adding-a-new-task-feature

Guide: Adding a New Task Feature

This walkthrough explains how to add a new feature (e.g., "Task") to the Task Manager application using Clean Architecture and DDD principles.

---

1. Plan the Feature

- Define the business requirements (e.g., what is a Task, what actions can users perform?).
- Identify domain concepts: aggregates, entities, value objects, domain events.

2. Domain Layer

- Create a `Task` aggregate (or entity) in `TaskManager.Domain/Task/`.
- Define value objects (e.g., `TaskId`, `TaskPriority`).
- Implement invariants (e.g., title required, due date in the future).
- Example:

  ```
  public sealed record TaskId(Guid Value);
  public sealed class Task
  {
      // ...properties, invariants, business methods...
      private Task(...) { /* private ctor */ }
      public static Task Create(...) { /* factory method */ }
  }
  ```

3. Application Layer

- Add use cases (commands/queries) in `TaskManager.Application/Task/`:
    - `CreateTaskCommand`, `GetTaskByIdQuery`, `UpdateTaskCommand`, etc.
- Implement handlers for each use case.
- Define ports (interfaces) for repositories/services.

4. Infrastructure Layer

- Implement repository adapters in `TaskManager.Infrastructure/Task/`.

- Integrate with persistence (e.g., EF Core DbContext).
- Add external integrations if needed (e.g., notifications).

5. Api Layer

- Expose endpoints in `TaskManager.Api/Task/` using Minimal API:
    - `POST /tasks`, `GET /tasks/{id}`, etc.
- Map requests to application commands/queries.
- Register services and repositories in DI.
- Handle errors with ProblemDetails.

6. Testing

- Add unit tests for domain and application logic in `tests/TaskManager.UnitTests/Task/`.
- Add integration tests for infrastructure and API in `tests/TaskManager.IntegrationTests/Task/`.
- Use xUnit, FakeItEasy, and Testcontainers as appropriate.

7. Documentation & Commit

- Update architecture docs and API docs as needed.
- Use Conventional Commits for all changes (e.g., `feat(task): add create task use case`).

See also: Sample Solution Architecture, Glossary, Tasks API