Тест-план

Версия 1.0

План тестирования страницы контактов пользователя

Содержание

Оглавление

[Введение 2](#_Toc71064249)

[1. Цель 2](#_Toc71064250)

[2. Идентификация объекта тестирования 2](#_Toc71064251)

[3. Список функций и компонентов системы для тестирования 2](#_Toc71064252)

[4. Стратегия тестирования 3](#_Toc71064253)

[5. Ресурсы 13](#_Toc71064254)

[5.1 Исполнители 13](#_Toc71064255)

[6. Системы 14](#_Toc71064256)

[7. Основные этапы проекта 14](#_Toc71064257)

[8. История изменений 14](#_Toc71064258)

# Введение

План тестирования предназначен для определения объема, стратегии, расписания, критериев начала и окончания тестирования страницы контактов пользователя.

В плане указываются элементы, которые должны быть протестированы, функции, которые должны быть протестированы, типы тестирования, которые необходимо выполнить, а также оценки рисков с вариантами их разрешения.

# Цель

Цель составления данного Тест Плана является описание процесса тестирования страницы контактов пользователя. Приоритезация задач по тестированию. Учет требуемых ресурсов (ПО, оборудование), необходимых для тестирования.

# Идентификация объекта тестирования

Объектом тестирования является страница контактов пользователя, на этой страницы вы можете изменять и удалять личные данные пользователя.

# Список функций и компонентов системы для тестирования

* Заменить фото
* Удалить
* Имя
* Фамилия
* Основной телефон
* Основная почта
* Skype
* Прочее
* Добавить контакт
* Сохранить
* Отменить

# Стратегия тестирования

Стратегия тестирования представляет собой рекомендуемый подход к тестированию продукта. Главными вопросами тестовой стратегии являются техники тестирования, которые будут использоваться, и критерий, по которому можно было бы определить, что тестирование завершено.

# Типы тестирования

# Функциональное тестирование

Функциональное тестирование - является одним из ключевых видов тестирования, задача которого – установить соответствие разработанного программного обеспечения (ПО) исходным функциональным требованиям заказчика. То есть проведение функционального тестирования позволяет проверить способность информационной системы в определенных условиях решать задачи, нужные пользователям.

|  |  |
| --- | --- |
| Цель тестирования | Обеспечьте надлежащую функциональность, включая навигацию, ввод, обработку и поиск данных. |
| Способы | Выполните каждый вариант использования, последовательность вариантов использования или функцию, используя валидные и не валидные данные, чтобы проверить следующее:   * Ожидаемые результаты достигаются при использовании валидных данных. * При использовании не валидных данных отображаются соответствующие сообщения об ошибках или предупреждениях. |
| Критерий завершения | * Все запланированные испытания выполнены. * Все выявленные дефекты устранены |
| Особые замечания | Определите или опишите те элементы или проблемы (внутренние или внешние), которые влияют на реализацию и выполнение функционального тестирования. |

# Тестирование пользовательского интерфейса

Тестирование интерфейса проверяет взаимодействие пользователя с программным обеспечением. GUI необходимо тестировать для того, чтобы удостовериться, что GUI предоставляет пользователю возможность передвижения и соответствующий доступ к функциям приложения. Кроме того, тестирование GUI подтверждает, что объекты внутри GUI функционируют, как предполагается и соответствуют корпоративным и промышленным стандартам.

|  |  |
| --- | --- |
| Цель тестирования | * Выявление ошибок функциональности * Выявление необработанных исключений при взаимодействии с интерфейсом * Выявление потери или искажения данных, передаваемых через элементы интерфейса * Выявление ошибки в интерфейсе (несоответствие проектной документации, отсутствие элементов интерфейса) |
| Способы | * Создайте или измените тесты для каждого окна, чтобы проверить правильность навигации и состояния объектов для каждого окна и объектов приложения. |
| Критерий завершения | Каждое окно успешно проверено на соответствие эталонной версии или приемлемому стандарту |
| Особые замечания | Доступны не все свойства пользовательских и сторонних объектов. |

* + 1. Тестирование производительности

Тестирование производительности оценивает время реагирования/ответа, скорость выполнения транзакции и другие временные критерии. Целью ТП является проверка и подтверждение полученных критериев производительности. ТП обычно проводится несколько раз, каждый раз в системе используется разная "загрузка в фоновом режиме". Начальный тест должен проводиться с "номинальной" загрузкой, подобной нормальной загрузке установленной (или прогнозируемой) для этой системы. Второй тест на производительность проводится с использованием максимальной загрузки.

