Capgemini Todo Project

Created by Alejandro Rodriguez

Approach and Design

- Objective: Command-line Todo list application.
- Modular architecture: Controller, Service, DAO, Models.
- SQLite database with `Project` and `Todo_Item` tables.
- Features: CRUD operations for projects and todos, cascading deletes.

Key Files and Folders

- `main.py` : Entry point.
- `src/Controller/` : Handles user commands.
- `src/Service/` : Business logic.
- `src/DAO/` : Database operations.
- `src/Models/` : Data models.
- `src/Resources/TodoListSetup.sql` : Schema and seed data.
- `scripts/` : Setup scripts.

Process to Run, Test, and Verify

- Run the application: `python3 main.py`.
- **DAO Tests**:
- `test_project_dao.py`: Tests for `ProjectDAO`.
- `test_todoitem_dao.py`: Tests for `TodoItemDAO`.
- **Service Tests**:
- `test_project_service.py`: Tests for `ProjectService`.
- `test_todoitem_service.py`: Tests for `TodoItemService`.
- Known bug with the delete cascade operation for Project deletion

Test Data and Seed Data

- Seed data in `TodoListSetup.sql`:
- Projects: `General`, `Work`, `Personal`.
- Todos: Example tasks for each project.
- Add new projects and todos to test functionality.