< **Межконтинентальный конвейер средств индивидуального перевоплощения «Иллюзия»**>

Test Plan

Version <0.2>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | Author |
| <15/06/14> | <0.1> | Создание документа | Грудина А.М. |
| <18/06/14> | <0.2> | Добавление информации | Грудина А.М. |

Table of Contents

1. Introduction 5

1.1 Purpose 5

1.2 Scope 5

1.3 Intended Audience 5

1.4 Document Terminology and Acronyms 5

1.5 References 5

2. Evaluation Mission and Test Motivation 6

2.1 Background 6

2.2 Evaluation Mission 6

2.3 Test Motivators 6

3. Target Test Items 7

4. Outline of Planned Tests 8

4.1 Outline of Test Inclusions 8

5. Test Approach 9

5.1 Testing Techniques and Types 9

5.1.1 Data and Database Integrity Testing 9

5.1.2 Function Testing 9

5.1.3 User Interface Testing 10

5.1.4 Performance Profiling 10

5.1.5 Stress Testing 11

5.1.6 Security and Access Control Testing 12

5.1.7 Configuration Testing 12

6. Entry and Exit Criteria 13

6.1 Test Plan 13

6.1.1 Test Plan Entry Criteria 13

6.1.2 Test Plan Exit Criteria 13

6.1.3 Suspension and Resumption Criteria 13

7. Deliverables 13

7.1 Test Evaluation Summaries 13

7.2 Incident Logs and Change Requests 13

8. Environmental Needs 14

8.1 Base Software Elements in the Test Environment 14

8.2 Productivity and Support Tools 14

9. Responsibilities, Staffing, and Training Needs 15

9.1 People and Roles 15

10. Risks, Dependencies, Assumptions, and Constraints 16

11. Management Process and Procedures 16

11.1 Measuring and Assessing the Extent of Testing 16

11.2 Approval and Signoff 16

**Test Plan <Master>**

# Introduction

## Purpose

Цель тест плана – сбор всей необходимой информации для планирования и контроля над тестовой активностью для отдельно взятой итерации тестирования. Он описывает подход к тестированию программного обеспечения и является планом верхнего уровня, сгенерированным и используемым менеджерами для управления тестовой активностью.

Документ Test Plan для проекта «Иллюзия» обеспечивает достижение следующих целей:

* Постановка целевых объектов тестирования;
* Поиск достаточного тестового покрытия;
* Описание используемых подходов к тестированию;
* Определение обязательных ресурсов и оценка времени на тестовую активность;
* Список элементов тестового окружения.

## Scope

Данный документ описывает уровни тестирования и типы тестирования, применимые на каждом из них. Также документ определяет значимые области, которые могут быть исключены из общего описания, так как их наличие предполагается другими областями.

## Intended Audience

Данный документ написан для команды тестирования, чтобы понимать нагрузку на команду и процесс тестирования приложения, и для менеджера проекта, который должен иметь общее представление о загрузке команды тестирования. Также данный документ является значимым для заказчиков проекта, как гарантия качества выпускаемого продукта.

## Document Terminology and Acronyms

Смотреть [Glossary].

## References

[Glossary] глоссарий проекта

[SRS] спецификация системных требований

[SAD] документ, описывающий архитектуру приложения

[Use case] варианты использования системы

# Evaluation Mission and Test Motivation

## Background

Целью для тестирования программного приложения является обеспечение требуемого уровня качества продукта и выявление как можно большего числа критических ошибок и ошибок, которые ведут к невозможности использования продукта в целом.

Запланированная тестовая активность является необходимой для проверки работоспособности системы с точки зрения функциональных и нефункциональных требований, описанных в [SRS] и [SAD].

## Evaluation Mission

Целью тестирования на данной итерации является:

* Найти, как можно больше ошибок;
* Выявить критические ошибки в работе системы;
* Проверить обеспечение функциональных и нефункциональных требований, описанных в [SRS];
* Убедить заказчиков в качестве продукта.

## Test Motivators

Мотивацией для тестирования является проверка функциональных и нефункциональных требований и вариантов использования системы, описанных в [Use case].

# Target Test Items

Целевыми объектами тестирования являются:

* Взаимодействие приложения с базой данных;
* Функциональные требования, представленные в приложении, как уровень бизнес-логики;
* Пользовательский интерфейс и его взаимодействие со всем приложением;
* Работа приложения в условиях нагрузки;
* Работа приложения на различных платформах (различное тестовое окружение);
* Безопасность и политика разграничения прав пользователей.

