**Santa Editor Pages menu**

**Test Plan**

**Version <1.**4**>**

\***Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 22/03/2015 | <1.4> | Initial version | Kovalenko Dmitriy |
| 04/09/2021 | <1.4> | Draft created | Safina Dinara |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. Introduction

1.1 Purpose

1.2 Background

1.3 Scope

1.4 Project Identification

2. Requirements for Test

3. Test Strategy

3.1 Testing Types

3.1.1 Data and Database Integrity Testing

3.1.2 Function Testing

3.1.3 Business Cycle Testing

3.1.4 User Interface Testing

3.1.5 Performance Profiling

3.1.6 Load Testing

3.1.7 Stress Testing

3.1.8 Volume Testing

3.1.9 Security and Access Control Testing

3.1.10 Failover and Recovery Testing

3.1.11 Configuration Testing

3.1.12 Installation Testing

3.2 Tools

4. Resources

4.1 Roles

4.2 System

5. Project Milestones

6. Deliverables

6.1 Test Model

6.2 Test Logs

6.3 Defect Reports

Appendix A Project Tasks

**Test Plan**

# **Introduction**

## **Purpose**

Целью настоящего тест плана является описание процесса тестирования фичи “Pages Panel”, которая позвооляет настраивать страницы в WIX HTML5 Editor. . Данный документ позволяет получить представление о плановых работах, сроках и стратегиях тестирования. В данном документе не предполагается описание текст кейсов, ссылки на найденные дефекты, а так же их анализ. В данном документе приведены чек лист Функционала feature Pages Panel, а также чек- лист безопасности и контроля доступа.

## **Background**

Перед вами меню управлением страницами сайта. Все изменения должны отображаются в структуре сайта и в объекте самого меню. Предназначено для создания/удаления страниц, управления структурой страниц и доступами. Также содержит все необходимые настройки для комфортного управления.

**Pages use story**

*Чтобы организовать страницы:*

*1. Нажмите на иконку Страниц.*

*2. Перетащите страницу в нужное место, чтобы изменить порядок страниц.*

*3. Нажмите на иконку настроек страницы.*

*4. Введите название, скройте, настройте все для SEO, дублируйте или удалите страницу. Нажмите Готово.*

*Вы сразу увидите изменения.*

*Чтобы создать суб-страницу, наведите курсор на левую часть страницы в списке и перетащите страницу вправо, чтобы превратить ее в суб-страницу. Если ваше меню сразу не превратится в выпадающее, то вам придется удалить старое меню и добавить новое меню на ваш сайт. Имейте в виду, что вы сможете увид****еть как работает ваше выпадающее меню только в режиме Предпросмотра*.**

[**Полный документ use cases**](https://docs.google.com/spreadsheets/d/1q0zTu_5RPQWvb3a8k-pB5sVuu7ByIkpbKn8lRwjJuEw/edit?usp=sharing)

## **Scope**

## Целью тестирования проекта является проверка всех его функциональных возможностей на различных версиях браузеров, при различных разрешениях монитора, а также проведение серии стресс-тестов для выявления узких мест и уязвимостей проекта.

## Итоговыми документами процесса тестирования будут являться: отчет о результатах тестирования, включающий в себя описание тестовых сред и найденных дефектов и недочетов. Заключение тестировщиков об общем состоянии проекта, представляющим собой график соотношения критических дефектов к общему их числу.

## Тестирование предполагается вести в ручном режиме, без использования автоматизированных систем.

## 

## **Project Identification**

The table below identifies the documentation and availability used for developing the *test plan*:

