**To-Do List Application Documentation**

**Overview**

This Flutter application provides a user-friendly interface for managing tasks. Users can add, edit, mark as completed, and delete tasks. Tasks can include a name, description, and optional due date and time.

**Features**

* **Task Management**: Users can add, edit, mark as completed, and delete tasks.
* **Due Date and Time**: Tasks can have an optional due date and time.
* **Persistent Storage**: Tasks are stored locally using **SharedPreferences**, allowing data persistence across app launches.

**Usage**

**Dependencies**

* Dart SDK
* Flutter

**Installation**

1. Clone the repository to your local machine.
2. Ensure that you have Flutter installed on your system.
3. Run **flutter pub get** to install the required dependencies.
4. Connect a device or start an emulator.
5. Run **flutter run** to launch the application.

**Functionality**

* **Adding a Task**: Tap the floating action button (add icon) to add a new task. Enter the task name, description, and optionally select a due date and time.
* **Editing a Task**: Long press on a task to edit its details. Update the task name, description, or due date and time as needed.
* **Completing a Task**: Tap the checkbox icon next to a task to mark it as completed.
* **Deleting a Task**: Long press on a task and confirm to delete it.
* **Due Date and Time**: When adding or editing a task, tap on "Set Due Date and Time" to specify a due date and time.

**Code Structure**

* **Task**: A class representing a task with properties such as name, description, completed status, and due date and time.
* **ToDoListPage**: The main screen of the application, displaying the list of tasks and providing options for task management.
* **SharedPreferencesHelper**: A helper class for managing the storage and retrieval of tasks using **SharedPreferences**.
* **main**: Entry point of the application, initializing the **ToDoListPage**.

**Future Enhancements**

* Implementing user authentication and cloud synchronization for multi-device usage.
* Adding notifications for approaching or overdue tasks.
* Improving the UI with animations and themes.