# Outline of Planned Tests

## Outline of Test Inclusions

Процесс тестирования включает в себя следующие типы тестирования:

* Data and Database Integrity Testing:
  + Проверка наличия доступа к базе данных со стороны приложения;
  + Проверка корректности сохранения, обновления и удаления данных;
  + Проверка корректности обновления и удаления для связанных данных.
* Function Testing:
  + Проверка всех ключевых вариантов использования, описанных в [Use Cases].
* User Interface Testing:
  + Проверка корректности отображаемых данных;
  + Проверка возможности навигации между всеми страницами приложения при различных вариантах использования.
* Performance Profiling:
  + Измерение среднее время обращения к базе данных для выборки данных и при сохранении различных данных в систему.
  + Измерение максимально возможное количество транзакций в секунду;
  + Измерение среднего времени при навигации между двумя соседними страницами приложения.
* Stress Testing:
  + Проверка работы системы в условиях параллельной (одновременной) работы максимального числа пользователей (20);
* Security and Access Control Testing:
  + Проверка невозможности пользователей системы обратиться к функциям, недоступным для их ролей.
* Configuration Testing:
  + Проверка работы системы на различных платформах (заявленных операционных системах и браузерах):
    - Windows 7/8, Chrome (latest);
    - Solaris OS 10, Chronium 25.

# Test Approach

Данный раздел представляет рекомендованные стратегии для разработки и выполнения обязательных тестов. Предыдущий раздел устанавливал, какие объекты будут протестированы и какие типы тестов будут проведены. Этот раздел описывает, как эти тесты будут реализованы.

Тестовые подходы описываются техниками, которые будут внедрены.

## Testing Techniques and Types

### Data and Database Integrity Testing

База данных и процессы в ней должны быть протестированы, как независимая система. Данный тип тестирования должен проверить подсистему без использования пользовательского интерфейса, как интерфейса для работы с данными, то есть он необходим для проверки самой базы данных и интеграции с ней приложения.

|  |  |
| --- | --- |
| Цель тестирования: | Проверить, что методы доступа к базе данных и процессы функционируют должным образом и без каких-либо повреждений данных. |
| Техника тестирования: | Выполнить проверку всех SQL-команд приложения, каждый из которых будет представлять отдельный тестовый метод, включенный в тот или иной тестовый класс.  Классы выделяются исходя из таблиц базы данных и специфических проверок: Connection (проверка соединение с базой данных с различными правами доступа к БД), Order (проверка создания/изменения/удаление заказа) и т.д. [SAD] |
| Источники: | SQL-команды приложения, выполняющие соответствующие действия с базой данных. |
| Инструменты тестирования: | JUnit 4 |
| Критерий успешности: | Подключение к базе данных работает;  Данные записываются в базу данных без потерь;  Данные считываются из базы данных без ошибок и искажений;  При удалении данных из базы остальные (связанные) данные не искажаются. |
| Специальные условия: | Тестирование проводится автоматизировано.  Тестирование выполняется на отдельной (чистой) базе данных небольшого размера. |

### Function Testing

Данный тип тестирования должен быть нацелен на проверку функциональных требований и бизнес-функций приложения. Функциональные требования описаны в [SRS]. Целью проведения таких тестов является проверка приема, обработки и извлечения данных и надлежащее исполнение бизнес-функций. Этот тип тестирования основан на методе черного ящика, то есть проверка приложения и его внутренних процессов производится через пользовательский интерфейс (GUI) и последующий анализ результатов выполнения тех или иных функций и отображения данных.

|  |  |
| --- | --- |
| Цель тестирования: | Проверить систему на соответствие заявленным функциональным требованиям. |
| Техника: | Выполнение каждого варианта использования поочередно и пошагово с использованием валидных и не валидных данных.  Перед началом прохождения варианты использования должны быть описаны соответствующими тестовыми сценариями, сгруппированные по тестовым наборам, с целью проверить:   1. Что ожидаемый результат получен при использовании валидных данных; 2. Соответствующие сообщения об ошибках и предупреждения выводятся при использовании не валидных данных; 3. Каждое бизнес правило добавлено в систему. |
| Источники: | [Use cases] |
| Инструменты тестирования: | JUnit 4, Selenium 2.5.0 |
| Критерий успешности: | Все запланированные тесты выполняются.  Все найденные дефекты будут устраняться. |
| Специальные условия: | Нет |

