

Google Drive file Manager

Overview

The Google Drive File Manager is a Python application that enables users to search for and view files stored in their Google Drive. It provides an interactive chatbot interface for easy file discovery and viewing capabilities.

Technical Requirements

Dependencies

- google-oauth2-client
- google-auth-oauthlib
- google-auth-httpplib2
- google-api-python-client

System Requirements

- **Python:** 3.6+
 - **Operating System Support:**
 - Windows
 - macOS
 - Linux (with xdg-open installed)
 - **Additional Requirements:**
 - Google Drive API access
 - Default PDF viewer installed on the system
-

Setup Instructions

1. Google Cloud Project Setup

1. Go to **Google Cloud Console**.
2. Create a new project or select an existing one.
3. Enable the **Google Drive API**.
4. Create **OAuth 2.0 credentials**.

5. Download the credentials and save them as credentials.json in the project directory.

2. Installation

1. Clone the repository.
2. Install required packages:

Copy code: `pip install google-oauth2-client google-auth-oauthlib google-auth-httplib2 google-api-python-client`

3. Place the credentials.json file in the project root directory.
-

Architecture

Class Structure

- **Google Drive File Manager**
Main class handling Google Drive operations and PDF management.
 - **Key Methods:**
 - `authenticate()`: Handles Google Drive authentication
 - `search_pdf(query)`: Searches for PDF files in Google Drive
 - `view_pdf(file_id)`: Downloads and opens PDF files
 - `_open_pdf(file_path)`: Opens PDF with system default viewer
 - **PDF Chatbot**
Interactive interface for user interactions.
 - **Key Methods:**
 - `start()`: Initiates the chatbot interface
 - `__init__()`: Initializes the PDF manager and authenticates
-

Authentication Flow

1. **First-time Authentication:**
 - Checks for an existing token (token.pickle).
 - If not found, launches browser-based OAuth flow.
 - Saves credentials for future use.
2. **Subsequent Authentication:**

- Uses stored token.
 - Automatically refreshes if expired.
 - Falls back to new authentication if refresh fails.
-

Usage Guide

Starting the Application

Run the main script:

Copy code : `python main.py`

Basic Operations

1. **Searching for PDFs:**
 - Enter search terms when prompted.
 - System displays matching PDFs from Google Drive in a numbered list.
 2. **Viewing PDFs:**
 - Select PDF by entering its number.
 - File downloads automatically and opens in the system default PDF viewer.
 3. **Navigation:**
 - Type 'back' to return to search.
 - Type 'quit' to exit application.
-

Security Considerations

1. **Credential Management:**
 - Keep `credentials.json` secure.
 - Avoid committing tokens to version control.
 - Use appropriate file permissions.
 2. **Token Storage:**
 - `token.pickle` contains sensitive data and is stored locally.
 - Implement appropriate access controls.
-

Error Handling

The application handles several types of errors, including:

- Authentication failures
 - Search errors
 - Download issues
 - File opening problems
 - Invalid user inputs
-

Temporary File Management

- Temporary files are created for PDF viewing.
 - Files are created with unique names.
 - Current implementation retains temporary files; consider implementing a cleanup routine.
-

Troubleshooting

Common Issues and Solutions

- 1. Authentication Failures:**
 - Verify credentials.json is present.
 - Check API enablement.
 - Ensure correct OAuth scopes.
 - 2. PDF Viewing Issues:**
 - Confirm default PDF viewer installation.
 - Check file permissions.
 - Verify temporary directory access.
 - 3. Search Problems:**
 - Check internet connectivity.
 - Verify Google Drive access.
 - Confirm search query format.
-

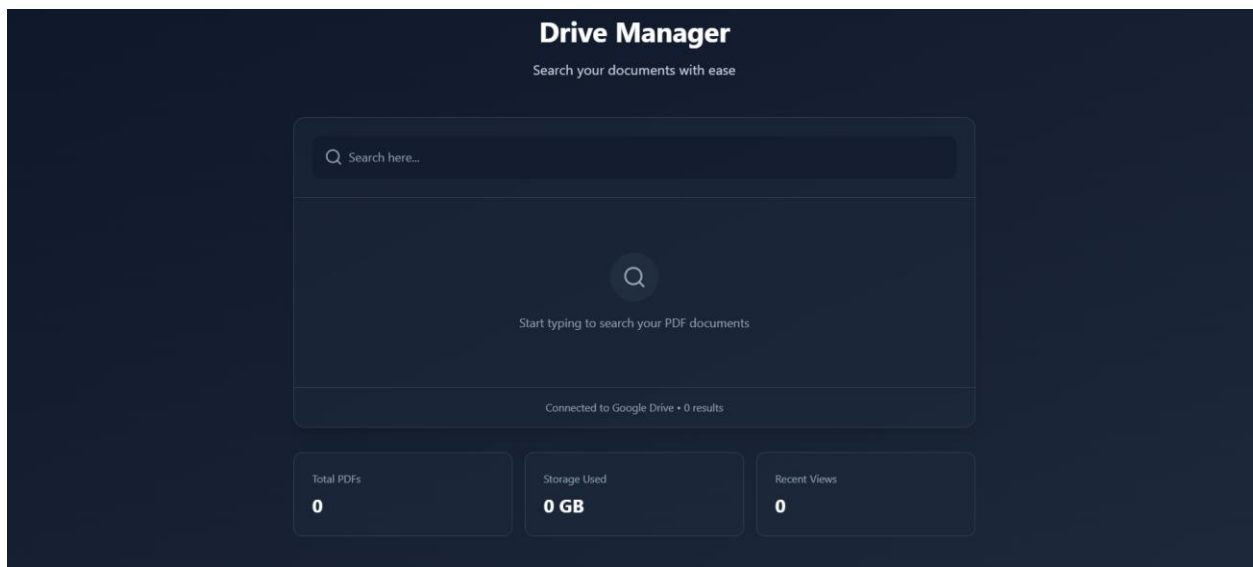
API Reference

Google Drive API Endpoints Used

- `files().list`: Search for files.
 - `files().get_media`: Download files.
 - **Scope**: <https://www.googleapis.com/auth/drive.readonly>
-

Front-End Design

The Google Drive PDF Manager's front end is built with React and Vite for speed and efficiency. The app is fully responsive, providing a clean, user-friendly interface with intuitive navigation. \



Support

For issues and support:

1. Check the troubleshooting guide.
2. Review Google Drive API documentation.
3. Verify system requirements.
4. Check for updates to dependencies