|  |  |
| --- | --- |
| Цель тестирования | Подтвердите время отклика системы для обозначенных транзакций или бизнес-функций при следующих двух условиях:   * измерение времени выполнения выбранных операций при определенных интенсивностях выполнения этих операций * определение количества пользователей, одновременно работающих с приложением * определение границ приемлемой производительности при увеличении нагрузки (при увеличении интенсивности выполнения этих операций) |
| Способы | * Используйте процедуры тестирования, разработанные для тестирования бизнес-моделей (системного тестирования). * Измените файлы данных (для увеличения количества транзакций) или сценарии, чтобы увеличить количество итераций, выполняемых каждой транзакцией. * Сценарии следует запускать на одной машине (лучший случай для тестирования одного пользователя, одной транзакции) и повторять с несколькими клиентами (виртуальными или фактическими, |
| Критерий завершения | * Одна транзакция / один пользователь: успешное завершение тестовых сценариев без сбоев и в пределах ожидаемого / необходимого распределения времени (на транзакцию) * Несколько транзакций / несколько пользователей: успешное завершение тестовых сценариев без сбоев и в приемлемое время. |
| Особые замечания | Комплексное тестирование производительности включает наличие «фоновой» нагрузки на сервер. Для этого можно использовать несколько методов, в том числе:   * Управляйте транзакциями» напрямую на сервер, обычно в форме вызовов SQL. * Создайте «виртуальную» пользовательскую нагрузку для имитации множества (обычно нескольких сотен) клиентов. Для выполнения этой нагрузки используются инструменты эмуляции удаленного терминала. * Используйте несколько физических клиентов, каждый из которых выполняет тестовые сценарии, чтобы создать нагрузку на систему. * Тестирование производительности должно выполняться на выделенном компьютере или в определенное время. Это обеспечивает полный контроль и точное измерение. * Базы данных, используемые для тестирования производительности, должны быть либо фактического размера, либо одинаково масштабируемыми. |

* + 1. Стрессовое тестирование

Стрессовое тестирование - позволяет проверить насколько приложение и система в целом работоспособны в условиях стресса и также оценить способность системы к регенерации, т.е. к возвращению к нормальному состоянию после прекращения воздействия стресса.

|  |  |
| --- | --- |
| Цель тестирования | Убедитесь, что система и программное обеспечение работают правильно и без ошибок в следующих стрессовых условиях:   * мало или совсем нет памяти на сервере (RAM и DASD) * максимальное (фактическое или физически возможное) количество подключенных (или смоделированных) клиентов несколько пользователей, выполняющих одни и те же транзакции с одними и теми же данными / аккаунтами * несколько пользователей, выполняющих одни и те же транзакции с одними и теми же данными / аккаунтами * объем / сочетание транзакций в наихудшем случае. |
| Способы | * Используйте тесты, разработанные для тестирования производительности. * Чтобы протестировать ограниченные ресурсы, тесты должны выполняться на одной машине, RAM и DASD на сервере должны быть уменьшены (или ограничены). * Для оставшихся стресс-тестов следует использовать несколько клиентов, выполняющих одни и те же тесты или дополнительные тесты для получения наихудшего объема / сочетания транзакций. |
| Критерий завершения | Все запланированные тесты выполняются, и указанные системные пределы достигаются / превышаются без сбоя программного обеспечения или программного обеспечения (или условия, при которых происходит сбой системы, выходят за рамки указанных условий). |
| Особые замечания | * Нагрузка на сеть может потребовать сетевых инструментов для загрузки сети сообщениями / пакетами. * DASD, используемый для системы, следует временно уменьшить, чтобы ограничить доступное пространство для роста базы данных. |

* + 1. Тестирование безопасности и контроля доступа

Тестирование безопасности - это стратегия тестирования, используемая для проверки безопасности системы, а также для анализа рисков, связанных с обеспечением целостного подхода к защите приложения, атак хакеров, вирусов, несанкционированного доступа к конфиденциальным данным.

