Rocket.Chat

Тестовая стратегия

Версия 1.0

История изменений

|  |  |  |  |
| --- | --- | --- | --- |
| ***Дата*** | ***Версия*** | ***Описание*** | ***Автор*** |
| **22/07/2023 – 23/07/2023** | 1.0 | Тестовая стратегия, тестовая документация | Сапожников М.И.  rickiest@student.21-school.ru |
| **23/07/2023 – 24/07/2023** | 1.0 | Use Case сценарий использования Тест-кейс | Сапожников М.И.  rickiest@student.21-school.ru |
| **24/07/2023 – 26/07/2023** | 1.0 | Тестирование:  - (UI) пользовательского интерфейса  - Удобства использования - Производительности | Сапожников М.И.  rickiest@student.21-school.ru |
| **27/07/2023** | 1.0 | Завершение тестирования | Сапожников М.И.  rickiest@student.21-school.ru |

Оглавление

[**1. Введение**………………………………………………………………………………….....**4**](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#_Toc137505097)

1.1 Общие сведения …………………………………………………………………..…**4**

1.2 Цель………………………………………………….……………………………......**4**

1.3 Исполнитель……………………………………………………………………….…**4**

1.4 Реализованые проекты………………………………………………………………………..**5**

[**2. Типы проводимого тестирования** ………………………………………..……………..**6**](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#_Toc137505098)

[2.1 Функциональное тестирование…………………………………………………….….](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#__RefHeading___Toc7121_3589186699).**6**

[2.2 Тестирование производительности…………………………………………………](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#__RefHeading___Toc3346_3422863242)....**6**

[2.3 Юзабилити тестирование ..……………………………………………………](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#__RefHeading___Toc3348_3422863242)…….....**7**

[2.4 Тестирование конфигурации…….…………………………………………………..](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#__RefHeading___Toc3350_3422863242)...**7**

[2.5 Тестирование безопасности………………………………………………………](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#__RefHeading___Toc3352_3422863242)…....**8**

[2.6 Тестирование на отказ и и восстановление после сбоев....…………………….…](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#__RefHeading___Toc3354_3422863242).....**9**

[**3. Модули, которые будут протестированы** …………………………………………..…**10**](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#_Toc137505101)

[3.1 Профиль…………………………………………………………..……………….....**10**](#_Toc141086088)

[3.2 Каналы……………………………………………………………………………….**10**](#_Toc141086089)

[3.3 Панель администрирования………………………………………………...………**10**](#_Toc141086090)

3.4 Модуль "Взаимодействия с другими пользователями Чата"…………………….**12**

[3.5 Домашняя страница…………………………………………………….………...…**12**](#_Toc141086091)

[**4. Окружение для работы. Описание операционных систем, версии приложения и браузеров**……………………………………………………..…………………………………**13**](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#_Toc137505102)

[**5. Виды тестовой документации**, которые будут составляться в процессе тестирования, обоснование выбора. Какие техники тест-дизайна будут использоваться при формировании тест-кейсов…………………………………………………………………………………….…**14**](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#_Toc137505103)

**6. План тестирования** согласно заданию. Когда тестирование можно будет считать завершённым?.…………………………………………………………..……………………….**15**

[**7. Время проведения тестирования** (время начала, время окончания каждого типа тестирования)................................................................................................................................**16**](file:///D:\Рабочий%20стол\диплом\акула%20топ%202.docx#_Toc137505104)

Тестовая стратегия

# Введение

## Общие сведения

*Rocket.Chat был основан Габриэлем Энгелем как проект с открытым исходным кодом.  
- Rocket.Chat служит приложением Интернета вещей (IoT) и хорошо работает на небольших .  
- Он также работает с несколькими сервисами*

*- Платформа поддерживает чат, обмен файлами, видеоконференции, службу поддержки и службы голосовых сообщений.*

Документ описывает методы и подходы к тестированию, которые будут использоваться для тестирования веб-версии чата Rocket.Chat.