### User Interface Testing

Данный тип тестирования проверяет взаимодействие пользователя с системой. Цель тестирования пользовательского интерфейса – удостовериться, что пользовательский интерфейс предоставляет пользователю соответствующий его роли доступ и возможность навигации между функциями целевого тестирования. Также тестирование пользовательского интерфейса подтверждает, что графические компоненты функционируют, как ожидалось и соответствуют все одному стандарту.

|  |  |
| --- | --- |
| Цель тестирования: | Проверить следующие возможности:   1. Навигация между целевыми объектами, относящимися к бизнес-функциям и требованиям. Проверка навигации включает в себя проверку перехода между страницами и полями различными способами (действия курсором мыши, ввод с клавиатуры, табуляция); 2. Объекты окна и их характеристики, такие как меню, размеры, позиция, состояние и фокус. |
| Техника: | Проверить каждую функциональную страницу приложения по вариантам использования и соответствующим им тестовым сценариям.  Каждый тестовый класс представляет собой одну функциональную страницу с методами, описывающими действия пользователя в качестве методов с проверками реакции интерфейса на эти действия. |
| Источники: | [Use cases], тестовые сценарии из функционального тестирования |
| Инструменты тестирования: | Selenium 2.5.0 |
| Критерий успешности: | Все элементы интерфейса отображаются и доступны для пользователя, как это ожидалось из вариантов использования.  Навигация между страницами и полями работает. |
| Специальные условия: | Нет |

### Performance Profiling

Данный тип тестирования проводится с целью проверки, что система удовлетворяет нефункциональным требованиям (производительности), которые были заявлены в [SRS]: возможность работы 10 пользователей в системе.

В соответствующих тестах измеряются и оцениваются время отклика, количество параллельных транзакций и другие требования, чувствительные ко времени.

|  |  |
| --- | --- |
| Цель тестирования: | Проверить поведение системы для предполагаемого числа транзакций и проверить заявленную производительность системы:   1. При нормальной ожидаемой нагрузке; 2. При наихудшем ожидаемом варианте нагрузки. |
| Техника: | Создать две ThreadGroup – для максимального числа пользователей и числа пользователей меньше максимального, и выполнить следующие запросы:   1. Авторизация в роли Горожанина; 2. Создание заказа; 3. Редактирование заказа; 4. Просмотр списка моих заказов 5. Просмотр галереи; 6. Выход из системы. |
| Источники: | Нет |
| Инструменты тестирования: | JMeter |
| Критерий успешности: | Система удовлетворяет заявленным нефункциональным требованиям (производительности). |
| Специальные условия: | База данных, используемая при тестировании, должна быть схожа с реальной базой приложения по размерам и характеристикам. |

### Stress Testing

Стресс-тестирование – это тип нагрузочного тестирования, выполняемый для определения максимальной выдерживаемой системой нагрузки.

|  |  |
| --- | --- |
| Цель тестирования: | Проверить поведение системы в стресс-условиях и выявить условия, при которых система откажет (не работают функциональные возможности системы). |
| Техника: | Использовать тесты, разработанные для нагрузочного тестирования с обязательным одновременным выполнением сценария – Создания заказа для максимального числа пользователей и последовательным увеличением для определения состояния, когда система откажет. |
| Источники: | Нет |
| Инструменты тестирования: | JMeter |
| Критерий успешности: | Система функционирует при загрузке, равной максимальной заявленной в нефункциональных требованиях. |
| Special Considerations: | Нет |

### Security and Access Control Testing

Данный тип тестирования нацелен на две ключевые области безопасности:

* Уровень безопасности приложения, включающий доступ к данным и бизнес-функциями для различных ролей;
* Уровень безопасности системы, включающий механизм авторизации.

