

1. Введение

1.1. Основная информация

Документ описывает методы и подходы к тестированию, которые будут использоваться тестировщиками отдела тестирования для тестирования сайта и приложения.

План тестирования может использоваться как тестировщиками, так и менеджерами, разработчиками.

Объект тестирования — деятельность, направленная на проверку работоспособности функций сайта «БелкаГид».

1.2. Цель

Тест-план проекта преследует следующие цели:

1. Определить существующую информацию о проекте и программных компонентах, подлежащих тестированию.
2. Описать стратегии тестирования, которые будут использоваться.
3. Определить необходимые ресурсы для проведения работ по тестированию.
4. Привести результаты тестирования.

2. Рамки проекта

2.1. Область тестирования сайта

В объем работ по тестированию сайта входит тестирование следующих компонентов и функций:

1. Загрузка страниц и соответствие адресации.
2. Валидация форм
3. Создание нового заказа на гида
4. Экспорт маршрута в Яндекс.Карты
5. Авторизация в личный кабинет
6. Фильтрация карточек мест
7. Редирект в карточке событий
8. Система избранных мест
9. Устойчивость сайта к работе в нагрузке

3. План работы

Стороны договорились следовать следующему плану работы:

1. Подготовка плана тестирования.
2. Согласование тест-плана.
3. Функциональное тестирование и отчеты об ошибках.
4. Подготовка финального отчета

4. План и стратегия тестирования

4.1. Модульное (unit) тестирование

Unit-тестирование, или модульное — это разновидность тестирования в программной разработке, которое заключается в проверке работоспособности отдельных функциональных модулей, процессов или частей кода приложения. Unit-тестирование позволяет избежать ошибок или быстро исправить их при обновлении или дополнении ПО новыми компонентами, не тратя время на проверку программного обеспечения целиком.

4.2. Интеграционное тестирование

Главная цель интеграционного тестирования — подтвердить, что различные программные компоненты, модули и подсистемы работают вместе как единая система, обеспечивая требуемую функциональность и производительность.

4.3. Ручное тестирование

Ручное тестирование — это процесс поиска ошибок в программе без использования специальных ПО, силами человека. Тестировщик имитирует реальные действия пользователя и старается охватить максимум функций продукта.

4.4. Нагрузочное тестирование

Нагрузочное тестирование — это проверка устойчивости и производительности программного обеспечения под нагрузкой, сопоставимой с реальными условиями использования. Цель тестирования — выявить слабые места в архитектуре веб-сервиса, оценить границы нагрузки, при которых система ещё способна работать без ошибок или проблем с сервером, определить ресурсы, необходимые для безотказной работы, и оценить, насколько система справляется с увеличением нагрузки.

4.5. Сообщения об ошибках

Отчеты об ошибках создаются для того, чтобы предоставить команде разработчиков и руководителю проекта исчерпывающую информацию об обнаруженных ошибках. Они должны быть полезны при определении причин ошибок и их исправлении.

Степень серьезности ошибок можно разделить на четыре категории:

1. Критические (блокирующие) дефекты - сбой всей программной системы или критической подсистемы, тестирование не может быть выполнено после возникновения дефекта. Это также относится к сбоям при потере данных и процессам, которые оставляют несогласованные данные, хранящиеся в базе данных.
2. Мажорные дефекты — ошибки, которые также приводят к выходу из строя всей или части системы, но существуют некоторые альтернативы обработки, которые позволяют продолжить работу системы. Это также относится к сбою или прерыванию работы системы во время нормальной работы некритического потока.
3. Незначительные дефекты не приводят к сбою, но приводят к тому, что система показывает неправильные, неполные или противоречивые результаты.
4. Тривиальные дефекты - небольшие ошибки, которые не влияют на функциональность: опечатки, грамматические ошибки, неправильная терминология и т.д.

Информация, указанная в каждом отчете об ошибке:

1. Название программного продукта.
2. Номер версии программного продукта.
3. Браузер, в котором проводились тесты.

Каждый баг-репорт содержит следующую информацию о дефекте:

1. Краткое описание, представляющее собой краткое описание проблемы.
2. Местонахождение дефекта в программном продукте.
3. Шаги для воспроизведения ошибки;
4. Частота возникновения дефекта.
5. Тяжесть дефекта.
6. Дополнительная информация о дефекте в виде прикрепленных скриншотов или видеозаписей.

5. Ресурсы

5.1. Инструменты

Будут использованы следующие инструменты:

Наименование процесса	Инструмент
Модульное (unit) тестирование	Django test suite
Интеграционное тестирование	Selenium
Нагрузочное тестирование	Locust
Захват изображений	Screenshots
Подготовка документации	Microsoft Word

6. Критерии качества

Продукт должен работать в соответствии с требованиями и техническим. Продукт не должен содержать критических и блокирующих дефектов в окончательной версии проекта.

7. Риски процесса тестирования

Следующие проблемы могут повлиять на результаты тестирования:

1. Изменения и модификации программного продукта, которые не были запланированы и не обсуждались заранее с командой тестирования.
2. Изменения в требованиях к программному обеспечению, которые не были предварительно обсуждены с командой тестирования.
3. Задержки в исправлении/исправлении ошибок.
4. Задержки в доставке новых сборок команде тестирования.

8. Ожидания команды тестирования

Команда тестирования должна быть обеспечена действительными, обновленными документами в течение всего процесса тестирования. Все необходимое оборудование, приборы, устройства и программное обеспечение должны быть приобретены и подготовлены до начала процесса тестирования. Все критические ошибки должны быть исправлены как можно скорее.

Примечание к выпуску следует добавлять к каждому выпуску программного обеспечения для команды тестирования. В примечании должно быть объяснено, какие элементы, функции и функции были добавлены в программу и как эти дополнения влияют на программное обеспечение. Разработчики должны исправить все ошибки в программных модулях перед выпуском новой версии.

9. Результаты

1. План тестирования.
2. Отчеты об ошибках и ходе тестирования.