|  |  |
| --- | --- |
| Цель тестирования | * Безопасность функций / данных: убедитесь, что пользователь может получить доступ только к тем функциям / данным, для которых его типу пользователя предоставлены разрешения. * Безопасность системы: убедитесь, что доступ к ним разрешен только тем пользователям, которые имеют доступ к системе и приложениям. |
| Способы | * Безопасность функций / данных: Определите и перечислите каждый тип пользователя, а также функции / данные, для которых у каждого типа есть разрешения. * Создавайте тесты для каждого типа пользователей и проверяйте каждое разрешение, создавая транзакции, специфичные для каждого типа пользователя. * Измените тип пользователя и повторно запустите тесты для тех же пользователей. В каждом случае убедитесь, что эти дополнительные функции / данные правильно доступны или запрещены. |
| Критерий завершения | Для каждого известного типа пользователя доступна соответствующая функция / данные, и все транзакции функционируют должным образом и выполняются в предыдущих тестах функций приложения. |
| Особые замечания | Доступ к системе должен быть рассмотрен / обсужден с соответствующим сетевым или системным администратором. Это тестирование может не потребоваться, поскольку это может быть функция сетевого или системного администрирования. |

* + 1. Тестирование падения / восстановления

Тестирование падения/восстановления обеспечивает такие условия, при которых приложение или вся система может "упасть" и восстановиться при большом количестве железных, программных или сетевых отказов/неправильных срабатываний с большой потерей или сохранением данных.

Тестирование на падение обеспечивает такие условия, для систем, которые должны продолжать работать, когда возникают условия для падения, при которых системы дублеры или резервные системы должным образом "замещают" упавшие системы без потери данных или транзакций.

Тестирование на восстановление является противоположным процессом, при котором приложение или система открыты для экстремальных условий (или смоделированных условий), таких как ошибка ввода/вывода или неверные указатели/ключи в базе данных. Включается процесс восстановления и приложение/система просматривается и/или проверяется, чтобы проверить корректность восстановления приложения/системы/данных.

|  |  |
| --- | --- |
| Цель тестирования | Убедитесь, что процессы восстановления (ручные или автоматические) правильно восстанавливают базу данных, приложения и систему до желаемого известного состояния. В тестирование должны быть включены следующие типы условий:   * Отключение питания клиента * Прерывание питания сервера * Прерывание связи через сетевой сервер * Прерывание, связь или потеря питания DASD и / или контроллера (ов) DASD * Неполные циклы (процессы фильтрации данных прерваны, процессы синхронизации данных прерваны). * Неверный указатель / ключи базы данных * Неверный / поврежденный элемент данных в базе данных |
| Способы | Тесты, созданные для тестирования функций приложения и бизнес-цикла, следует использовать для создания серии транзакций. Как только желаемая начальная точка тестирования будет достигнута, следующие действия должны быть выполнены (или смоделированы) индивидуально:   * Прерывание питания клиента: выключить ПК * Прерывание питания клиента: выключить телефон * Прерывание питания сервера: имитируйте или инициируйте процедуры отключения питания для сервера * Прерывание, связь или потеря питания для DASD и / или контроллера (ов) DASD: имитируйте или физически устраняйте связь с одним или несколькими контроллерами или устройствами DASD.   После достижения вышеуказанных условий / смоделированных условий следует выполнить дополнительные транзакции, а по достижении этого состояния второй точки тестирования следует вызвать процедуры восстановления. При тестировании на неполные циклы используется тот же метод, что и описанный выше, за исключением того, что сами процессы базы данных должны быть прерваны или преждевременно завершены. Для тестирования следующих условий требуется достижение известного состояния базы данных. Некоторые поля базы данных, указатели и ключи должны быть повреждены вручную и непосредственно в базе данных (с помощью инструментов базы данных). |
| Критерий завершения | Во всех вышеупомянутых случаях приложение, база данных и система должны после завершения процедур восстановления вернуться в известное желаемое состояние. Это состояние включает повреждение данных, ограниченное известными поврежденными полями, указателями / ключами и отчетами, указывающими на процессы или транзакции, которые не были завершены из-за прерываний. |
| Особые замечания | * Тестирование восстановления очень навязчиво. Процедуры отключения кабеля (имитация потери питания или связи) могут быть нежелательными или выполнимыми. Могут потребоваться альтернативные методы, такие как диагностические программные средства. * Требуются ресурсы из групп «Системы» (или «Операции с компьютером»), «База данных» и «Сеть». * Эти тесты следует проводить в нерабочее время или на изолированном компьютере |

