

メモ帳アプリ 詳細設計書

1. 概要

1.1 アプリケーションの目的

シンプルで効率的にメモを作成、保存、閲覧、削除できるローカルメモ帳アプリケーションを開発します。SQLite データベースを使用し、メモはアプリ起動時にロードされ、データベースに保存されます。

1.2 使用技術

- プログラミング言語: Python 3.12
- データベース: SQLite
- 開発環境: Spyder

2. システム構成

2.1 システムアーキテクチャ

本アプリケーションは、シンプルなコマンドラインインターフェース(CLI)で構築され、SQLite データベースをバックエンドとして使用します。

- メモのデータは SQLite のメモテーブルに保存される
- メモは作成後、自動的に保存され、アプリ終了後も保持される

2.2 システムフロー

1. **アプリ起動:**
 - アプリケーションを起動すると、memo.db データベースに接続し、保存されたメモを一覧で表示する。
2. **メモ作成:**
 - ユーザーが新規メモを作成し、タイトルと内容を入力して保存ボタンを押すと、メモがデータベースに保存される。
3. **メモ表示:**
 - アプリ起動後、保存されたメモをタイトルのリストとして表示し、ユーザーは選択してメモ内容を閲覧できる。
4. **メモ削除:**
 - ユーザーが不要なメモを選択し、削除ボタンを押すと、選択されたメモがデータベースから削除される。

2.3 データベース構成

2.3.1 メモテーブル (memo)

カラム名	データ型	説明
id	INTEGER	メモ ID (主キー、自動インクリメント)
title	TEXT	メモのタイトル
content	TEXT	メモの内容
created_at	DATETIME	作成日時
updated_at	DATETIME	更新日時

2.4 データベース操作

1. **メモ保存:**
 - メモ作成時に、ユーザーが入力したタイトルと内容を memo テーブルに保存する。
 - created_at と updated_at のカラムは自動的に保存時の日時で更新。
2. **メモ表示:**
 - アプリ起動時、memo テーブルからすべてのメモをタイトルで並べて表示。
 - メモの選択後、内容を表示する。
3. **メモ削除:**
 - ユーザーが選択したメモを memo テーブルから削除。

3. 画面設計

3.1 メイン画面

- メモ一覧表示:
 - 保存されているメモのタイトルをリストで表示。
 - タイトルをクリックすることで、メモ内容を表示する。
- メモ作成ボタン:
 - 新しいメモを作成するためのボタン。クリックで新規メモ作成画面に移。

3.2 メモ作成画面

- タイトル入力欄: ユーザーがメモのタイトルを入力。
- 内容入力欄: ユーザーがメモの内容を入力。
- 保存ボタン: 入力されたデータを保存するボタン。

3.3 メモ詳細画面

- タイトル表示: メモのタイトルを表示。
- 内容表示: メモの内容を表示。
- 削除ボタン: メモを削除するボタン。

4. 主要機能詳細

4.1 メモ作成機能

- ユーザーがメモのタイトルと内容を入力し、「保存」ボタンを押すことで、メモがデータベースに保存されます。
- 保存されたメモは自動的に一覧画面に反映されます。

4.2 メモ一覧表示機能

- アプリ起動時に、保存されているすべてのメモがタイトル順に表示されます。
- ユーザーがタイトルを選択すると、そのメモの詳細が表示されます。

4.3 メモ削除機能

- ユーザーがメモを選択し、「削除」ボタンを押すことで、選択されたメモがデータベースから削除されます。

5. エラーハンドリング

- メモ作成時にタイトルや内容が空の場合、エラーメッセージを表示。
- メモ削除時に存在しない ID を指定した場合、エラーメッセージを表示。

6. 今後の改良点

- より多機能な GUI を追加することで、ユーザーの操作性を向上させる。
- メモ検索機能を実装し、特定のタイトルや内容でメモを検索できるようにする。
- メモにタグ付け機能を追加して、カテゴリ別に整理できるようにする。