

План тестирования

Компания	ООО «КЕХ eКоммерц»
Дата составления	27.08.2023
Версия	1.0.1
Автор	Балушева Е.Ю.

Содержание

1. Введение
 - 1.1. Основная информация
 - 1.2. Цель документа
 - 1.3. Цели тестирования
2. Область тестирования
3. План работы
4. Тест-план и стратегия тестирования
 - 4.1 Стратегия тестирования
 - 4.2. Процедура тестирования
 - 4.3. Функциональное тестирование
 - 4.4. Автоматизированное тестирование
 - 4.5. Отчеты об ошибках
 - 4.6. Регистрация дефектов
5. Используемые ресурсы
 - 5.1 Тестовая команда
 - 5.2. Инструменты
 - 5.3. Список браузеров
6. Критерии качества
7. Риски процесса тестирования
8. Результаты

1. Введение

1.1. Основная информация

Данный документ описывает процесс, методы и подходы к тестированию, которые будут использоваться тестировщиками отдела тестирования, для тестирования функциональности веб-сервиса для размещения объявлений. План тестирования может использоваться всеми членами команды разработки.

Объект тестирования — это деятельность, направленная на проверку работоспособности определенной функциональности Desktop-версии интернет-сервиса в браузере Chrome в операционных системах Windows и macOS.

1.2. Цель документа

План тестирования преследует следующие цели:

- Определить существующую информацию о проекте и программных компонентах, подлежащих тестированию.
- Описать стратегии тестирования, которые будут применяться.
- Обеспечить полноту тестирования в соответствии с требованиями, охват всех необходимых типов тестирования.

- Определить методы регистрации дефектов, необходимого программного обеспечения.
- Определить критерии качества.
- Определить необходимые ресурсы для проведения работ по тестированию.
- Описать виды предоставляемых отчетов о результатах тестирования.

1.3. Цели тестирования

Основными целями тестирования являются:

- Выявление проблем, связанных с несоответствием требованиям к разрабатываемой функциональности;
- Отслеживание статуса проблем;
- Предоставление оценки качества продукта;
- Снижение рисков проекта, связанных с качеством разрабатываемой функциональности.

2. Область тестирования

В объем работ по тестированию входит регрессионное тестирование реализованного ранее функционала (компонентов и функций). Проверка проводится только для неавторизованного пользователя:

1. Добавление объявления в «Избранное» со страницы объявления:
 - a. Попытка добавить объявление в "Избранное".
 - b. Проверка отсутствия уведомления о необходимости авторизации.
 - c. Проверка успешного добавления объявления в список "Избранное".
2. Добавление в "Избранное" со страницы поиска:
 - a. Попытка добавить объявление в "Избранное".
 - b. Проверка отсутствия уведомления о необходимости авторизации.
 - c. Проверка успешного добавления объявления в список "Избранное".
3. Проверка списка избранных товаров:
 - a. Проверка успешного перехода на страницу "Избранное".
 - b. Просмотр списка "Избранное" для неавторизованного пользователя.
 - c. Проверка отсутствия уведомления о необходимости авторизации.
 - d. Проверка отображения добавленных объявлений в списке "Избранное".
4. Удаление из "Избранного":
 - a. Для каждого объявления в списке "Избранное":
 - Удаление объявления из "Избранного".
 - Проверка успешного удаления объявления из списка "Избранное".
5. Проверка отображения на Desktop-платформе:
 - a. Открытие интернет-сервиса в браузере Chrome.
 - b. Проведение всех вышеуказанных действий.
6. Проверка корректного переключения между вкладками на странице "Избранное".
7. Проверка на отображение пустого списка "Избранное".
8. Попытка добавления большого количества объявлений в "Избранное" и проверка ограничений.
9. Попытка добавления объявлений в "Избранное" из разных категорий:
 - a. Переход по категориям, добавление объявлений любым из двух способов.
 - b. Проверка, что все добавленные объявления присутствуют в списке "Избранное".
10. Проверка отображения уведомлений при успешном добавлении и удалении объявлений из "Избранное".
11. Проверка на сохранение данных:
 - a. Проверка, что данные в списке "Избранное" сохраняются после перезапуска приложения.