*[Note: Delete or add items as appropriate.]*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document  (and version / date)** | **Created or Available** | **Received or Reviewed** | **Author or Resource** | **Notes** |
| Requirements Specification | **□ Yes**  □ No | **□ Yes**  □ No |  |  |
| Functional Specification | **□ Yes** □ No | **□ Yes** □ No |  |  |
| Use-Case Reports | **□ Yes** □ No | **□ Yes** □ No |  |  |
| Project Plan | **□ Yes** □ No | **□ Yes**  □ No |  |  |
| Design Specifications | **□ Yes** □ No | **□ Yes** □ No |  |  |
| Prototype | **□ Yes** □ No | **□ Yes**  □ No |  |  |
| User’s Manuals | □ Yes  **□ No** | □ Yes  **□ No** |  |  |
| Business Model or Flow | □ Yes **□ No** | □ Yes **□ No** |  |  |
| Data Model or Flow | □ Yes **□ No** | □ Yes **□ No** |  |  |
| Business Functions and Rules | □ Yes  **□ No** | □ Yes  **□ No** |  |  |
| Project or Business Risk Assessment | □ Yes **□ No** | □ Yes **□ No** |  |  |

## 

## 1.4 Team members

|  |  |
| --- | --- |
| **Role** | **Resource Name** |
| Developer |  |
| Product Manager | Kovalenko Dmytro |
| QA | Safina Dinara |

# 1.5 Dependencies

|  |  |
| --- | --- |
| **Module** | **Developer** |
| Менеджер фоновых изображений |  |
| Page transitions |  |
| Возможность добавления страницы |  |
| Page settings |  |
| Поиск по странице |  |
| Возможность редактирования страницы(rename/duplicate/hide) |  |
| Application Pages |  |

# \*1.6 Related Documents

Cпецификация - <https://drive.google.com/file/d/0B04XE92Y7D6UejNsbU1fdTBHb00/view?usp=sharing>

USE cases - <https://docs.google.com/spreadsheets/d/1q0zTu_5RPQWvb3a8k-pB5sVuu7ByIkpbKn8lRwjJuEw/edit?usp=drive_web>

прототип - <https://c1x0jy.axshare.com/final.html>

# 1.7 **Requirements for** Pages

The listing below identifies those items⎯use cases, functional requirements, and non-functional requirements⎯that have been identified as targets for testing. This list represents what will be tested.

***Основные функциональные елементы:***

* *Анимированное перетаскивание страниц и групп*
* *Возможность свернуть/развернуть под-страницы*
* *При нажатии на страницу происходит переход*
* *Функция поиска страниц по названию*
* *Действия на страницами : под страница, скрыть, отобразить, дублировать, переименовать*
* *Добавление/удаление действующих элементы: страница, под-страница, меню страниц, ссылка (6 видов)*
* *Анимированный переход между страницами*

***Меню настроек страниц***

* *Основные настройки*
* *SEO*
* *Landing page (без футера и хеадера)*
* *Защита страницы: пароль, регистрация*

*Не функциональные требования:*

* *Возможность сохранить новые страницы*
* *Возможность опубликовать новые страницы*
* *Время сохранения до 5 с*
* *Время публикации до 3 с*

# **Test Strategy**

Стратегия процесса тестирования планируется в три этапа:

Первый этап заключается в анализе ТЗ, составлении критического чек-листа, составлении тест плана, а также частичного прогона функциональных тестов.

Второй этап будет посвящен детализации функционального чек-листа и детальному прогону функциональных тестов с выявлением и описанием дефектов.

На третьем этапе будет произведено стресс-тестирование и нагрузочное тестирование с описанием найденных дефектов или демонстрацией соответствия системным требованиям.

Таким образом, достигается максимальная детализация глубины тестирования, что, в свою очередь, позволяет более точно определить затрачиваемые ресурсы, а также позволяет разработчикам проекта начать исправлять дефекты на самых ранних этапах.

Ввиду отказа от ведения дефектов в багтрекере, все обнаруженные дефекты будут передаваться менеджерам проекта в письменном виде через баг листы.

На первом этапе будет применен смоук тестинг, при котором будут уточняться требования, определяться и конфигурироваться тестовые среды.

К началу второго этапа будут сформированы критический чек-лист, а так же чек-лист по функциональному тестированию и юзер стори.

На втором этапе производится детальное тестирование функционала проекта, собираются и описываются дефекты. Каждый чек-лист прогоняется для каждого браузера.