* + 1. Инсталляционное тестирование

Инсталляционное тестирование преследует две цели. Первая - убедиться, что программное обеспечение может быть установлено на всех возможных конфигурациях, таких как новая инсталляция, модернизация и полная или обычная инсталляция, а также при нормальных и отклоняющихся от нормы условиях. Отклоняющиеся от нормы условия включают в себя недостаток места на диске, отсутствие разрешения на создание директорий и т.д. Вторая - проверить, что однажды установленное программное обеспечение работает корректно. Обычно это предполагает проведение большого количества тестов, которые были разработаны для функционального тестирования.

|  |  |
| --- | --- |
| Цель тестирования | Убедитесь, что клиентское программное обеспечение проекта правильно установлено на каждом клиенте при следующих условиях:   * Новая установка, новый компьютер, ранее никогда не устанавливавшийся с «Страница контактов пользователя» * Обновить ранее установленную машину «Страница контактов пользователя», та же версия * Обновить ранее установленную машину «Страница контактов пользователя», старая версия |
| Способы | * Вручную или разработайте автоматизированные сценарии для проверки состояния целевой машины (новый – «Страница контактов пользователя» никогда не устанавливался, «Страница контактов пользователя» та же версия или более старая версия уже установлена). * Запустить / выполнить установку. * Используя предопределенный набор скриптов интеграции или тестирования системы, запустите транзакции. |
| Критерий завершения | Транзакции «Страница контактов пользователя» выполняются успешно и без сбоев. |
| Особые замечания | Какие транзакции «Страница контактов пользователя» должны быть выбраны для проверки достоверности того, что приложение «Страница контактов пользователя» было успешно установлено и ни один из основных компонентов программного обеспечения не пропал? |

# Ресурсы

Этот раздел представляет рекомендуемые для проекта «Страница контактов пользователя» ресурсы, их главные обязанности, знания и навыки.

# 5.1 Исполнители

|  |  |  |
| --- | --- | --- |
| Человеческие ресурсы | | |
| Должность | Минимальные рекомендуемые ресурсы | Конкретные обязанности / комментарии |
| Менеджер по тестированию / руководитель тестового проекта | ­- | Обеспечивает надзор со стороны руководства Обязанности:   * Обеспечить техническое руководство * Получать соответствующие ресурсы * Управлять отчетностью |
| Тест дизайнер | - | Определяет, расставляет приоритеты и реализует тестовые примеры Обязанности:   * Создать план тестирования * Создать тестовую модель * Оценить эффективность тестирования |
| Тестер | - | Выполняет тесты Обязанности:   * Провести тесты * Журнал результатов * Восстановление после ошибок * Задокументировать дефекты |
| Системный администратор тестирования | - | Обеспечивает управление и обслуживание тестовой среды и активов. Обязанности:   * Администрирование системы управления тестированием * Установка / управление доступом сотрудников к тестовым системам |
| Администрирование баз данных / Менеджер баз данных | - | Обеспечивает управление и обслуживание среды и активов тестовых данных (базы данных). Обязанности:   * Администрирование тестовых данных (база данных) |

# Системы

В следующей таблице представлены системные ресурсы для проекта тестирования.

|  |  |
| --- | --- |
| Системные ресурсы | |
| Ресурс | Имя / Тип |
| Сервер базы данных | - |
| Сеть или подсеть | Будет определено |
| Имя сервера | Будет определено |
| Имя базы данных | Будет определено |
| Клиентские тестовые ПК | Будет определено |
| требования | Будет определено |
| Репозиторий тестов SQA | - |
| ПК для разработки тестов | Будет определено |

# Основные этапы проекта

В таблице должно содержаться описание трудоемкости для каждого объема работ, указанного в предыдущих секциях. Для связи статуса проекта и трудоемкости, необходимо определить отдельные этапы тестирования.

|  |  |  |  |
| --- | --- | --- | --- |
| Контрольная задача | Попытка | Дата начала | Дата окончания |
| Планирование тестирования | - | - | - |
| Дизайн теста | - | - | - |
| Разработка тестов | - | - | - |
| Выполнение теста | - | - | - |
| Оценка теста | - | - | - |

# История изменений

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Версия | Дата утверждения | Статус |
| - | 1.0 | - | - |