

Testing the Application Code



Gill Cleeren

CTO Xpirit Belgium

@gillcleeren | xpirit.com/gill

Overview

Understanding the different test types

Creating unit tests

Writing integration tests



**Did we create a
testable code base
using our architecture?**



Understanding the Different Test Types



Different Types of Tests

Unit test

Integration test

Functional test



**A unit test is code that will,
in an automated way,
invoke code to be tested.**

**It will check an assumption
about the behavior of the code
under test.**



Unit Tests

Test public API

Run in isolation

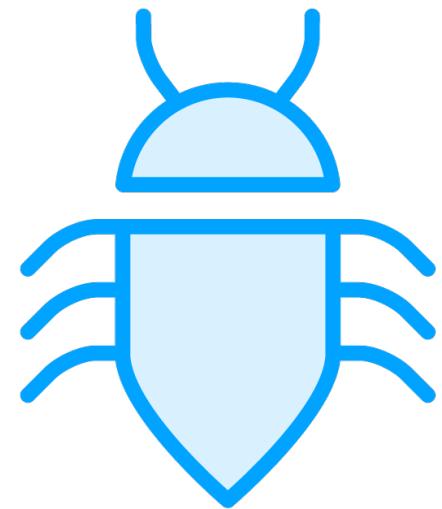
Consistent results

Fast

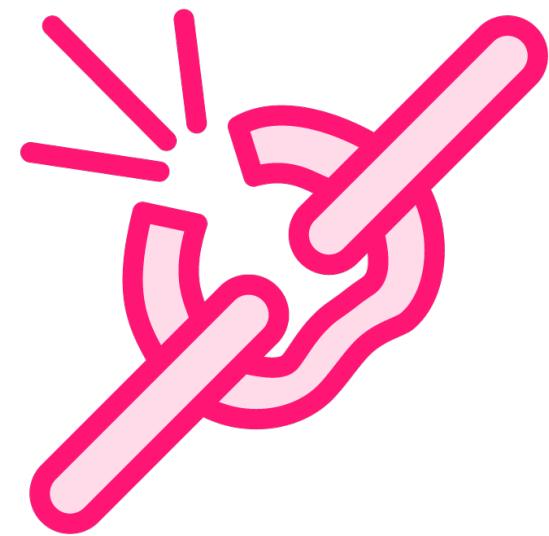
Often automated



Why Do We Need Unit Tests?