Третий этап завершает работы по тестированию. В нем проводится установленный набор тестов для выявления уязвимостей. Такой вид тестирования довольно затратный по времени, поэтому необходимый набор тест кейсов разрабатывается совместно с разработчиками проекта.

## **Testing Types**

### *Data and Database Integrity Testing*

*[The databases and the database processes should be tested as a subsystem within the <Project* ***Name>****. These subsystems should be tested without the target-of-test’s User Interface as the interface to the data. Additional research into the DataBase Management System (DBMS) needs to be performed to identify the tools and techniques that may exist to support the testing identified below.]*

|  |  |
| --- | --- |
| Test Objective: | *[Ensure database access methods and processes function properly and without data corruption.]* |
| Technique: | *∙ [Invoke each database access method and process, seeding each with valid and invalid data or requests for data.*  *∙ Inspect the database to ensure the data has been populated as intended, all database events occurred properly, or review the returned data to ensure that the correct data was retrieved for the correct reasons]* |
| Completion Criteria: | *[All database access methods and processes function as designed and without any data corruption.]* |
| Special Considerations: | *∙ [Testing may require a DBMS development environment or drivers to enter or modify data directly in the databases.*  *∙ Processes should be invoked manually.*  *∙ Small or minimally sized databases (limited number of records) should be used to increase the visibility of any non-acceptable events.]* |

***Cross Browser Testing (Тестирование в определенной среде)***

Цель: Проверить корректную работу и дизайн Проекта в различных браузерах и при различных разрешениях монитора

**Описание конфигураций**

|  |  |  |  |
| --- | --- | --- | --- |
| Браузер | Version | OS | Screen Resolution |
| Chrome | latest | win/macOS | 1024 x 768, 1280 x 720 |
| FireFox | latest | linux/macOS/win |  |
| IE | 11,10,9,8 |  |  |
| Safari | latest | macOS |  |

### *Function Testing*

Цель: Выявление функциональных ошибок, несоответствий ТЗ и ожиданиям пользователя путем реализации стандартных, а также нетривиальных тестовых сценариев.

|  |  |
| --- | --- |
| Test Objective: | Функционал feature Pages Panel |
| Technique: | **Выполняя каждый use case или функцию используя действительные и недействительные данные, проверьте следующее:**   1. **Общее**  * Уменьшение/увеличение масштаба страницы (элементы должны соответственно перераспределиться с сохранением пропорций) * Проверить верстку страницы * кликнуть на значок крестика в правом верхнем углу панели, проверить, что кнопка кликабельна и происходит закрытие панели  1. **Панель настройки страницы**  * проверить кликабельность кнопки settings напротив названия страницы в меню страницы * проверить кликабельность кнопки в виде синего треугольника справа от settings * проверить возможность сворачивания/разворачивания разделов (черные треуольники) в Pages panel * проверить возможность удаления страницы * проверить возможность редактирования названия страницы * проверить возможность удаления страницы * проверить возможность редактирования URL- адреса страницы * проверить возможность скрыть страницу из меню сайта * в Page settings проверить кликабельность разделов(Page info, layout ...) * в каждом разделе проверить кликабельность кнопок, текстовых полей, чекбокса, радиокнопок. * проверить, что возможен выбор только одной из радиокнопок * проверить возможность изменения конфиденциальности страницы * редактироват настройки страницы SEO, проверить, что настройи сохранились * проверить возможность изменения стиля страницы * проверить возможность изменения стиля страницы * проверить возможность изменения фона страницы  1. **Добавление новой страницы**  * кликнуть на кнопку Add Page, проверить, что появляется окно с возможнотью добавления новой страницы * проверить, что можно ввести название новой страницы * проверить, что можно выбрать слой (Select layout) * выбрать тип добавляемой страницы, проверить что можно добавить новую страницу * навести курсор на треугольник справа от кнопки add page, проверить, что появляется возмоность добавить категорию меню и меню ссылок * нажать на крестик в правом верхнем углу окошка добавления страницы, проверить, что страница закрылась  1. **Переходы между страницами**  * кликнуть на иконку с бабочкой, проверить, что можно выбрать переходы страницы (Page transitions) * проверить, что можно закрыть Page transitions, нажав на крестик в правом верхнем углу окошка  1. **Поиск по странице**  * проверить, что можно осуществить поиск по странице, введя поисковое слово в строке поиска внизу Panel pages * проверить, что можно закрыть панел поиска, нажав на крестик в правом верхнем углу  1. **Application pages**  * проверить кликабельность кнопок manages * кликнуть на кнопку manages, проверить, что в появивишемся окошке все кнопки кликабельны * Проверить кликабельность иконки и содержание информации в меню “Help” * проверить, что можно закрыть окошко с помощью иконки “close” |
| Completion Criteria: | Чек лист считается завершенным когда будет покрыто тестами 100% функционала |
| Special Considerations: | *[Identify or describe those items or issues (internal or external) that impact the implementation and execution of function test]* |

