ box-sizing:border-box; }

body{ margin:0; font-family: system-ui, Arial, sans-serif; background:var(--bg); color:var(--text); }

/\* Header \*/

.app-header{ position:sticky; top:0; z-index:20; display:flex; align-items:center; gap:10px; padding:10px; background:#202020; border-bottom:1px solid #2a2a2a; }

.logo{ color:#ff3b3b; font-weight:800; letter-spacing:.2px; }

.search-box{ display:flex; gap:8px; flex:1; }

.search-box input{ flex:1; padding:10px 12px; border:none; outline:none; border-radius:6px; background:#121212; color:var(--text); }

.search-box button{ border:none; padding:10px 14px; border-radius:6px; background:#303030; color:#fff; cursor:pointer; }

.header-actions{ display:flex; align-items:center; gap:10px; }

.ghost{ background:transparent; color:#ddd; border:1px solid #3a3a3a; padding:8px 10px; border-radius:8px; cursor:pointer; }

.small{ padding:6px 8px; font-size:12px; }

/\* User chip \*/

.user-chip{ display:flex; align-items:center; gap:8px; }

.user-chip img{ width:28px; height:28px; border-radius:50%; object-fit:cover; }

.hidden{ display:none !important; }

/\* History \*/

.history{ background:#111; border-bottom:1px solid #2a2a2a; }

.history-head{ display:flex; align-items:center; justify-content:space-between; padding:10px 12px; }

.history-list{ display:flex; flex-direction:column; gap:10px; padding:0 12px 12px; }

.h-item{ display:flex; gap:10px; align-items:flex-start; background:#171717; border-radius:10px; overflow:hidden; cursor:pointer; }

.h-thumb{ width:160px; aspect-ratio:16/9; object-fit:cover; background:#000; }

.h-info{ padding:10px; }

.h-title{ margin:0 0 6px; font-weight:700; font-size:15px; }

.h-meta{ margin:0; color:var(--muted); font-size:12px; }

/\* Feed \*/

main{ padding:12px; }

.video-grid{ display:flex; flex-direction:column; gap:14px; }

.video-card{ background:var(--card); cursor:pointer; border-radius:8px; overflow:hidden; }

.video-thumb{ width:100%; aspect-ratio:16/9; object-fit:cover; display:block; background:#000; }

.video-info{ padding:10px; }

.video-title{ margin:0 0 6px; font-size:16px; font-weight:700; line-height:1.35; }

.video-meta{ margin:0; font-size:13px; color:var(--muted); }

.empty-state{ text-align:center; color:var(--muted); margin-top:18vh; }

/\* Modal player \*/

.modal{ position:fixed; inset:0; background:rgba(0,0,0,.6); display:flex; align-items:center; justify-content:center; padding:14px; }

.modal-content{ width:min(900px,100%); background:#111; border-radius:12px; overflow:hidden; position:relative; }

.player-wrap{ position:relative; width:100%; aspect-ratio:16/9; background:#000; }

#ytPlayer{ position:absolute; inset:0; width:100%; height:100%; }

/\* Close button \*/

.close-btn{ position:absolute; top:8px; right:8px; border:none; background:#222; color:#fff; padding:6px 10px; border-radius:8px; cursor:pointer; z-index:30; }

.close-btn:hover{ background:#333; }

/\* Modal meta \*/

.modal-meta{ padding:12px 14px 16px; }

.modal-meta h3{ margin:0 0 6px; font-size:16px; }

.modal-meta p{ margin:0; color:var(--muted); font-size:13px; }

/\* Fullscreen fallback \*/

.modal-content.fullscreen-mode{

width:100vw !important;

height:100vh;

border-radius:0;

}

.modal-content.fullscreen-mode .player-wrap{ aspect-ratio: auto; height: calc(100vh - 80px); }