**Rocket.Chat** — корпоративная платформа обмена сообщениями, которая позволяет командам общаться и сотрудничать в режиме реального времени. Корпоративный мессенджер предоставляет множество функций для видеозвонков и аудиоконференций, обмена сообщениями и файлами, совместного использования экрана и многого другого.

Rocket.chat является одним из популярных программ живого чата, отчасти - проект с открытым исходным кодом в прямом эфире. Встроенные в Rocket.Chat инструменты помогают компаниям и командам улучшить корпоративную коммуникацию, сотрудничество и производительность.

Особенности Rocket.Chat:

- Общение в каналах, групповых и личных чатах;

- Обеспечение защиты данных;

- Интеграция с облачными сервисами;

- Передача файлов;

- Персонализация мессенджера;

- Размещение на выделенных серверах.

## Цель

**Тестирую:** Веб версия Rocket.Chat, приложение нужно протестировать полностью, это последняя версия.

Составление тестовой стратегии имеет следующие цели:

- Ознакомиться с информацией, руководством использования веб-приложения Rocket.Chat.

- Определить компоненты приложения, подлежащие тестированию.

- Описать типы тестирования, которые будут использоваться.

- Определить необходимые ресурсы для проведения работ по тестированию.

- Привести результаты тестирования в виде отчетов.

Все найденные ошибки будут отслеживаться с помощью баг-трекинговой системы.

## Исполнитель

##### QA-инженер/Тестировщик цифровых продуктов Сапожников Максим Игоревич (логин на платформе «Школа 21»: [rickiest@student.21-school.ru](mailto:rickiest@student.21-school.ru) )

**Работал в проектах, используя следующие инструменты ручного тестирования:** Notion, Test IT, Saucedemo, FakeRESTApi.Web V1, Postman, Fiddler, Chrome DevTools, Appetize, Android Studio, GitLab, Git Bash и другие.

**1.4 Реализованные проекты:**

* Проведение UI-тестирования модального окна авторизации на странице <https://sbermarket.ru/>.
* Для демо-версии интернет-магазина <https://www.saucedemo.com/>:
  1. составление списка сценариев позитивного тестирования;
  2. составление тест-кейсов на основе сценариев;
  3. составление чек-листа с результатами прогона по тест-кейсам.
* Описание сценариев использования модального окна авторизации на странице <https://sbermarket.ru/>.
* Проведение проверки полей на странице авторизации <https://www.saucedemo.com/> с использованием техники попарного тестирования.
* Для тестовой формы авторизации <http://testingchallenges.thetestingmap.org/login/login.php>:
  1. проведение интуитивного тестирования в соответствии с требованиями;
  2. составление баг-репорта в случае расхождения фактического результата с требованиями.
* Для системы TestIT:
  1. в веб-приложении Notion описаны все необходимые сценарии для тестирования всех функциональных возможностей страницы "Ручные тесты" в системе TestIT;
  2. в TestIT на основе созданных сценариев составлены тест-кейсы.
* Для первых двух страниц ("1. Выбор полиса" и "2. Оформление") сайта [СберСтрахования](https://online.sber.insure/store/propertyins/) описаны XPath's каждого элемента.
* Создание тест-кейсов, позволяющих провести полное ручное UI-тестирование официального сайта [TestIT](https://testit.software/" \o "https://testit.software/).
* Для [Swagger тестового API](https://fakerestapi.azurewebsites.net/index.html) описаны все возможные запросы (27 эндпоинтов).
* Составление тест-плана с оптимальным набором тест-кейсов для проверки всех эндпоинтов в Fake REST API.
* Проведение тестирования API с Postman, используя переменные окружения.
* Для веб-приложения [СберМегаМаркета](https://sbermegamarket.ru/):
  1. поиск ошибок в DevTools в и составление баг-репортов;
  2. поиск запросов в DevTools, отправленных на сервер для различных действий в веб-приложении;
  3. используя Fiddler отловлены запросы для различных действий в веб-приложении.
* Проведение мобильного тестирования нативного и веб-приложения «tutu».
* Добавление объектов в веб-хранилища "Session Storage" и "Local Storage".
* Создание SQL-запросов к базам данных тестового [сайта](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_in).