### *Business Cycle Testing*

*[Business Cycle Testing should emulate the activities performed on the <Project Name> over time. A period should be identified, such as one year, and transactions and activities that would occur during a year’s period should be executed. This includes all daily, weekly, and monthly cycles and, events that are date-sensitive, such as ticklers.]*

|  |  |
| --- | --- |
| Test Objective | *[Ensure proper target-of-test and background processes function according to required business models and schedules.]* |
| Technique: | *[Testing will simulate several business cycles by performing the following:*  *∙ The tests used for target-of-test’s function testing will be modified or enhanced to increase the number of times each function is executed to simulate several different users over a specified period.*  *∙ All time or date-sensitive functions will be executed using valid and invalid dates or time periods.*  *∙ All functions that occur on a periodic schedule will be executed or launched at the appropriate time.*  *∙ Testing will include using valid and invalid data to verify the following:*  *∙ The expected results occur when valid data is used.*  *∙ The appropriate error or warning messages are displayed when invalid data is used.*  *∙ Each business rule is properly applied.* |
| Completion Criteria: | *∙ [All planned tests have been executed.*  *∙ All identified defects have been addressed.}* |
| Special Considerations: | *∙ [System dates and events may require special support activities*  *∙ Business model is required to identify appropriate test requirements and procedures.]* |

### *User Interface Testing*

*[User Interface (UI) testing verifies a user’s interaction with the software. The goal of UI testing is to ensure that the User Interface provides the user with the appropriate access and navigation through the functions of the target-of-test. In addition, UI testing ensures that the objects within the UI function as expected and conform to corporate or industry standards.]*

|  |  |
| --- | --- |
| Test Objective: | В разделе Usability исследуется удобство использования функционала, качество GUI (графического интерфейса), орфографические ошибки, и прочие не функциональные параметры |
| Technique: | * Проверка GUI   + Соответствие дизайну мокапов   + Соответствие корпоративному стилю   + Выравнивание текста   + Шрифты   + Масштабирование окон * Проверка орфографии * Наличие подсказок и понятность функционала * Производительность (в разрезе удобства использования, например горячие комбинации клавиш) |
| Completion Criteria: | Личное чувство удоволетворенности удобством проекта |
| Special Considerations: |  |

### *Performance Profiling*

*[Performance profiling is a performance test in which response times, transaction rates, and other time-sensitive requirements are measured and evaluated. The goal of Performance Profiling is to verify performance requirements have been achieved. Performance profiling is implemented and executed to profile and tune a target-of-test's performance behaviors as a function of conditions such as workload or hardware configurations.*