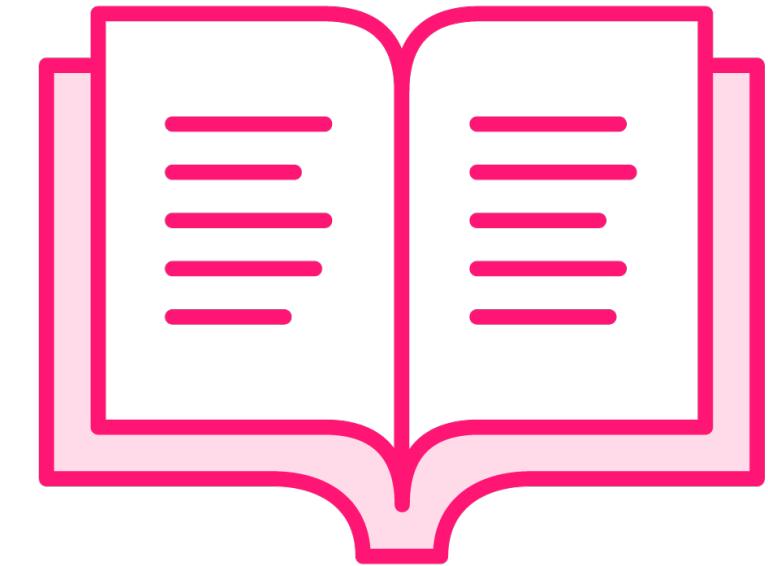
Find bugs



**Change without
fear of breaking
something**



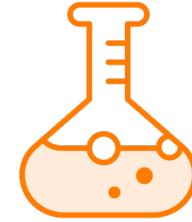
Improve quality



**Documentation
of the code**



Integration Tests



Test infrastructure code