# Типы проводимого тестирования

## Функциональное тестирование

Функциональное тестирование веб-приложения - это процесс проверки функциональности приложения с целью убедиться, что оно работает должным образом и соответствует требованиям и ожиданиям пользователей.

В ходе функционального тестирования веб-приложения «Rocket.Chat» проверяются следующие аспекты:

1. Навигация: проверка работы ссылок, кнопок и меню, чтобы убедиться, что пользователь может свободно перемещаться по приложению.

2. Формы: проверка корректности работы форм, включая ввод данных, обработку ошибок и сохранение данных.

3. Аутентификация и авторизация: проверка работы системы входа в приложение и управления доступом пользователей к различным функциям и данным.

4. Функциональность: проверка работы всех функций приложения, включая создание, чтение, обновление и удаление данных, а также выполнение других задач, предусмотренных приложением.

5. Интеграция: проверка взаимодействия приложения с другими системами или сервисами, такими как базы данных, платежные шлюзы или сторонние API.

Функциональное тестирование веб-приложения может выполняться как вручную, так и с использованием автоматизированных инструментов тестирования.

## UI тестирование

UI тестирование веб-приложения - это процесс проверки пользовательского интерфейса приложения с целью убедиться, что он работает должным образом и соответствует требованиям и ожиданиям пользователей.

В ходе UI тестирования веб-приложения «Rocket.Chat» проверяются следующие аспекты:

1. Внешний вид: проверка соответствия дизайна и визуального оформления приложения заданным требованиям и стандартам.

2. Интерактивность: проверка работы элементов пользовательского интерфейса, таких как кнопки, поля ввода, выпадающие списки и другие элементы, чтобы убедиться, что они реагируют на действия пользователя.

3. Навигация: проверка удобства и интуитивности навигации по приложению, включая расположение элементов управления, меню и ссылок.

4. Адаптивность: проверка корректности отображения приложения на различных устройствах и экранах, включая мобильные устройства, планшеты и компьютеры.

5. Мультимедиа: проверка работы медиа-элементов, таких как изображения, видео или аудио, чтобы убедиться, что они отображаются и воспроизводятся корректно.

6. Локализация: проверка работы приложения на разных языках и культурах, чтобы убедиться, что тексты и переводы отображаются правильно.

UI тестирование веб-приложения выполняется вручную, поскольку требует визуальной оценки пользовательского интерфейса. Однако, в некоторых случаях, можно использовать автоматизированные инструменты для выполнения некоторых аспектов UI тестирования, таких как проверка расположения элементов или проверка доступности.

**2.3 Тестирование удобства использования**

Тестирование удобства использования веб-приложения включает в себя проверку, насколько легко и интуитивно пользователи могут взаимодействовать с приложением и выполнять необходимые им действия.

В ходе тестирования удобства использования «Rocket.Chat» проверяются следующие аспекты:

1. Легкость навигации: проверка, насколько просто пользователи могут перемещаться по различным разделам и страницам приложения с помощью меню, ссылок или кнопок.

2. Простота использования: проверка, насколько интуитивно понятны элементы управления и функции приложения для пользователей. Например, проверка, насколько легко пользователям разобраться, как добавить новую запись или выполнить поиск.

3. Понятность сообщений и инструкций: проверка, насколько понятными и информативными являются сообщения об ошибках, подсказки и инструкции для пользователей. Например, проверка, насколько понятно пользователю, что он должен заполнить определенные поля перед отправкой формы.

