

# onboarding

- [Onboarding Guide](#)
  - [1. Repository Structure](#)
  - [2. Key Concepts](#)
  - [3. Getting Started](#)
  - [4. Coding Standards](#)
  - [5. Adding Features](#)
  - [6. Testing](#)
  - [7. Documentation](#)
  - [8. Getting Help](#)

## Onboarding Guide

Welcome to the Task Manager application! This guide will help new contributors get started quickly, understand the repo structure, and follow best practices.

---

### 1. Repository Structure

```
src/  
  TaskManager.Domain/  
  TaskManager.Application/  
  TaskManager.Infrastructure/  
  TaskManager.Api/  
tests/  
  TaskManager.UnitTests/  
  TaskManager.IntegrationTests/  
docs/  
  design/  
  api/  
  guides/  
  adr/
```

### 2. Key Concepts

- Clean Architecture: Separation of concerns by layer.
- DDD: Aggregates, entities, value objects, domain events.
- CQRS: Separate commands (write) and queries (read).
- TDD: Write tests first, then code.

### 3. Getting Started

1. Clone the repository.
2. Review the [Sample Solution Architecture](#) and [Glossary](#).
3. Install .NET 9 SDK and required tools.
4. Run tests: `dotnet test`
5. Explore the codebase by feature folders (e.g., Task/).

## 4. Coding Standards

- Follow [Copilot Instructions](#).
- Use feature-oriented folders.
- Write descriptive Conventional Commits (e.g., feat(task): add create task use case).
- Prefer async/await, DI, and value objects.

## 5. Adding Features

- See [Feature Walkthrough](#).
- Start with TDD: write failing tests, then implement code.
- Update docs and API specs as needed.

## 6. Testing

- Unit tests for domain/application logic.
- Integration tests for infrastructure/api.
- Use xUnit, FakeItEasy, Testcontainers.

## 7. Documentation

- Architecture: docs/design/architecture.md
- API: docs/api/tasks.md
- ADRs: docs/adr/
- Diagrams: docs/design/diagrams/

## 8. Getting Help

- Review documentation in docs/.
- Ask questions in project discussions or issues.

---

*Welcome aboard! Contribute, learn, and help improve the Task Manager application.*