Interaction between different layers

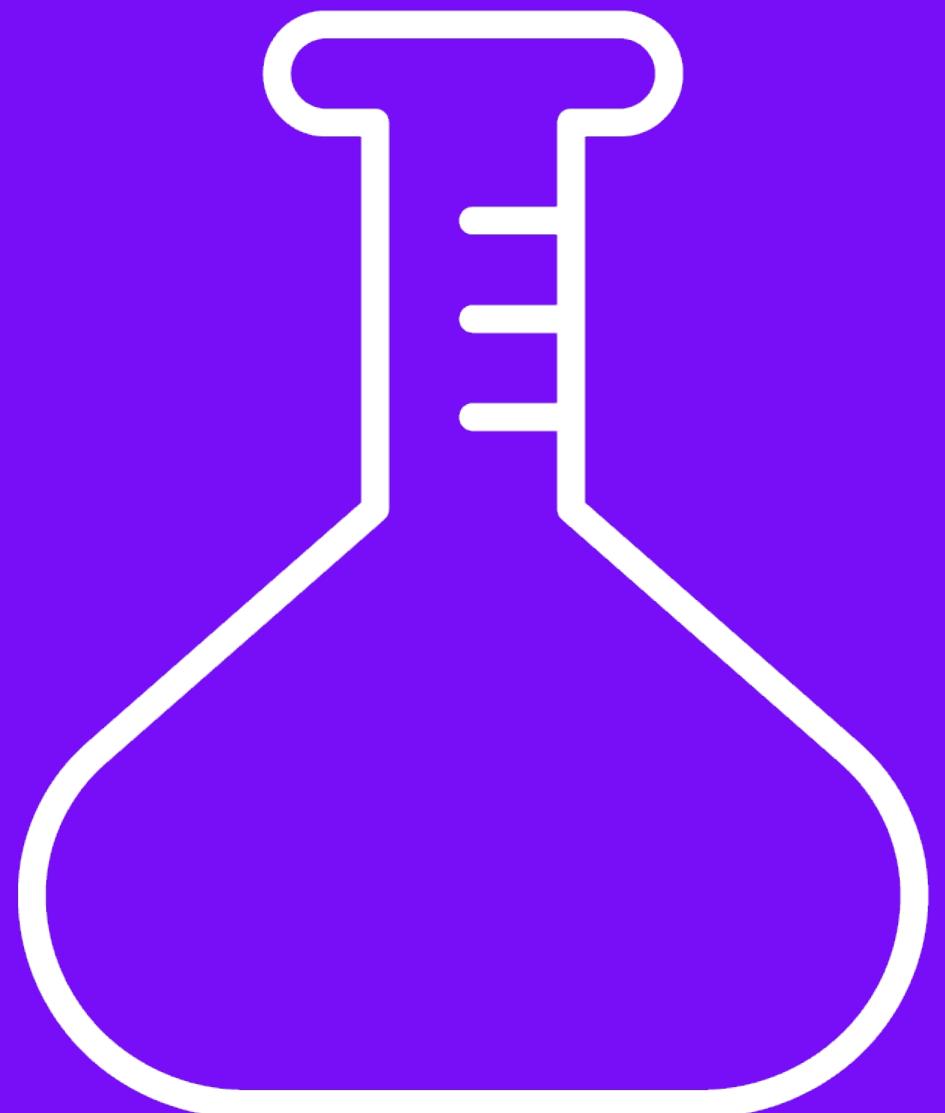


More work to set up



Often linked with database





Functional tests

Test to see if the system behaves as expected