4. Эффективность использования: проверка, насколько быстро и легко пользователи могут выполнить необходимые им задачи с помощью приложения. Например, проверка времени, которое требуется пользователям для выполнения определенной операции.

5. Консистентность: проверка, насколько последовательным и однородным является пользовательский интерфейс приложения. Например, проверка, насколько одинаково выглядят и работают различные элементы управления на разных страницах.

6. Обратная связь: проверка, насколько ясно и наглядно приложение сообщает пользователям о результатах их действий. Например, проверка, насколько понятно пользователю, что его действие было успешно выполнено или что произошла ошибка.

В ходе тестирования могут использоваться различные методы, такие как: тестирование сценариев использования, анализ задач пользователей или проведение экспертных оценок.

**2.4 Тестирование производительности**

Тестирование производительности веб-приложения включает в себя проверку его скорости, отзывчивости и способности обрабатывать большую нагрузку.

В ходе тестирования производительности «Rocket.Chat» проверяются следующие аспекты:

1. Загрузка страниц: проверка, насколько быстро приложение загружает страницы и отображает их содержимое. Это может включать проверку времени загрузки главной страницы, страниц с большим количеством контента или страниц, содержащих медиафайлы.

2. Отклик интерфейса: проверка, насколько быстро приложение реагирует на действия пользователей, такие как нажатие кнопок или ввод текста. Например, проверка времени, которое требуется приложению для обработки запросов пользователя.

3. Обработка большой нагрузки: проверка, насколько хорошо приложение справляется с большим количеством одновременных пользователей или запросов. Например, проверка, насколько приложение остается отзывчивым при одновременном выполнении нескольких операций.

4. Оптимизация ресурсов: проверка, насколько эффективно приложение использует ресурсы сервера, такие как память или процессорное время. Например, проверка, насколько приложение оптимизировано для работы с базой данных или обработки больших объемов данных.

5. Масштабируемость: проверка, насколько хорошо приложение масштабируется при увеличении нагрузки или количества пользователей. Например, проверка, насколько приложение продолжает работать стабильно при увеличении количества одновременных пользователей.

Тестирование производительности веб-приложения выполняется с помощью специализированных инструментов и методов, таких как нагрузочное тестирование, стресс-тестирование или профилирование производительности. В ходе тестирования анализируются результаты и определяются возможные узкие места или проблемы производительности, которые требуют оптимизации или улучшения.

**2.5 Тестирование совместимости**

Тестирование совместимости веб-приложения включает проверку работы приложения на различных платформах, браузерах и устройствах. Это важно, так как пользователи могут использовать разные операционные системы (Windows, macOS, Linux), различные версии браузеров (Chrome, Firefox, Safari, Internet Explorer) и разные устройства (настольные компьютеры, ноутбуки, планшеты, мобильные телефоны).

В процессе тестирования совместимости веб-приложения «Rocket.Chat» проводятся следующие проверки:

1. Проверка работы приложения на разных операционных системах: приложение должно работать стабильно и без ошибок на различных операционных системах, таких как Windows, macOS и Linux.

2. Проверка работы приложения в разных браузерах: приложение должно корректно отображаться и функционировать в различных браузерах, таких как Chrome, Firefox, Safari, Internet Explorer и других. Важно проверить, что все функции приложения работают одинаково хорошо во всех поддерживаемых браузерах.

3. Проверка работы приложения на разных устройствах: приложение должно быть адаптивным и корректно отображаться на различных устройствах с разными размерами экранов, таких как настольные компьютеры, ноутбуки, планшеты и мобильные телефоны. Важно проверить, что все элементы интерфейса приложения правильно отображаются и доступны для пользователей на всех устройствах.

4. Проверка совместимости с различными версиями программного обеспечения: приложение должно работать с разными версиями операционных систем, браузеров и другого программного обеспечения. Важно проверить, что приложение не зависит от конкретных версий программного обеспечения и может работать с любыми поддерживаемыми версиями.