.modal-content.fullscreen-mode #ytPlayer{ height:100%; }

/\* Small screens \*/

@media (max-width:480px){

.logo{ font-size:18px; }

.h-thumb{ width:120px; }

}

</style>

<!-- Google Identity Services -->

<script src="https://accounts.google.com/gsi/client" async defer></script>

<script src="https://cdn.jsdelivr.net/npm/jwt-decode/build/jwt-decode.min.js"></script>

</head>

<body>

<!-- Header -->

<header class="app-header">

<div class="logo">MyTube</div>

<div class="search-box">

<input id="searchInput" type="text" placeholder="Search">

<button id="searchBtn" aria-label="Search">🔍</button>

</div>

<div class="header-actions">

<button id="historyToggle" class="ghost">History</button>

<div id="gSignIn"></div>

<div id="userChip" class="user-chip hidden">

<img id="userPic" alt="user">

<span id="userName"></span>

<button id="logoutBtn" class="ghost small">Logout</button>

</div>

</div>

</header>

<!-- History Drawer -->

<section id="historyDrawer" class="history hidden" aria-hidden="true">

<div class="history-head">

<h3>Watch History</h3>

<div>

<button id="clearHistory" class="ghost small">Clear</button>

<button id="closeHistory" class="ghost small">Close</button>

</div>

</div>

<div id="historyList" class="history-list"></div>

</section>

<!-- Feed -->

<main>

<div id="videoGrid" class="video-grid"></div>

<div id="emptyState" class="empty-state">Loading videos...</div>

</main>

<!-- Player Modal -->

<div id="playerModal" class="modal hidden" role="dialog" aria-modal="true">

<div id="modalContent" class="modal-content">

<button id="closeModal" class="close-btn" aria-label="Close">✖</button>

<button id="fullscreenBtn" class="close-btn" style="right:50px;" aria-label="Fullscreen">⛶</button>

<div id="playerWrap" class="player-wrap">

<iframe id="ytPlayer" src="" title="YouTube player"

frameborder="0"

allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share"

allowfullscreen></iframe>

</div>

<div class="modal-meta">

<h3 id="modalTitle"></h3>

<p id="modalChannel"></p>

</div>

</div>

</div>

<script>

const API\_KEY = "AIzaSyCVxFfRog-3ExopV6CpqJ9pxD9JT4wspRU";

const CLIENT\_ID = "350289349728-3n7o87c41qa9run2e1tsfo26kodsm76v.apps.googleusercontent.com";

const searchInput = document.getElementById("searchInput");

const searchBtn = document.getElementById("searchBtn");

const videoGrid = document.getElementById("videoGrid");

const emptyState = document.getElementById("emptyState");

const modal = document.getElementById("playerModal");

const modalContent = document.getElementById("modalContent");

const playerWrap = document.getElementById("playerWrap");

const closeModal = document.getElementById("closeModal");

const ytPlayer = document.getElementById("ytPlayer");

const modalTitle = document.getElementById("modalTitle");

const modalChannel = document.getElementById("modalChannel");

const fullscreenBtn = document.getElementById("fullscreenBtn");

const historyToggle = document.getElementById("historyToggle");

const historyDrawer = document.getElementById("historyDrawer");

const historyList = document.getElementById("historyList");

const closeHistory = document.getElementById("closeHistory");

const clearHistory = document.getElementById("clearHistory");

const gSignIn = document.getElementById("gSignIn");

const userChip = document.getElementById("userChip");

const userPic = document.getElementById("userPic");

const userName = document.getElementById("userName");

const logoutBtn= document.getElementById("logoutBtn");

let currentUser = getSavedUser();

let isPlayerOpen = false;

let lastRequestedFullscreen = false;

document.addEventListener("DOMContentLoaded", () => {

initGoogleSignIn();

updateAuthUI();

loadPopularVideos();

renderHistory();

attachOrientationHandlers();

searchInput.focus();