*Note: Transactions below refer to “logical business transactions”. These transactions are defined as specific use cases that an actor of the system is expected to perform using the target-of-test, such as add or modify a given contract.]*

|  |  |
| --- | --- |
| Test Objective: | *[Verify performance behaviors for designated transactions or business functions under the following conditions:*  *∙ normal anticipated workload*  *∙ anticipated worst case workload]* |
| Technique: | *∙ [Use Test Procedures developed for Function or Business Cycle Testing.*  *∙ Modify data files to increase the number of transactions or the scripts to increase the number of iterations each transaction occurs.*  *∙ Scripts should be run on one machine (best case to benchmark single user, single transaction) and be repeated with multiple clients (virtual or actual, see Special Considerations below).]* |
| Completion Criteria: | *∙ [Single Transaction or single user: Successful completion of the test scripts without any failures and within the expected or required time allocation per transaction.]*  *∙ [Multiple transactions or multiple users: Successful completion of the test scripts without any failures and within acceptable time allocation.]* |
| Special Considerations: | *[Comprehensive performance testing includes having a background workload on the server.*  *There are several methods that can be used to perform this, including:*  *∙ “Drive transactions” directly to the server, usually in the form of Structured Query Language (SQL) calls.*  *∙ Create “virtual” user load to simulate many clients, usually several hundred. Remote Terminal Emulation tools are used to accomplish this load. This technique can also be used to load the network with “traffic”.*  *∙ Use multiple physical clients, each running test scripts to place a load on the system.*  *Performance testing should be performed on a dedicated machine or at a dedicated time. This permits full control and accurate measurement.*  *The databases used for Performance Testing should be either actual size or scaled equally.]* |

### *Load Testing*

Нагрузочное тестирование - тест производительности, который подвергает цель теста различной нагрузке для измерения и оценки поведения и эффективности и способности продолжать успешно функционировать должным образом при различных рабочих и экстремальных нагрузках. Цель нагрузочного тестирования, определить точки выхода системы из равновесия и гарантировать, что система функционирует должным образом за пределами ожидаемой максимальной рабочей нагрузки. Кроме того, нагрузочное тестирование оценивает характеристики, такие как время отклика, время транзакции, и других операций чувствительных к времени.

|  |  |
| --- | --- |
| Test Objective: | Целью тестирования является измерения времени операций, транзакций, времени отклика системы |
| Technique: | Процесс тестирование: генерировать псевдо пользователей, которые выполняют сценарии user story, выполняют бизнес нагрузку.  В целом задача генерировать указанную в ТЗ пиковую нагрузку с помощью средств автоматизации тестирования |
| Completion Criteria: | Выполнения всех нагрузок согласно ТЗ |
| Special Considerations: |  |

### *Stress Testing*

*[Stress testing is a type of performance test implemented and executed to find errors due to low resources or competition for resources. Low memory or disk space may reveal defects in the target-of-test that aren't apparent under normal conditions. Other defects might result from competition for shared resources like database locks or network bandwidth. Stress testing can also be used to identify the peak workload the target-of-test can handle.]*

*[Note: References to transactions below refer to logical business transactions.]*

|  |  |
| --- | --- |
| Test Objective: | *[Verify that the target-of-test functions properly and without error under the following stress conditions:*  *∙ little or no memory available on the server (RAM and DASD)*  *∙ maximum actual or physically capable number of clients connected or simulated*  *∙ multiple users performing the same transactions against the same data or accounts*  *∙ worst case transaction volume or mix (see Performance Testing above).*  *Notes: The goal of Stress Testing might also be stated as identify and document the conditions under which the system FAILS to continue functioning properly.*  *Stress Testing of the client is described under section 3.1.11, Configuration Testing.]* |
| Technique: | *∙ [Use tests developed for Performance Profiling or Load Testing.*  *∙ To test limited resources, tests should be run on a single machine, and RAM and DASD on server should be reduced or limited.*  *∙ For remaining stress tests, multiple clients should be used, either running the same tests or complementary tests to produce the worst-case transaction volume or mix.* |
| Completion Criteria: | *[All planned tests are executed and specified system limits are reached or exceeded without the software failing or conditions under which system failure occurs is outside of the specified conditions.]* |
| Special Considerations: | *∙ [Stressing the network may require network tools to load the network with messages or packets.*  *∙ The DASD used for the system should temporarily be reduced to restrict the available space for the database to grow.*  *∙ Synchronization of the simultaneous clients accessing of the same records or data accounts.]* |