5. Проверка работы с различными разрешениями экранов: приложение должно корректно отображаться на различных разрешениях экранов, чтобы пользователи с разными устройствами и настройками экрана могли использовать приложение без проблем.

Тестирование совместимости веб-приложения помогает убедиться, что приложение доступно и функционирует правильно для всех пользователей, независимо от их операционной системы, браузера или устройства. Это важный аспект тестирования веб-приложений для обеспечения качества и удовлетворения потребностей всех пользователей.

**2.6 Тестирование безопасности**

Тестирование безопасности веб-приложения является одним из важных этапов в процессе разработки и обеспечения безопасности приложения. Веб-приложения подвержены различным угрозам, таким как атаки на сетевой уровень, уязвимости в коде приложения, несанкционированный доступ к данным пользователей и другие.

В процессе тестирования безопасности веб-приложения «Rocket.Chat» проводятся следующие проверки:

1. Проверка наличия уязвимостей в коде приложения: проводится анализ кода приложения с использованием специализированных инструментов и методов, чтобы выявить потенциальные уязвимости, такие как SQL-инъекции, XSS-атаки, CSRF-атаки и другие.

2. Проверка аутентификации и авторизации: проводится проверка механизмов аутентификации и авторизации приложения, чтобы убедиться в их надежности и защите от несанкционированного доступа к данным пользователей.

3. Проверка защиты от атак на сетевой уровень: проводится проверка наличия защиты от атак, таких как DDoS-атаки, перехват данных, подделка запросов и другие.

4. Проверка защиты данных: проводится проверка механизмов шифрования данных, хранения паролей и других конфиденциальных данных пользователей, чтобы убедиться в их надежности и защите от несанкционированного доступа.

5. Проверка безопасности сетевых протоколов: проводится проверка безопасности протоколов, используемых приложением для обмена данными с сервером или другими системами, чтобы убедиться в защите от атак на протоколы.

Тестирование безопасности веб-приложения помогает выявить уязвимости и проблемы безопасности приложения, а также предотвратить возможные атаки и потенциальные угрозы. Это важный аспект тестирования веб-приложений для обеспечения безопасности данных пользователей и защиты от потенциальных угроз.

# Тестируемые модули

## Профиль

Модуль «Профиль» позволяет произвести настройку личного профиля пользователя и выполнить такие действия как:

* просмотр профиля;
* редактирование профиля;
* изменение темы веб-приложения;
* изменение статуса;
* выход из профиля.

## Каналы

Модуль «Каналы» позволяет настроить различные каналы и возможные взаимодействия в них:

* просмотр списка каналов;
* создание канала;
* создание команды;
* создание личной переписки;
* создание обсуждения;
* добавление элемента списка каналов в избранное;
* присоединение к каналам;
* просмотр участников;
* поиск сообщений;
* поиск файлов;
* просмотр избранных сообщений;
* экспорт сообщений;
* удаление сообщений;
* просмотр внешнего вида списка чатов;
* изменение внешнего вида списка чатов;
* создание звонка;
* создание видеоконференции;
* просмотр информации о чате;
* настройки уведомлений;
* загрузка файлов;
* создание треда.

## Панель администрирования

Модуль «Панель администрирования» предоставляет широкий выбор настроек для управления рабочим пространством и его кастомизации и включает в себя следующие разделы:

* руководство по пробному периоду

Позволяет совершать такие действия как: просмотр и изменение пользовательских стилей CSS, просмотр и изменение пользовательских скриптов, просмотр и изменение настроек домашней страницы, просмотр и изменение настроек авторизации, просмотр и изменение настроек пользовательского интерфейса;

* информация

Позволяет получить общую информацию о версии рабочего пространства, актуальной лицензии и ее возможностях, о статистике использования рабочего пространства, также позволяет включить и настроить опцию «Федерация», которая позволяет неограниченному количеству рабочих пространств взаимодействовать друг с другом;

