**Documentation and Instructions**

### Prerequisites

- Go 1.16 or later

- ScyllaDB cluster

**Installation**

1. Navigate to the project directory: cd todo-api
2. Install the dependencies: go mod download
3. Run the application: go run main.go

The server will start running on `http://localhost:8080`.

**Documentation**

Creating aTODO item: The createTodo function in main.go demonstrates how to create a new TODO item and save it to a file.

Retrieving a TODO item: The getTodo function in main.go shows how to retrieve a specific TODO item from a file.

Updating a TODO item: The updateTodo function in main.go illustrates how to update an existing TODO item and save the changes to the file.

Listing TODO items: The listTodos function in main.go demonstrates how to retrieve a list of TODO items from the files, with the ability to filter by status.

Deleting a TODO item: The deleteTodo function in main.go shows how to delete a specific TODO item by removing the corresponding file. The todo.go file contains the implementation of the TODO-related functions, such as saving, loading, and managing the TODO items.

**Design Decisions**

1. Separation of Parts: The code is divided into different sections, like the API, the database, and the main logic. This makes it easier to understand and work on each part separately.

2. Dependency Sharing: The main part of the code (the service) uses the database part (the repository) without needing to know how the database works. This makes it easier to change the database in the future.

3. Pagination: The list of TODO items can be shown in smaller parts, instead of all at once. This is useful when there are a lot of items, as it prevents the server from having to send everything at once.

4. Filtering: You can filter the TODO items by their status (pending or completed). This allows users to easily find the items they're interested in.

5. Configuration File: The application uses a configuration file to store settings like the database connection details. This makes it easy to change the settings without modifying the code.

6. Error Handling: The code includes basic error handling, which means it will provide helpful messages if something goes wrong when using the API.

7. Simplicity: The code is designed to be simple and straightforward, focusing on the main features of the TODO API. This makes the code easier to understand and maintain.