### *Volume Testing*

*[Volume Testing subjects the target-of-test to large amounts of data to determine if limits are reached that cause the software to fail. Volume Testing also identifies the continuous maximum load or volume the target-of-test can handle for a given period. For example, if the target-of-test is processing a set of database records to generate a report, a Volume Test would use a large test database and check that the software behaved normally and produced the correct report.]*

|  |  |
| --- | --- |
| Test Objective: | *[Verify that the target-of-test successfully functions under the following high volume scenarios:*  *∙ Maximum (actual or physically- capable) number of clients connected, or simulated, all performing the same, worst case (performance) business function for an extended period.*  *∙ Maximum database size has been reached (actual or scaled) and multiple queries or report transactions are executed simultaneously.]* |
| Technique: | *∙ [Use tests developed for Performance Profiling or Load Testing.*  *∙ Multiple clients should be used, either running the same tests or complementary tests to produce the worst-case transaction volume or mix (see Stress Testing above) for an extended period.*  *∙ Maximum database size is created (actual, scaled, or filled with representative data) and multiple clients used to run queries and report transactions simultaneously for extended periods.]* |
| Completion Criteria: | *∙ [All planned tests have been executed and specified system limits are reached or exceeded without the software or software failing.]* |
| Special Considerations: | *[What period of time would be considered an acceptable time for high volume conditions, as noted above?]* |

### *Security and Access Control Testing*

Тестирование безопасности и контроля доступа сосредотачивается на двух ключевых областях:

* ∙безопасности на уровне приложений, в том числе доступ к данным или бизнес-функциям
* ∙системный уровень безопасности, в том числе возможность входа или удаленного доступа к системе.
* защита от взлома хакерскими атаками

Безопасность на уровне приложений гарантирует, что, основываясь на желаемой безопасности, пользователи ограничены специфическими функциями или границами, или ограничены в данных, которые доступны для них. Например, каждому может быть разрешено вводить данные и создавать новые учетные записи, но только менеджеры могут удалять их.

|  |  |
| --- | --- |
| Test Objective: | * Application-level Security: уровни доступа к данным позльзователей * System-level Security: ограничение возможностей пользователей * Защита от взлома |
| Technique: | * Application-level Security: проверить доступ к страницам редактора для не зарегистрированных и зарегистрированных пользователей * System-level Access: возможность изменения уровня доступа только у владельца и даминистратора сайта. Не возможность изменения данных простым пользователем * Защита от взома   + XSS   + SQL injection   + XSRF   + Data corruption |
| Completion Criteria: | Считается завершенным после прохождения всего чек листа |
| Special Considerations: |  |

### *Failover and Recovery Testing*

*[Failover and Recovery Testing ensures that the target-of-test can successfully failover and recover from a variety of hardware, software or network malfunctions with undue loss of data or data integrity.*

*Failover testing ensures that, for those systems that must be kept running, when a failover condition occurs, the alternate or backup systems properly “take over” for the failed system without loss of data or transactions.*

*Recovery testing is an antagonistic test process in which the application or system is exposed to extreme conditions, or simulated conditions, to cause a failure, such as device Input/Output (I/O) failures or invalid database pointers and keys. Recovery processes are invoked and the application or system is monitored and inspected to verify proper application, or system, and data recovery has been achieved.]*