* консоль модерации (moderation console)

Позволяет просматривать и искать изменения в рабочем пространстве;

* импорт

Позволяет просматривать список импортированных файлов, импортировать файлы типа CSV, HipChat, Slack, пользователи Slack;

* пользователи

Позволяет просматривать список пользователей рабочего пространства, совершать поиск пользователей, добавление пользователя, приглашение нового пользователя;

* комнаты

Позволяет просматривать список комнат рабочего пространства, совершать поиск комнат, поиск комнат по фильтрам;

* приглашения

Позволяет просматривать список входящих и исходящих приглашений рабочего пространства;

* регистрация

Позволяет настроить всплывающие уведомления на мобильных устройствах, подключенных к рабочему пространству, установить магазин Rocket.Chat Marketplace, настроить взаимодействие рабочего пространства с социальными сетями, настроить параметры входа в рабочее пространство с помощью сторонних приложений;

* просмотр логов

Позволяет просматривать логии рабочего пространства;

* пользовательские звуки

Позволяет просматривать список пользовательских звуков, совершать прослушивание пользовательского звука из списка, добавление нового пользовательского звука;

* панель федерации (federation dashboard)

Позволяет включить и настроить опцию «Федерация», которая позволяет неограниченному количеству рабочих пространств взаимодействовать друг с другом;

* почтовые ящики

Позволяет просматривать список почтовых ящиков, совершать добавление нового почтового ящика;

* пользовательские эмодзи

Позволяет просматривать список пользовательских эмодзи, совершать добавление нового пользовательского эмодзи, удаление пользовательского эмодзи;

* интеграции

Позволяет просматривать список входящих интеграций рабочего пространства, список исходящих интеграций рабочего пространства, создавать новые интеграции с рабочим пространством;

* приложения OAuth

Позволяет настроить методы аутентификации помимо имени пользователя и пароля, а также просматривать список приложений OAuth, добавлять новое приложение OAuth;

* отправка почты

Позволяет настроить конфигурации для отправки широковещательных писем рабочего пространства;

* пользовательский статус (user status)

Позволяет просматривать список пользовательских статусов, добавлять новый статус пользователя;

* права доступа

Позволяет совершать просмотр и изменение правд доступа участников различных ролей, создавать новую роль;

* настройки

Позволяет гибко и глубоко настроить взаимодействие с рабочим пространством и включает в себя следующие подразделы такие как : активы, аналитика, боты, Видеоконференция, Внешний вид, Выгрузка пользовательских данных, Загрузка файлов, Интеграция WebDav, Настройки Omnichannel, Поддержка IRC, SlackBridge, Протокол LDAP, OAuth, SAML разметка, Rate Limiter, Push-уведомления, Smarsh и т.д.;

* управление устройствами

Позволяет просматривать список авторизованных в рабочем пространстве устройств, удалять авторизованные устройства;

* панель взаимодействия (engagement dashboard)

Позволяет совершать просмотр панели взаимодействия, в том числе аналитики по пользователям рабочего пространства, по сообщениям рабочего пространства, по каналам рабочего пространства, совершать экспорт аналитических данных.

**3.4 Модуль "Взаимодействия с другими пользователями Чата"**

Отвечает за:  
- Эмблема логина, Имя пользователя ,Роль (Admin или user), Местное время, Статус   
- Смотреть полный профиль - Личная переписка - Звонок(с Видео)

## Домашняя страница

Модуль «Домашняя страница» предоставляет быстрый и удобный доступ к основным функциям Rocket.Chat, таким как:

* добавление пользователей в пространство;
* создание каналов;
* просмотр каталога рабочего пространства;
* просмотр документации о Rocket.Chat;
* загрузка мобильного приложения;
* загрузка приложения для компьютера.