12. Проверка, что ссылки на объявления в "Избранном" ведут на правильные страницы с объявлениями.
13. Проверка отображения всплывающих сообщений при возникновении ошибок во время операций с избранными товарами.
14. Проверка корректности работы функциональности "Избранное" на разных операционных системах.
15. Проверка динамического обновления списка избранных товаров без необходимости перезагрузки страницы.
16. Проверка, что объявления, которые были удалены или изменены, верно обрабатываются в "Избранном".
17. Проверка правильности сортировки объявлений в списке "Избранное".
18. Проверка корректного отображения количества добавленных объявлений в "Избранное".

3. План работы

Обозначен следующий план работы:

1. Подготовка плана тестирования.
2. Согласование тест-плана.
3. Проведение тестирования, заведение баг-репортов.
4. Подготовка финального отчета.

В процессе проведения работ по тестированию предполагается проведение регрессионного тестирования – проверка исправления дефектов, зарегистрированных на первых итерациях, тест новой функциональности.

4. Тест-план и стратегия тестирования

4.1. Стратегия тестирования

Основным методом проверки данного программного продукта будет ручное функциональное тестирование с использованием метода «черного ящика», который базируется на использовании требований и спецификаций, и не предполагает наличия каких-либо специальных знаний о конфигурации и внутренней структуре объекта испытаний.

4.2. Процедура тестирования

Типы тестирования, которые должны быть выполнены:

1. Функциональное тестирование.
2. Автоматизированное тестирование.

Также будет проверено, как продукт запускается в браузере Chrome на различных операционных системах, сколько времени требуется для запуска.

Цель тестирования:	Обеспечение надлежащего качества целевой функциональности
Техника:	Выполнение каждого сценария, используя допустимые и недопустимые данные: <ul style="list-style-type: none"> • Ожидаемые результаты возникают при использовании достоверных данных. • Соответствующие сообщения об обнаруженных ошибках или предупреждения отображаются, когда используются неверные данные. • Каждый пункт проверен.
Входные критерии:	Подготовлено тестовое окружение, тестируемая задача задеплоена на тестовый стенд.
Критерий приемки:	<ul style="list-style-type: none"> • Все дымовые тесты пройдены.

	<ul style="list-style-type: none"> • Нет блокирующих багов. • Все баги с высоким приоритетом поправлены. • Ручные тесты пройдены. • Автоматические тесты пройдены.
--	--

4.3. Функциональное тестирование

Цель функционального тестирования состоит в том, чтобы убедиться, что программный продукт работает в соответствии с требованиями, и в приложении не появляются существенных ошибок. Функциональное тестирование включает в себя проверку правильности реализации характеристик, заложенных в продукт. Программный продукт должен пройти все запланированные тесты. Т.к. только в этом случае можно быть уверенным в его качестве.

Будет включать два этапа:

Название	Описание
Минимальный приемочный тест (Smoke test)	Короткий тест, проверяющий основную функциональность. По результатам приемочного теста принимается решение о продолжении проведения дальнейшего тестирования. Основная задача – это оперативное отслеживание регрессии качества и получение высокоуровневой картины работоспособности продукта.
Регрессионное тестирование	Для обеспечения надлежащего качества продукта на каждом последующем этапе будет производиться регрессионное тестирование функциональности, разработанной на предыдущей стадии и ее совместимости с новой функциональностью. Такое тестирование подразумевает под собой повторное прохождение всех тестовых случаев протестированных после завершения предыдущего этапа разработки, проверки исправления дефектов, и, что при их исправлении не внесены новые.

4.4. Автоматизированное тестирование

Основной целью автоматизации тестирования является решение сложных задач, выполнение которых не может быть осуществлено вручную в рамках сжатых сроков при минимальном привлечении ресурсов.

Автоматизированное тестирование подразумевает под собой использование специального программного обеспечения для проведения автоматизированных тестов на основе набора тестовых данных и написанных скриптов. Осуществляется разработчиком автоматизированных тестов.

4.5. Отчеты об ошибках

Отчеты об ошибках создаются для того, чтобы предоставить команде разработчиков и менеджеру проекта исчерпывающую информацию об обнаруженных ошибках. Они должны быть полезны при определении причин, вызывающих эти ошибки, с целью их последующего исправления.