|  |  |
| --- | --- |
| Test Objective: | *[Verify that recovery processes (manual or automated) properly restore the database, applications, and system to a desired, known, state. The following types of conditions are to be included in the testing:*  *∙ power interruption to the client*  *∙ power interruption to the server*  *∙ communication interruption via network servers*  *∙ interruption, communication, or power loss to DASD and or DASD controllers*  *∙ incomplete cycles (data filter processes interrupted, data synchronization processes interrupted).*  *∙ invalid database pointer or keys*  *∙ invalid or corrupted data element in database]* |
| Technique: | *[Tests created for Function and Business Cycle testing should be used to create a series of transactions. Once the desired starting test point is reached, the following actions should be performed, or simulated, individually:*  *∙ Power interruption to the client: power the PC down.*  *∙ Power interruption to the server: simulate or initiate power down procedures for the server.*  *∙ Interruption via network servers: simulate or initiate communication loss with the network (physically disconnect communication wires or power down network servers or routers.*  *∙ Interruption, communication, or power loss to DASD and DASD controllers: simulate or physically eliminate communication with one or more DASD controllers or devices.*  *Once the above conditions or simulated conditions are achieved, additional transactions should be executed and upon reaching this second test point state, recovery procedures should be invoked.*  *Testing for incomplete cycles utilizes the same technique as described above except that the database processes themselves should be aborted or prematurely terminated.*  *Testing for the following conditions requires that a known database state be achieved. Several database fields, pointers, and keys should be corrupted manually and directly within the database (via database tools). Additional transactions should be executed using the tests from Application Function and Business Cycle Testing and full cycles executed.]* |
| Completion Criteria: | *[In all cases above, the application, database, and system should, upon completion of recovery procedures, return to a known, desirable state. This state includes data corruption limited to the known corrupted fields, pointers or keys, and reports indicating the processes or transactions that were not completed due to interruptions.]* |
| Special Considerations: | *∙ [Recovery testing is highly intrusive. Procedures to disconnect cabling (simulating power or communication loss) may not be desirable or feasible. Alternative methods, such as diagnostic software tools may be required.*  *∙ Resources from the Systems (or Computer Operations), Database, and Networking groups are required.*  *∙ These tests should be run after hours or on an isolated machine.]* |

### *Configuration Testing*

*[Configuration testing verifies the operation of the target-of-test on different software and hardware configurations. In most production environments, the particular hardware specifications for the client workstations, network connections and database servers vary. Client workstations may have different software loaded⎯for example, applications, drivers, and so on⎯and at any one time, many different combinations may be active using different resources.]*

|  |  |
| --- | --- |
| Test Objective: | Проверить корректную работу и дизайн Проекта в различных браузерах и при различных разрешениях монитора |
| Technique: | 1680\*1050/Internet Explorer 11  1024 x 768, 1280 x 720/ Chrome |
| Completion Criteria: |  |
| Special Considerations: | *∙ [What non-target-of-test software is needed, is available, and is accessible on the desktop?*  *∙ What applications are typically used?*  *∙ What data are the applications running; for example, a large spreadsheet opened in Excel or a 100- page document in Word?*  *∙ The entire systems, netware, network servers, databases, and so on also needs to be documented as part of this test.]* |

### *Installation Testing*

*[Installation testing has two purposes. The first is to insure that the software can be installed under different conditions⎯such as a new installation, an upgrade, and a complete or custom installation⎯under normal and abnormal conditions. Abnormal conditions include insufficient disk space, lack of privilege to create directories, and so on. The second purpose is to verify that, once installed, the software operates correctly. This usually means running a number of the tests that were developed for Function Testing.]*

|  |  |
| --- | --- |
| Test Objective: | *Verify that the target-of-test properly installs onto each required hardware configuration under the following conditions:*  *∙ new installation, a new machine, never installed previously with <Project Name>*  *∙ update, machine previously installed <Project Name>, same version*  *∙ update, machine previously installed <Project Name>, older version* |
| Technique: | *∙ [Manually or develop automated scripts, to validate the condition of the target machine⎯ new - <Project Name> never installed; <Project Name> same version or older version already installed).*  *∙ Launch or perform installation.*  *∙ Using a predetermined sub-set of function test scripts, run the transactions.]* |
| Completion Criteria: | *<Project Name> transactions execute successfully without failure.* |
| Special Considerations: | *[What <Project Name> transactions should be selected to comprise a confidence test that <Project Name> application has been successfully installed and no major software components are missing?]* |

## **Tools**

The following tools will be employed for this project:

*[Note: Delete or add items as appropriate.]*

|  |  |  |  |
| --- | --- | --- | --- |
|  | Tool | Vendor/In-house | Version |
| Test Management | Google docs |  |  |
| Defect Tracking | joxy, kazam |  |  |
| ASQ Tool for functional testing |  |  |  |
| ASQ Tool for performance testing |  |  |  |
| Test Coverage Monitor or Profiler |  |  |  |
| Project Management |  |  |  |
| DBMS tools |  |  |  |

# **Resources**

*[This section presents the recommended resources for the <Project Name> project, their main responsibilities, and their knowledge or skill set.]*

## **Roles**

This table shows the staffing assumptions for the project.