# 

# Окружение для работы. Описание операционных систем, версии приложения и браузеров

**Версия Приложения Rocket.Chat:**

**Версия:** 6.2.8

**Индикатор развертывания:** 6499

**Версия движка приложения:** v14.21.3

**Миграция базы данных:** 294(27 июня 2023 12:52)

**MongoDB:** 4.4.16/wifedTiger (oplog Включено)

**Подробности:** HEAD:(27753d300)

Ветка HEAD

Ид процесса 1

**Операционная система Пк:**

**Выпуск:** Windows 11

**Версия:** 21H2

**Сборка Ос**: 22000.1696

**Процессор:**

12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz

**Мобильный телефон:**

Samsung Galaxy A72,

**версия Android** -13,

**версия**  One UI 5.1

**Браузеры:**

1. **Firefox**  
- Версия 115.0.1 (64-разрядный)

2. **Google Chrome**  
- Версия 114.0.5735.199 (64 бит)

3. **Яндекс Браузер**

- Версия 23.7.0.2534 (64-bit)

# Виды тестовой документации, которые будут составляться в процессе тестирования

1. **Стратегия тестирования**  **(Test strategy)**:

Это статический документ высокого уровня, обычно разрабатываемый менеджером проекта. Это документ, который отражает подход к тестированию продукта и достижению целей, и дает четкое представление о том, что команда тестирования будет делать для всего проекта

**2**. **Test plan**:

Это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения

**3**. **Use Cases, сценарий использования:**

Это часть такого документа. Он описывает, какие действия выполняет пользователь и как система должна на них реагировать

**4. Test case**:

Это форма записи проверки, которую проводит тестировщик. По сути, это алгоритм действий, по которому предполагается тестировать уже написанную программу. В нём подробно прописаны шаги, которые нужно сделать для подготовки к тесту, сама проверка и ожидаемый результат

**5**. **Тест-Дизайн**

**Тестирование граничных значений**

Это техника проверки поведения продукта на крайних (граничных) значениях входных данных. Граничное тестирование также может включать тесты, проверяющие поведение системы на входных данных, выходящих за допустимый диапазон значений

**Таблица принятия решений**

Другое название метода – матрица принятия решений. Эта техника подходит для более сложных систем, например – двухфакторной аутентификации. Предположим, чтобы войти в систему, пользователю нужно ввести сначала логин и пароль, а затем еще подтвердить свою личность присланным в смс кодом

**Попарное тестирование**

Суть этого метода, также известного как pairwise testing, в том, что каждое значение каждого проверяемого параметра должно быть протестировано на взаимодействие с каждым значением всех остальных параметров. После составления такой матрицы мы убираем тесты, которые дублируют друг друга, оставляя максимальное покрытие при минимальном необходимом наборе сценариев.

**6.** **Баг-репорт (bug report)** :

это технический документ, который подробно описывает ошибку в работе программы, приложения или другого ПО. Его составляет, тестировщик чтобы разработчикам было понятно, что работает неправильно, насколько дефект критичен и что нужно исправить.

**7.** **Отчёт о тестировании** **(test-report):**

это отчет о выполнении тест-кейсов, в нем обычно отмечается общая статистика, количество выполненных тест-кейсов и количество найденных ошибок

# План тестирования согласно заданию

Приложение «Rocket.Chat» находится на стадии эксплуатации, значит все уровни тестирования уже пройдены. В проекте будет проводиться только системное функциональное тестирование.

Системное тестирование ПО — это тестирование программного обеспечения, выполняемое на полной, интегрированной системе, с целью проверки соответствия системы исходным требованиям. Системное тестирование проводится методом чёрного ящика, и, следовательно, не требует знаний о внутреннем устройстве системы.

Системное тестирование веб-приложения «Rocket.Chat» будет произведено на базе сценариев использования (use case based) по следующему плану:

1. На основе представления о способах использования приложения конечным пользователем создаются сценарии использования системы (Use Cases).

