**Слайд 3**

*Тестирование веб-приложений — это процесс проверки и оценки веб-приложений для выявления ошибок и обеспечения их корректной работы. Оно необходимо для повышения качества продукта, улучшения пользовательского опыта и предотвращения возможных проблем на этапе эксплуатации. (на слайде)*

Тестирование (testing) - Процесс, содержащий в себе все активности жизненного цикла, как динамические, так и статические, **Активности жизненного цикла тестирования** — это этапы процесса тестирования, которые включают в себя определённую последовательность действий.

**Слайды 4-12**

Основная задача тестирования веб-приложения это выявление дефектов.

В тестировании важны следующие вещи:

*- Для начала проверить продукт на предмет того, соответствует ли он указанным требованиям? (на слайде)*

*-позитивное тестирование. Нужно убедиться, что поля работают правильно и выполняют свои основные задачи. Для этого проводят базовые проверки и позитивные сценарии, в которых пользователь делает то, что от него ожидается. (на слайде)*

*-негативное тестирование. После позитивных проверок нужно перейти к негативным — проверить, поля на ввод неправильных данных, будет ли приложение показывать сообщение об ошибке или нет?  (на слайде)*

*- Далее необходимо проверить веб-приложение на наличие дефектов, это может быть например отсутствие валидации в полях для заполнения данных, это значит что в поле предназначенным для числовых значений вы можете без труда ввести текст или что еще хуже, провести xss или sql инъекцию. (на слайде)*

*- составление отчета для начальства о проведенной работе. (на слайде)*

**Слайд (Виды тестирования веб-приложений)**

Существует несколько видов тестирования. Функциональное тестирование проверяет, выполняет ли приложение заявленные функции, в то время как нефункциональное тестирование оценивает характеристики, такие как производительность и безопасность. Сейчас мы разберем подробнее.

**Слайд Функциональное тестирование**

*Функциональное тестирование — это проверка, выполняет ли веб-приложение все необходимые функции. Например, мы можем протестировать форму регистрации, чтобы убедиться, что все поля работают корректно. Например, если пользователь не заполнил обязательные поля, система должна выдать соответствующее сообщение об ошибке. (на слайде)*

\*\*Комментарий к примеру:\*\* Проверка формы регистрации помогает убедиться, что пользователи не смогут зарегистрироваться без выполнения всех необходимых условий, что является критически важным для работы приложения.

- Функциональное тестирование

Это такое тестирование, которое проверяет возможность веб-приложения решать задачи для которых и было создано.

\*\*пример\*\*

**Слайд с ящиками**

К слову, раз уж мы упомянули функциональное тестирование, у него есть несколько видов, это черный ящик, серый ящик и белый ящик

\*\*обьяснение что это такое\*\*

**Слайд (нефункциональное тестирование)**

- Нефункциональное тестирование

Это тип тестирования веб-приложения для проверки нефункциональных частей приложения, это например – css-стили, скорость загрузки приложения, адаптивность дизайна, удобная навигация по сайту.

*Нефункциональное тестирование оценивает, как приложение работает, а не что оно делает. Например, мы можем провести тестирование юзабилити, чтобы понять, насколько удобно пользователю взаимодействовать с интерфейсом. Это включает в себя тестирование навигации, оформления и общего восприятия. (на слайде)*

\*\*пример\*\*

\*\*Комментарий к примеру:\*\* Юзабилити-тестирование позволяет выявить проблемные места в интерфейсе, которые могут затруднить пользователям выполнение задач, что в конечном итоге влияет на уровень удовлетворенности пользователей.

**Слайд (Регрессионное тестирование)**

- Регрессионное тестирование

Это проверка веб-приложения на наличие ошибок после внесения изменений в работающий код.

Строго говоря, вы внесли какие то изменения в приложения, добавили какие-нибудь фишки в структуру веб приложения для облегчения работы системы, но у вас вдруг отказал навбар к примеру, или какая нибудь формочка для заполнения данных, или кнопка.

Что бы не произошло такого что вы добавили что то новое в код, выгрузили его в продакшен, а потом оказалось что из за вашей новой фишечки сайт больше не работает для этого и существует регрессионное тестирование.

\*\*Комментарий к примеру:\*\* Регрессионное тестирование важно для поддержания стабильности приложения и предотвращения появления новых ошибок после изменений.

**Слайд (Тестирование производительности)**

- Тестирование производительности

Это проверка веб-приложения, направленное на определение его способности работать под большой нагрузкой. Будет ли веб приложение выдерживать то или иное количество пользователей или же сайт просто будет падать.

