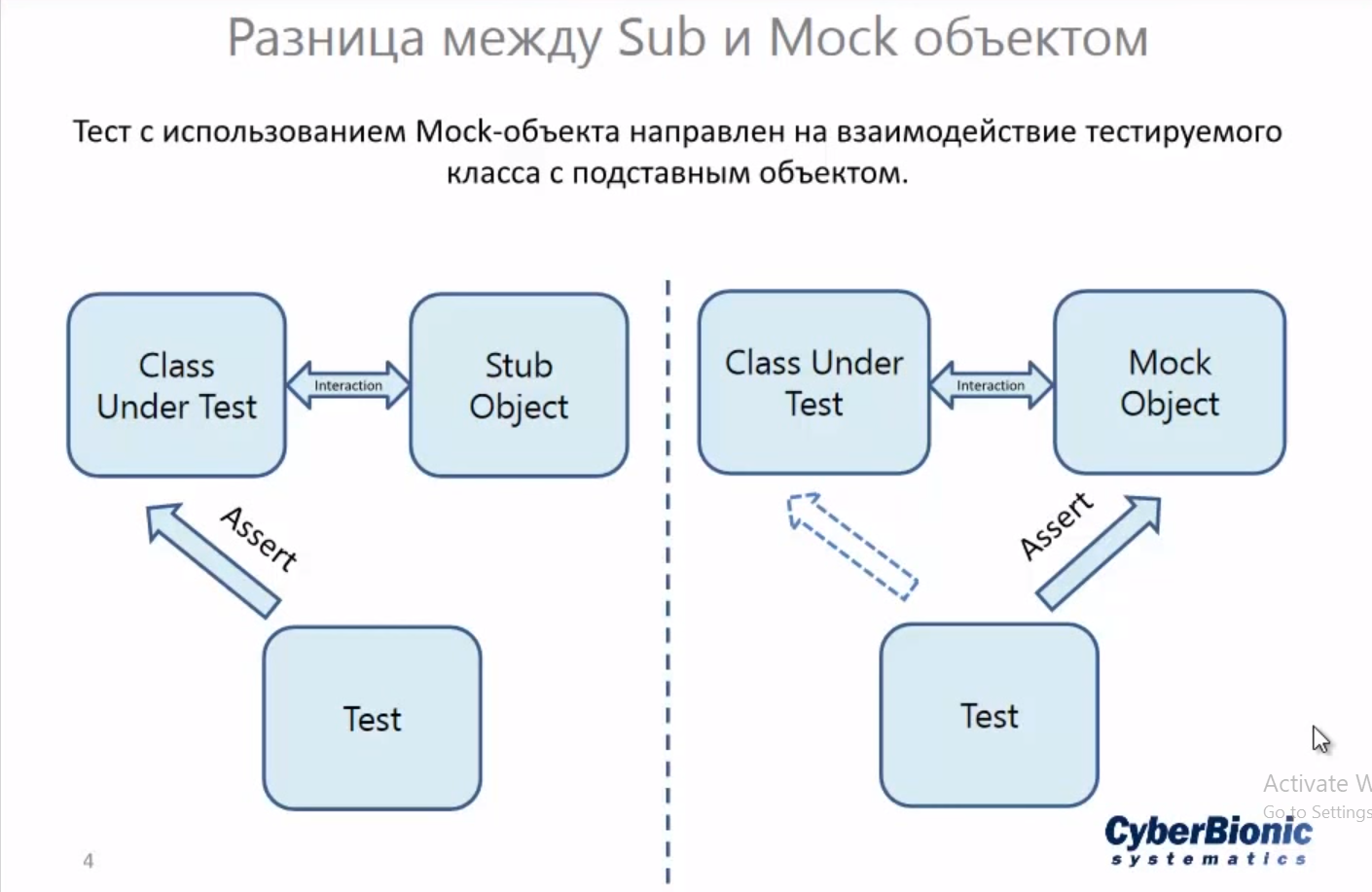
При юнит-тестировании вы проверяете отдельные модули кода. В любом приложении ваш код будет ссылаться и зависеть от других классов или модулей. По мере разработки, отношения между классами будут усложняться, будет тяжелее найти источник ошибки.Чтобы этого избежать, вы можете заменить эти зависимости **mocks** или **stubs**.

**Mock-объект** - это управляемая замена существующих зависимостей в системе. Mock- объекты заменяют реальные объекты системы и позволяют проверить вызовы своих членов тестируемым классом. Mock- объекты отличаются от Stub объектов тем, что они могут быть причиной не прохождения тестов.Главное различие между stubs и mocks заключается в том, что в одном случае мы управляем состоянием, а в другом - поведением.

Когда мы используем mocks, мы заменяем весь модуль на mock (ложный, тестовый объект, имитирующий настоящий). А stub - это функция, которая всегда выводит один и тот же результат, вне зависимости от того, что было подано на вход. Mocks используют для того, чтобы проверить, была ли функция вызвана с правильными аргументами, а stubs, чтобы протестировать, как функция работает с полученным ответом. Стабы нужны для проверки состояния метода, моки используются для регулировки поведения.



Пример использования **Mock:**

Мы не хотим отправлять электронные письма каждый раз, когда запускаем тест. Кроме того, в тестах нелегко проверить, что письмо на самом деле было отправлено. Единственно, что мы можем сделать — это проверить результаты функциональности, которые выполняется в нашем тесте. Другими словами — убедиться, что был вызван сервис отправки электронной почты.

Пример использования **Stubs:**

Вместо того, чтобы делать вызов к базе данных таблицы с оценками (gradebook), чтобы получить настоящие оценки студентов, мы предварительно настраиваем заглушку с оценками, которые будут возвращены. Мы определяем именно столько данных для тестирования, сколько нужно для вычисления средней оценки.