2. По конкретному сценарию использования составляются один или более тест-кейсов. Они заводятся в TestIT.

3. В TestIT из составленных тест-кейсов создаётся тест-план и проводится функциональное системное тестирование.

4. Результаты тестирования выгружаются в таблицу в формате xlsx, анализируются.

5. Описываются найденные баги (составляется баг-репорт).

6. Формируется отчет.

# Когда тестирование можно будет считать завершённым?

# Тестирование считается завершенным когда:

# - Достигнут дедлайн;

- Все критические ошибки исправлены и протестированы;

- Все 100% требований учтены;

- Достигнуто полное покрытие функционала;

- Дефекты установлены/ожидаемое число дефектов обнаружено;

**-** Все дефекты исправлены;

**-** Бюджет исчерпан;

- Вся документация по тестированию, подлежащая сдаче (отчет о тестировании),

подготовлена, проверена и передана.

# Время проведения тестирования

*Таблица 1. Время проведения тестирования*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **Создание тестовых сценариев** | **Создание тест-кейсов** | **Проведение тестирования** | **Фиксация результатов тестирования** | **Составление отчета** |
|  |  | Время, мин | Время, мин | Время, мин | Время, мин | Время, мин |
| 1 | Функциональное тестирование | 600 | 600 | 300 | 120 | 60 |
| 2 | UI-тестирование | 600 | 600 | 300 | 120 | 60 |
| 3 | UX-тестирование | 600 | 600 | 300 | 120 | 60 |
| 4 | Тестирование производительности | 600 | 600 | 300 | 120 | 60 |
| 5 | Тестирование локализации | 300 | 300 | 150 | 60 | 60 |
| 6 | Кроссбраузерное тестирование | 300 | 300 | 150 | 60 | 60 |
| 7 | Тестирование безопасности | 600 | 600 | 300 | 120 | 60 |
|  |  | 3600 | 3600 | 1800 | 720 | 420 |

Следует выделить несколько основных критериев для завершения тестирования:

1. время – сроки, выделенные заказчиком на проведение тестирования;
2. бюджет – объем денежных средств, выделенных заказчиком на проведение тестирования;
3. все тест кейсы пройдены, найденные баги локализованы и описаны (см. таблица 2. Критерии оценки возможности завершения тестирования).

*Таблица 2. Критерии оценки возможности завершения тестирования*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Выполнение тест-кейсов** | **Поиск и локализация багов/дефектов** | **Составление рекомендаций по устранению дефектов** | **Продукт корректно выполняет свои функции** |
| 1 | Функциональное тестирование | ✔ | ✔ | ✔ | ✔ |
| 2 | UI-тестирование | ✔ | ✔ | ✔ | ✔ |
| 3 | UX-тестирование | ✔ | ✔ | ✔ | ✔ |
| 4 | Тестирование производительности | ✔ | ✔ | ✔ | ✔ |
| 5 | Тестирование локализации | ✔ | ✔ | ✔ | ✔ |
| 6 | Кроссбраузерное тестирование | ✔ | ✔ | ✔ | ✔ |
| 7 | Тестирование безопасности | ✔ | ✔ | ✔ | ✔ |

*Таблица 3. Период выполнения задания*

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **Дата начала** | **Дата завершения** |
| 1 | Составление тестовой стратегии | 21/07/2023 | 21/07/2023 |
| 2 | Составление тестовых сценариев | 22/07/2023 | 22/07/2023 |
| 3 | Составление тест-кейсов | 23/07/2023 | 23/07/2023 |
| 4 | Составление тест-плана | 24/07/2023 | 24/07/2023 |
| 5 | Проведение тестирования | 25/07/2023 | 25/07/2023 |
| 6 | Составление отчета о тестировании | 26/07/2023 | 26/07/2023 |
| 7 | Составление баг-репорта | 27/07/2023 | 27/07/2023 |