Можно выделить четыре категории ошибок по степени их серьезности:

- Критические дефекты (Critical) – это сбой всей программной системы или критической подсистемы, тестирование не может быть выполнено после возникновения дефекта.
- Серьезные дефекты (Major) — ошибки, которые также приводят к некорректной работе части функциональности, но при этом существуют некоторые альтернативы обработки, которые позволяют продолжить работу.
- Дефекты со средним приоритетом (Average) - не приводят к сбою, но приводят к тому, что система показывает неправильные, неполные или противоречивые результаты.
- Дефекты с низким приоритетом (Minor) - небольшие ошибки, которые не влияют на функциональность: опечатки, грамматические ошибки, неправильная терминология и т.д.

4.6. Регистрация дефектов

Стороннее программное обеспечение будет использоваться для отчетности об обнаруженных ошибках. Средством регистрации и отслеживанием текущего состояния дефектов выбран инструмент Qase. Выбор обусловлен необходимостью постоянного контроля заинтересованными сторонами появления новых дефектов, возможностью своевременного управления статусами и отсутствие необходимости составления дополнительных отчетов, содержащих информацию о качестве. Будет установлен локально.

Каждый баг-репорт содержит следующую информацию о ошибке:

- Идентификационный номер
- Краткое описание - привязка дефекта к функции
- Предусловия
- Местонахождение дефекта в продукте.
- Шаги для воспроизведения ошибки;
- Важность дефекта.
- Приоритет дефекта.
- Дополнительная информация о дефекте в виде прикрепленных скриншотов или видеозаписей.
- Браузер, в котором проводились тесты.

Команда тестирования регистрирует в бег-трекинговой системе все дефекты, которые будут обнаружены в процессе тестирования.

5. Используемые ресурсы

5.1. Тестовая команда

Роль	Ответственность
Руководитель группы тестирования	<ul style="list-style-type: none">● Контактное лицо по проектным вопросам;● Сбор и изучение требований;● Планирование работ, написание проектной тестовой документации, разработка сценариев тестирования;● Отчеты о ходе выполнения, количестве и серьезности обнаруженных ошибок.● Разработка наборов тестовых данных для автоматизации тестирования.
Функциональный тестировщик	<ul style="list-style-type: none">● Функциональное тестирование;● Разработка сценариев тестирования, тест-кейсов;● Регистрация обнаруженных ошибок в системе отслеживания ошибок.
Тестировщик-автоматизатор	<ul style="list-style-type: none">● Автоматизация функционального тестирования:<ul style="list-style-type: none">1) Разработка наборов тестовых данных;2) Подготовка среды для проведения автоматизированных тестов;3) Разработка скриптов для автоматизации тестирования.

5.2. Инструменты

Будут использованы следующие инструменты:

Наименование процесса	Инструмент
Баг-трекинговая система	Jira
Тест-кейсы	Qase
Выполнение автотестов	Selenium + Allure

5.3. Список браузеров

Наименование браузера	Версия браузера
Chrome	последняя (стабильная)

6. Критерии качества

Продукт должен работать в соответствии со всеми заданными требованиями и техническим заданием, если они были поставлены.

Продукт не должен содержать критических и блокирующих дефектов в финальной версии.

7. Риски процесса тестирования

Приведен перечень проблем, которые могут повлиять на результаты тестирования:

- Изменения программного продукта, которые не были запланированы и не обсуждались заранее с командой тестирования и разработки.
- Изменения в требованиях к функциональности продукта, которые не были предварительно обсуждены с руководством и командой.
- Задержки в исправлении выявленных багов.
- Задержки в доставке новых сборок для команды тестирования.

Предупредительные меры:

- Тестовая документация должна иметь адаптивный характер, предусмотреть возможность быстрого внесения изменений (новых требований) и минимизирования затрат на проведение дополнительных тестов.
- Своевременная оповещение группой тестирования команды разработчиков обо всех новых дефектах, идентифицированных при проведении функциональных и технических тестов.
- Чёткое планирование временных затрат на выполнение поставленных задач, своевременная оповещение обо всех проблемах, возникающих в процессе тестирования и исправления дефектов и внедрения новой функциональности.

8. Результаты

- План тестирования.
- Отчеты по результатам тестов.
- Отчеты об ошибках и отчеты о ходе тестирования.