});

searchBtn.addEventListener("click", searchVideos);

searchInput.addEventListener("keydown", e => { if (e.key === "Enter") searchVideos(); });

closeModal.addEventListener("click", closeVideoAndExitFullscreen);

modal.addEventListener("click", e => { if (e.target === modal) closeVideoAndExitFullscreen(); });

historyToggle.addEventListener("click", () => { historyDrawer.classList.toggle("hidden"); renderHistory(); });

closeHistory.addEventListener("click", () => historyDrawer.classList.add("hidden"));

clearHistory.addEventListener("click", () => { saveHistory([]); renderHistory(); });

logoutBtn.addEventListener("click", () => {

localStorage.removeItem("mytube\_user");

currentUser = null;

updateAuthUI();

});

fullscreenBtn.addEventListener("click", () => {

if (document.fullscreenElement) exitFullscreenIfAny();

else requestFullscreenOnPlayer();

});

async function loadPopularVideos() {

showEmpty("Loading trending videos...");

try {

const url = `https://www.googleapis.com/youtube/v3/videos?part=snippet&chart=mostPopular&maxResults=20&regionCode=IN&key=${API\_KEY}`;

const res = await fetch(url);

const data = await res.json();

if (!data.items) return showEmpty("No videos found.");

const vids = data.items.map(v => ({

id: v.id,

title: v.snippet.title,

channel: v.snippet.channelTitle,

thumb: bestThumb(v.snippet.thumbnails),

date: new Date(v.snippet.publishedAt).toLocaleDateString()

}));

renderVideos(vids);

} catch (err) { console.error(err); showEmpty("Error loading videos."); }

}

async function searchVideos() {

const q = searchInput.value.trim();

if (!q) return;

showEmpty("Searching videos...");

try {

const url = `https://www.googleapis.com/youtube/v3/search?part=snippet&type=video&maxResults=20&q=${encodeURIComponent(q)}&key=${API\_KEY}`;

const res = await fetch(url);

const data = await res.json();

if (!data.items) return showEmpty("No results found.");

const vids = data.items.map(v => ({

id: v.id.videoId,

title: v.snippet.title,

channel: v.snippet.channelTitle,

thumb: bestThumb(v.snippet.thumbnails),

date: new Date(v.snippet.publishedAt).toLocaleDateString()

}));

renderVideos(vids);

} catch (err) { console.error(err); showEmpty("Error loading search results."); }

}

function renderVideos(videos) {

emptyState.style.display = "none";

videoGrid.innerHTML = "";

videos.forEach(v => {

const card = document.createElement("div");

card.className = "video-card";

card.innerHTML = `

<img class="video-thumb" src="${v.thumb}" alt="">

<div class="video-info">

<h3 class="video-title">${escapeHTML(v.title)}</h3>

<p class="video-meta">${escapeHTML(v.channel)} • ${v.date}</p>

</div>`;

card.addEventListener("click", () => { openPlayer(v.id, v.title, v.channel, v.thumb); addToHistory(v); });

videoGrid.appendChild(card);

});

}

function openPlayer(videoId, title, channel, thumb) {

if (!videoId) return alert("Video ID not found");

isPlayerOpen = true;

modal.classList.remove("hidden");

modalTitle.textContent = title;

modalChannel.textContent = channel;

ytPlayer.src = `https://www.youtube.com/embed/${videoId}?autoplay=1&rel=0`;

setTimeout(() => tryEnterFullscreenIfLandscape(), 120);

}

function closeVideoAndExitFullscreen(){

if (document.fullscreenElement) document.exitFullscreen().catch(()=>{});

modalContent.classList.remove("fullscreen-mode");

stopVideo();

modal.classList.add("hidden");

isPlayerOpen = false;

}

function stopVideo(){ ytPlayer.src = ""; }

async function requestFullscreenOnPlayer(){

try {

if (playerWrap.requestFullscreen) await playerWrap.requestFullscreen();

else if (playerWrap.webkitRequestFullscreen) await playerWrap.webkitRequestFullscreen();

else modalContent.classList.add("fullscreen-mode"