Written from the perspective of the user

Often involves UI testing



Creating Unit Tests



Setting Up Unit Tests

Test Project
xUnit

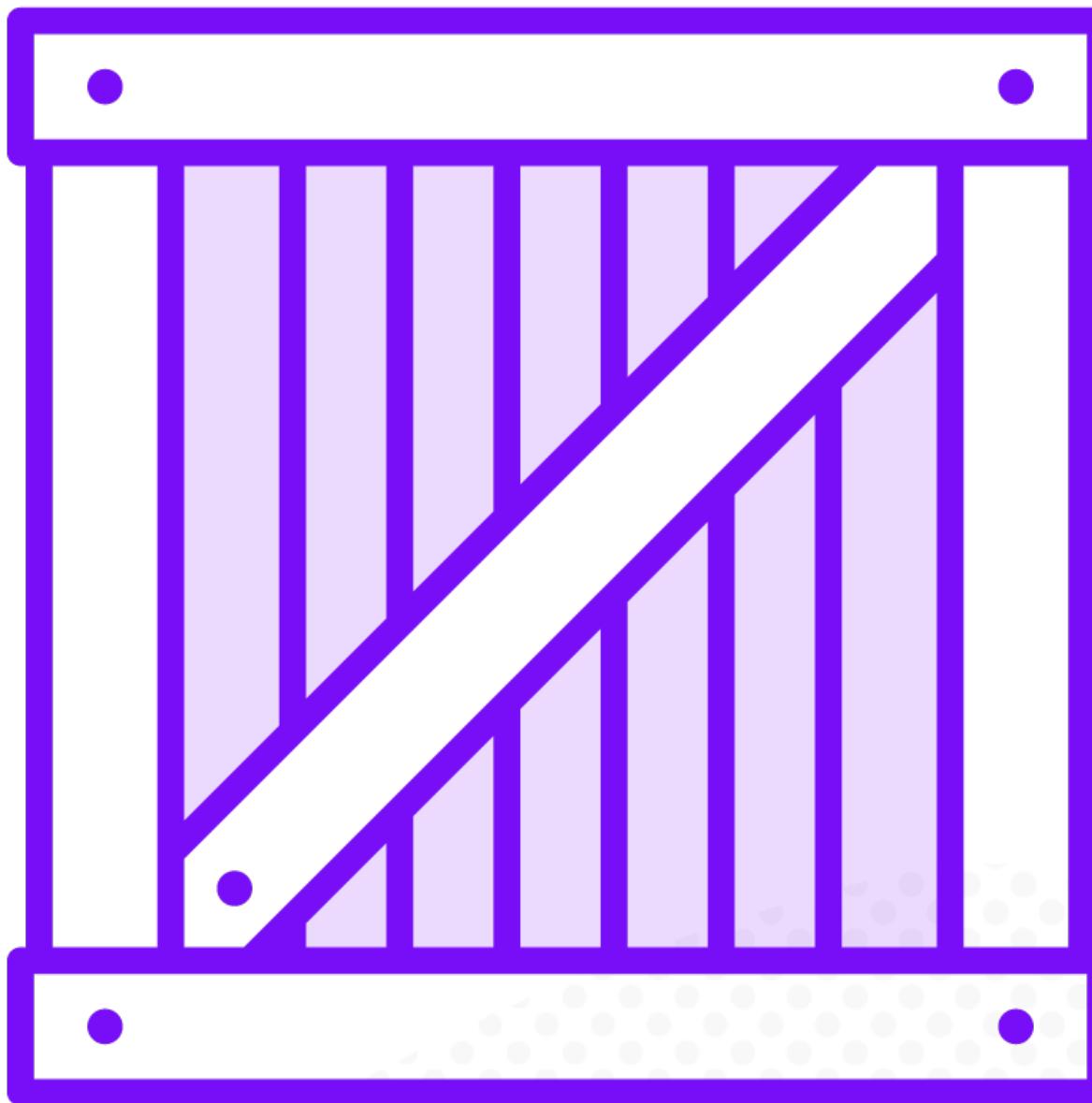
Mock data
Manual or using
framework

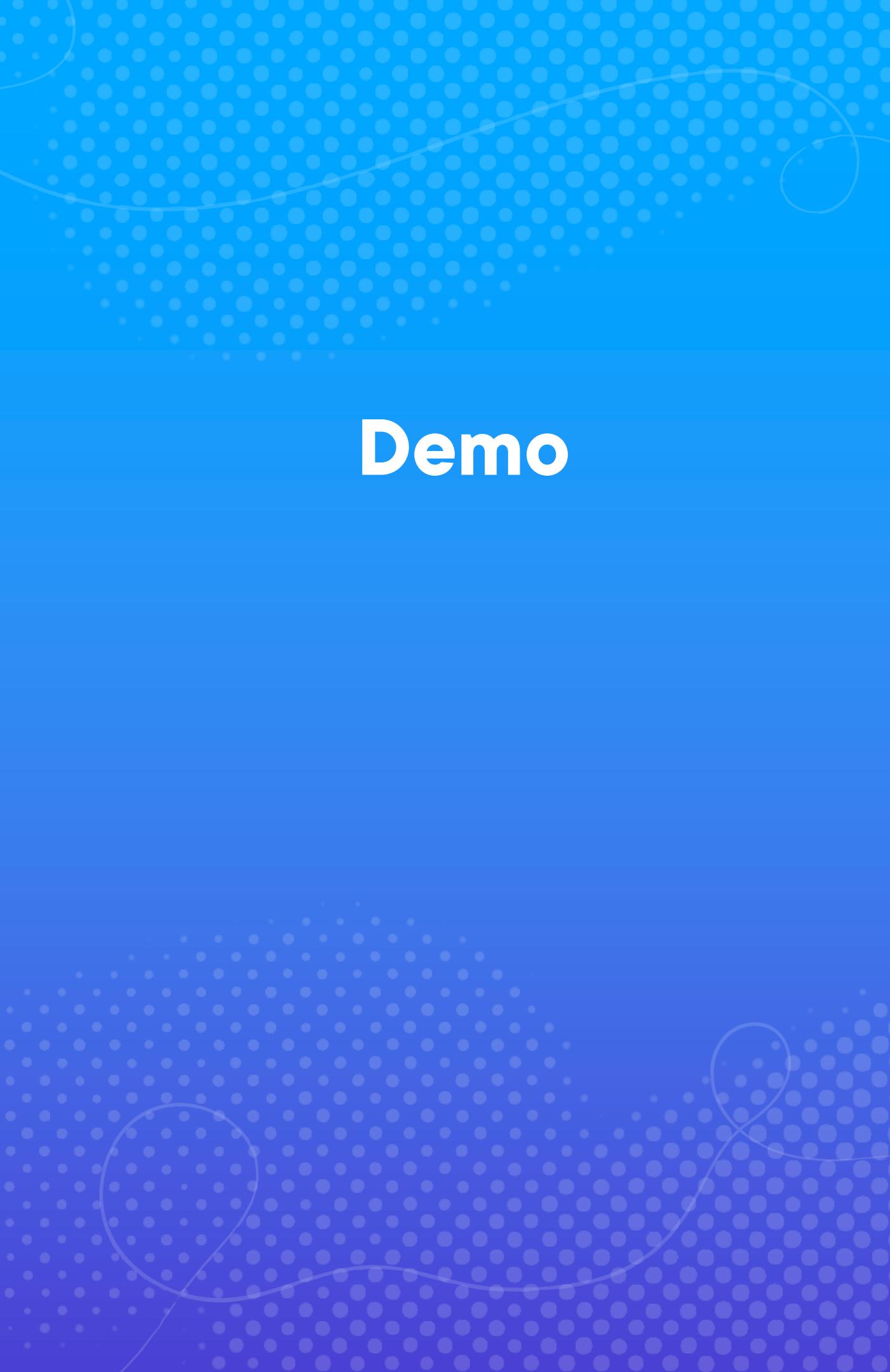
