Лабораторная работа №3

В ходе выполнения лабораторной работы вам будет необходимо выполнить следующие пункты:

1. Добавить в решение проект модульного тестирования (Выбрать шаблон проекта NUnit)
2. Затем в созданный проект добавить ссылки на существующие проекты.
3. Реализовать два класса для тестирования:
   1. TestsProductController – класс, в котором будет происходить тестирование методов контроллера. (Все методы котроллера доступные для вызова публичным API должны быть покрыты тестами)
   2. TestsProductService – класс, в котором будет происходить тестирование методов класса. (Все доступные для вызова методы класса должны быть покрыты тестами).
4. Провести интеграционный тест. Внедрить зависимость внутрь контроллера используя явную реализацию интерфейса.
5. Провести модульный тест контроллера с помощью Stub-ов и Mock-ов.

Ход выполнения лабораторной работы

В этом разделе будет размещена небольшой кусок теоретического материала.

**Stub (заглушка)** - это объект, который содержит заранее заданные данные и используется для ответов на вызовы во время тестирования. Она используется, когда вы не можете или не хотите задействовать объекты, которые отвечают реальными данными или имеют нежелательные побочные эффекты.

В качестве примера можно привести объект, которому нужно получить некоторые данные из базы данных, чтобы ответить на вызов метода. Вместо реального объекта мы создали заглушку и определили, какие данные должны быть возвращены.

**Mock (имитация)** — это объекты, которые регистрируют вызовы, которые они получают. В тестовом утверждении вы можете проверить, что все ожидаемые действия были выполнены. Вы используете макеты, когда не хотите вызывать рабочий код или когда нет простого способа проверить, что нужный код был выполнен. Нет возвращаемого значения и простого способа проверить изменение состояния системы. Примером может служить функция, которая вызывает службу отправки электронных писем.

Вы же не хотите отправлять электронные письма каждый раз, когда запускаете тест. Более того, в тестах нелегко проверить, что было отправлено правильное электронное письмо. Единственное, что вы можете сделать, — это проверить результаты работы функции, которая используется в нашем тесте. Другими словами, проверьте, что была вызвана служба отправки электронных писем.

Ссылки для самостоятельного изучения:

1. <https://softwarepatternslexicon.com/patterns-c-sharp/15/2/>
2. <https://www.csharp.com/UploadFile/dacca2/understand-stub-mock-and-fake-in-unit-testing/>
3. <https://www.iditect.com/faq/csharp/stubbing--mocking-a-database-in-c.html>
4. <https://peterdaugaardrasmussen.com/2022/05/15/csharp-how-to-mock-or-stub-a-class-without-an-interface/>