*[NOTE: Delete or add items as appropriate.]*

|  |  |  |
| --- | --- | --- |
| Human Resources | | |
| Worker | Minimum Resources Recommended  (number of full-time roles allocated) | Specific Responsibilities or Comments |
| Test Manager,  Test Project Manager |  | Provides management oversight.  Responsibilities:   * provide technical direction * acquire appropriate resources * provide management reporting |
| Test Designer |  | Identifies, prioritizes, and implements test cases.  Responsibilities:   * generate test plan * generate test model * evaluate effectiveness of test effort |
| Tester |  | Executes the tests.  Responsibilities:   * execute tests * log results * recover from errors * document change requests |
| Test System Administrator |  | Ensures test environment and assets are managed and maintained.  Responsibilities:   * administer test management system * install and manage access to test systems |
| Database Administrator, Database Manager |  | Ensures test data (database) environment and assets are managed and maintained.  Responsibilities:   * administer test data (database) |
| Designer |  | Identifies and defines the operations, attributes, and associations of the test classes.  Responsibilities:   * identifies and defines the test classes * identifies and defines the test packages |
| Implementer |  | Implements and unit tests the test classes and test packages.  Responsibilities:   * creates the test classes and packages implemented in the test model |

## **System**

The following table sets forth the system resources for the testing project.

*[The specific elements of the test system are not fully known at this time. It is recommended that the system simulate the production environment, scaling down the accesses and database sizes if and where appropriate.]*

*[Note: Delete or add items as appropriate.]*

|  |  |
| --- | --- |
| System Resources | |
| Resource | Name / Type |
| Database Server | wix.com |
| —Network or Subnet | TBD |
| —Server Name | TBD |
| —Database Name | TBD |
| Client Test PC's |  |
| —Include special configuration requirements | TBD |
| Test Repository |  |
| —Network or Subnet | TBD |
| —Server Name | TBD |
| Test Development PC's | TBD |

# **Project Milestones**

*[Testing of <Project Name> should incorporate test activities for each of the test efforts identified in the previous sections. Separate project milestones should be identified to communicate project status accomplishments.]*

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone Task** | **Effort** | **Start Date** | **End Date** |
| Plan Test |  |  |  |
| Design Test |  |  |  |
| Implement Test |  |  |  |
| Execute Test |  |  |  |
| Evaluate Test |  |  |  |

# **Deliverables**

*[In this section, list the various documents, tools, and reports that will be created, by whom, delivered to who, and when delivered.]*

## **Test Model**

*[This section identifies the reports that will be created and distributed from the test model. These artifacts in the test model need to be created or referenced in the ASQ tools.]*

## **Test Logs**

*[Describe the method and tools used to record and report on the test results and testing status.]*

## **Defect Reports**

*[In this section, identify the method and tools used to record, track, and report on test incidents and their status.]*

# **Appendix A Project Tasks**

Below are the test-related tasks:

∙ Plan Test

* + identify requirements for test
  + assess risk
  + develop test strategy
  + identify test resources
  + create schedule
  + generate Test Plan

∙ Design Test

- prepare workload analysis

- identify and describe test cases

- identify and structure test procedures

- review and assess test coverage

∙ Implement Test

* + record or program test scripts
  + identify test-specific functionality in the Design and Implementation Model
  + establish external data sets

∙ Execute Test

- execute Test procedures

- evaluate execution of Test

- recover from halted Test

- verify the results

- investigate unexpected results

- log defects

∙ Evaluate Test

- evaluate Test-case coverage

- evaluate code coverage

- analyze defects

- determine if Test Completion Criteria and Success Criteria have been achieved