Take-Home Test

Objective

Develop a small application that integrates with Google Drive. The application should be able to perform the following tasks:

1. Authenticate the user using OAuth 2.0.
2. List files in the user's Google Drive.
3. Upload a file to the user's Google Drive.
4. Download a file from the user's Google Drive.
5. Delete a file from the user's Google Drive.

Requirements

- Use a modern programming language (e.g., Python, Java, JavaScript).
- Follow best practices for software design and development.
- Write unit and integration tests.
- Provide clear documentation.

Task Breakdown

1. **Authentication**: Implement OAuth 2.0 authentication to allow users to log in with their Google account and authorize the application to access their Google Drive.
2. **List Files**: Develop functionality to list all files in the user's Google Drive. Display file names, types, and last modified dates.
3. **Upload File**: Implement a feature to upload a file to the user's Google Drive. Allow the user to select a file from their local system and upload it to a specified folder in Google Drive.
4. **Download File**: Implement a feature to download a file from the user's Google Drive. Allow the user to select a file from the list of files and download it to their local system.
5. **Delete File**: Implement a feature to delete a file from the user's Google Drive. Allow the user to select a file from the list of files and delete it.

Deliverables

- Source code hosted on a public repository (e.g., GitHub).
- Documentation and testing instructions.
- A short video (5-10 minutes) demonstrating the application and explaining your approach.

Submission Guidelines

Code Quality

- Ensure your code is clean, readable, and well-documented.
- Use appropriate design patterns where applicable.
- Adhere to SOLID principles.

Testing

- Write unit tests for individual components.
- Write integration tests for end-to-end functionality.
- Include instructions on how to run the tests.

Documentation

- Provide a README file with:
  - An overview of the application.
  - Instructions on setting up the development environment.
  - Steps to run the application.
  - Any assumptions or design decisions made.
- Comment your code where necessary to explain complex logic.