**Dependency
injection**



Used packages

Moq
Shouldly
xUnit





Demo

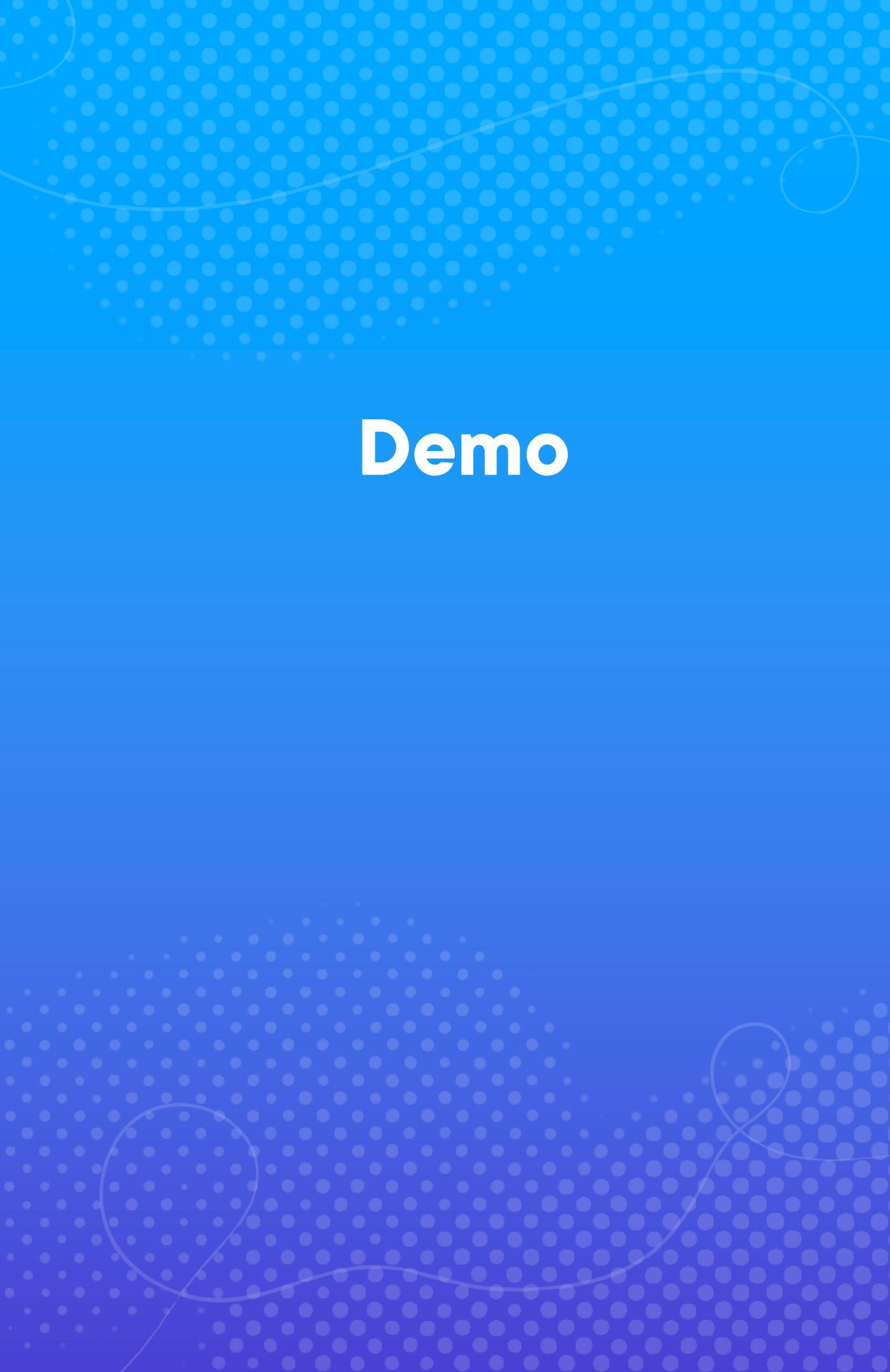
Creating unit tests for Core code

Adding mocks



Writing Integration Tests

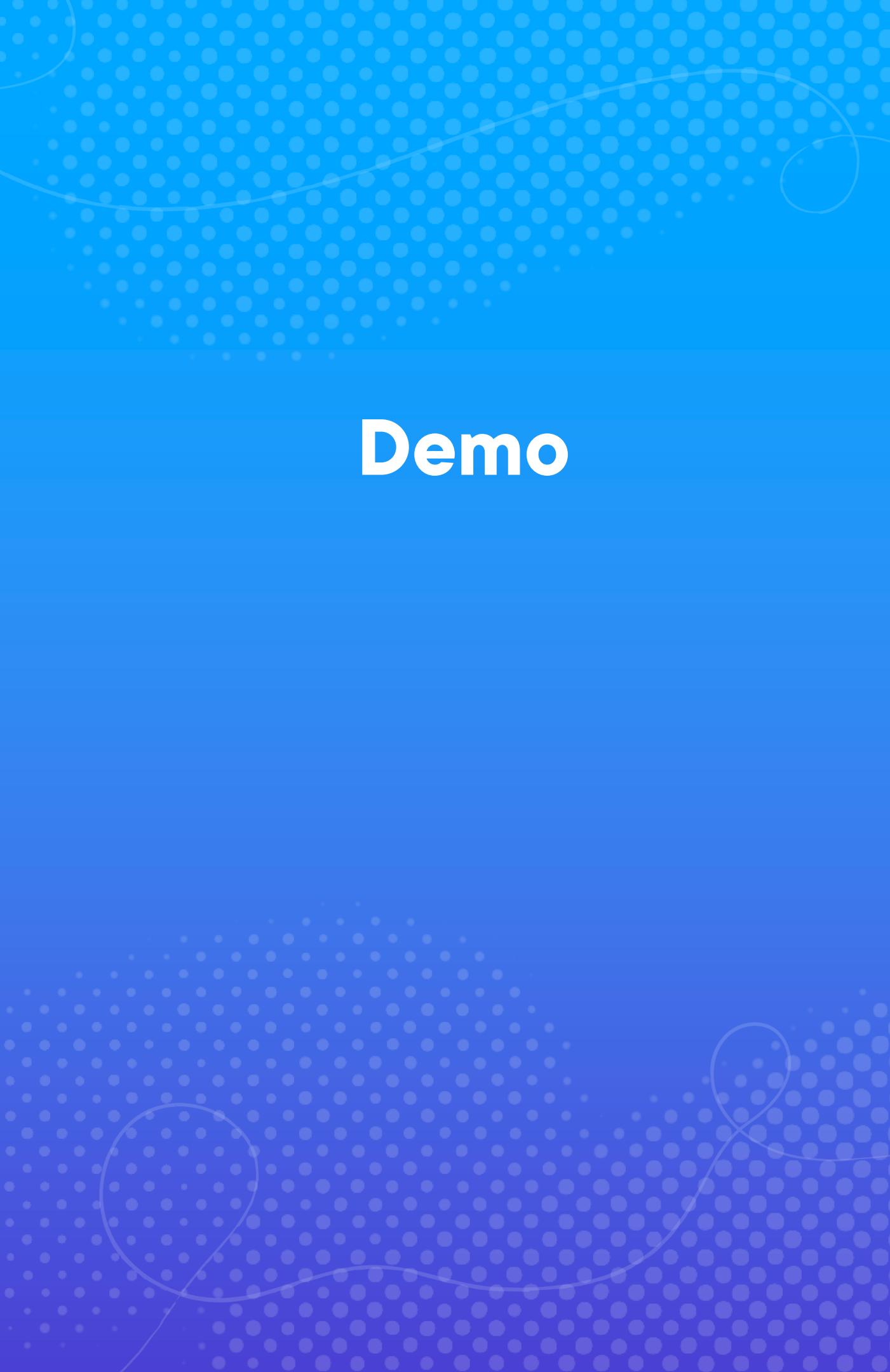




Demo

Testing the Infrastructure code





Demo

Testing the controller code



Summary

Application architecture supports testing on different levels

Different types of tests are required



Up Next:

Adding a Blazor UI