*Тестирование производительности оценивает, как приложение работает под нагрузкой. Например, мы можем провести нагрузочное тестирование, чтобы определить, сколько пользователей приложение может обрабатывать одновременно без замедления работы. Это включает в себя симуляцию большого количества пользователей, чтобы выявить узкие места в системе. (на слайде)*

\*\*Комментарий к примеру:\*\* Нагрузочное тестирование помогает выявить узкие места в системе и гарантирует, что приложение сможет справляться с ожидаемыми объемами трафика, что критично для успешного функционирования веб-сервиса. Это особенно важно в периоды пиковых нагрузок, например, во время распродаж или акций.

**Нагрузочное тестирование**. Проверка системы на работоспособность под определённой нагрузкой, например, при одновременной работе определённого количества пользователей или при выполнении определённого количества запросов.

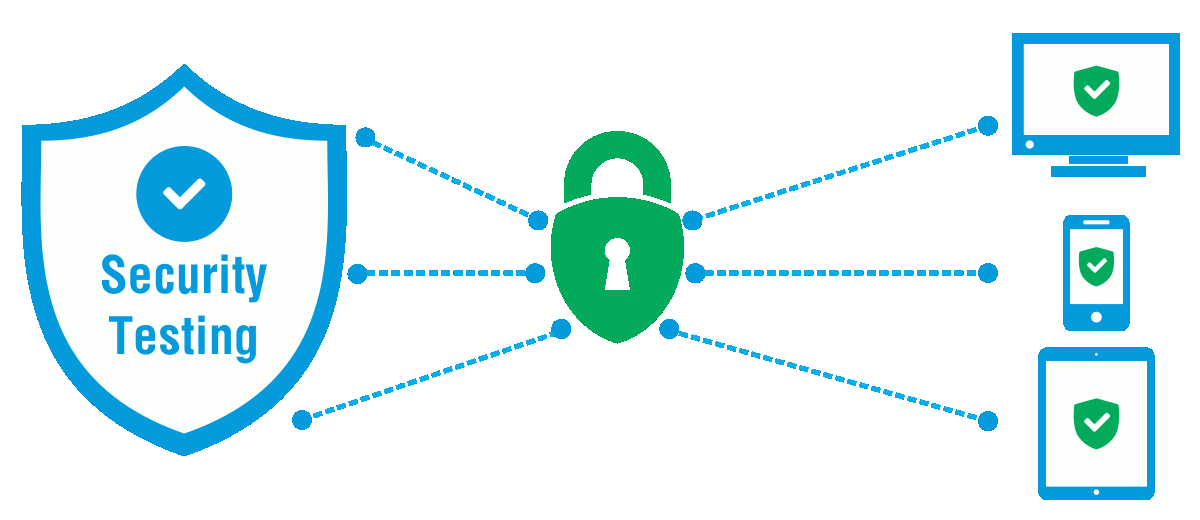
**Стресс-тестирование**. Определение предельных рабочих условий системы, при которых она продолжает функционировать, но с ограничениями по производительности, а также выявление точек отказа системы.

**Тестирование стабильности**. Проверка системы на способность работать без сбоев и с заданной производительностью на протяжении длительного времени.

**Тестирование масштабируемости**. Определение способности системы адекватно функционировать при увеличении рабочих нагрузок или количества пользователей.

**Слайд (Тестирование безопасности)**

- Тестирование безопасности



Тестирование безопасности это тестирование направленное на проверку стойкости веб приложения при попытках злоумышленников получить данные пользователей или попытки положить сервер через DDoS атаки.



Вообще тестирование безопасности включает в себя все самые опасные виды тестирования, это sql иньекции, Xss иньекции, перехват трафика, XPath инъекции и другие.

Тестирование безопасности направлено на выявление уязвимостей в веб-приложении, которые могут быть использованы злоумышленниками. Например, мы можем протестировать приложение на наличие SQL-инъекций, чтобы убедиться, что злоумышленники не могут получить доступ к базе данных через уязвимости в коде. (на слайде)

Комментарий к примеру: Тестирование на безопасность критически важно для защиты данных пользователей и предотвращения утечек информации, что может привести к серьезным последствиям для компании, включая финансовые потери и ухудшение репутации.

**Слайд (Тестирование юзабилити)**

- Тестирование юзабилити

Это пожалуй самый безобидный вид тестирования. Это метод оценки того, насколько легко конечным пользователям взаимодействовать с веб-приложением.

*Юзабилити-тестирование – это метод оценки того, насколько легко конечные пользователи могут взаимодействовать с веб-сайтом, приложением, продуктом или системой. Цель этого процесса заключается в выявлении проблем юзабилити (удобства использования). (на слайде)*

**Слайд (заключение)**

*Тестирование веб-приложений — это ключевой этап разработки, который помогает обеспечить качество и надежность продукта. Важно внедрять различные виды тестирования на всех этапах разработки, чтобы минимизировать риски и повысить удовлетворенность пользователей. Регулярное тестирование позволяет выявлять и устранять проблемы на ранних стадиях, что в конечном итоге экономит время и ресурсы. (на слайде)*