|  |  |
| --- | --- |
| Цель тестирования: | Проверить, что пользователю системы доступны только те функции системы, которые установлены его ролью в системе. Проверить механизм авторизации и изменения отображения до и после ее прохождения. |
| Техника: | 1. Выполнить варианты использования, доступные разным пользователям от лица пользователей, которым они не должны быть доступны, включая Гостя. 2. Выполнить вход в систему через пользовательский интерфейс и зайти на недоступные для этой роли страницы через ввод адреса к этим страницам в адресную строку браузера. |
| Источники: | [Use cases], тестовые сценарии функционального тестирования, скрипты Selenium для UI тестирования |
| Инструменты тестирования: | JUnit 4 и Selenium |
| Критерий успешности: | Пользователи с различными ролями имеют доступ только к предназначенным для них функциям.  Проверено для всех ролей в системе. |
| Специальные условия: | Нет |

### Configuration Testing

Данный тип тестирования проверяет работу приложения на различных платформах (заявленные браузеры).

|  |  |
| --- | --- |
| Цель тестирования: | Проверить корректность работы приложения в заявленных браузерах и операционных системах. |
| Техника: | Выполнить все тесты пользовательского интерфейса на различных платформах:   1. Windows 7/8, Chrome (latest); 2. Solaris 25, Firefox (latest). |
| Источники: | Скрипты Selenium для UI тестирования |
| Инструменты тестирования: | 1. Selenium 2. Рабочие машины с установленными платформами. |
| Критерии успешности: | Приложение работает на заявленных платформах корректно. Различия в поведения отсутствуют или являются некритичными для выполнения бизнес-функций. |
| Специальные условия: | Машины для тестирования не должны иметь специфического программного обеспечения, связанного с разработкой программного обеспечения |

# Entry and Exit Criteria

## Test Plan

### Test Plan Entry Criteria

Завершена разработка системы (основные функции системы, которые необходимо проверить).

### Test Plan Exit Criteria

Выполнены все тесты в данной итерации тестирования.

*Заметка! При нахождении незначительных для продолжения тестирования ошибок в системе и их последующем исправлении – выполняется еще одна независимая итерация выполнения тестового плана.*

### Suspension and Resumption Criteria

Процесс тестирования откладывается (приостанавливается) при нахождении критических ошибок или ошибок, мешающих продолжению хода тестирования.

Отложенный процесс тестирования возобновляется после исправления ошибок командой разработки.

# Deliverables

В данном разделе описываются документы, которые будут созданы в процессе тестирования и использованы для заказчиков, как отчеты о тестировании.

## Test Evaluation Summaries

Представляет собой итоговый анализ результатов испытаний и ключевые метрики тестирования для анализа и оценки (обычно ключевые метрики устанавливаются заинтересованными сторонами). Кроме того, этот документ может содержать общее утверждение относительного качества и обеспечить рекомендации для будущих тестов.

## Incident Logs and Change Requests

Содержит список всех ошибок, найденных в ходе тестирования, с указанием текущего статуса этих ошибок и шагов для воспроизведения.

При исправлении или изменении поведения системы, затрагивающего ошибки из списка, необходимо писать документ Change Request.

# Environmental Needs

Этот раздел описывает технические и программные ресурсы, необходимые для выполнения тестового плана.

## Base Software Elements in the Test Environment

|  |  |  |
| --- | --- | --- |
| **Software Element Name** | **Version** | **Type and Other Notes** |
| Solaris OS | 10 | Operating System |
| Windows | 7/8 | Operating System |
| Chrome | latest | Internet Browser |
| Firefox | latest | Internet Browser |
| JVM | 1.7.1 | Virtual Machine for executing |
| Glassfish | 4.0 | Application Server |

## Productivity and Support Tools

Программное обеспечение необходимое для проведения тестирования:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tool Category or Type** | **Tool Brand Name** | **Vendor or In-house** | **Version** |
| Test Management | Microsoft Word | Microsoft | 2010+ |
| Defect Tracking | Microsoft Excel | Microsoft | - |
| ASQ Tool for functional testing | JUnit; IntelliJ IDEA | Java | 4; latest |
| ASQ Tool for performance testing | JMeter | Java | Latest |
| Project Management | Microsoft Word; Excel | Microsoft | 2010+ |
| DBMS tools | Oracle Database | Oracle | 11g |

# Responsibilities, Staffing, and Training Needs

Этот раздел показывает обязательные ресурсы для тестовой активности при выполнении этого тестового плана; основные обязанности и знания (умения).

## People and Roles

Таблица показывает персонал, необходимый для тестирования:

|  |  |  |
| --- | --- | --- |
| **Human Resources** | | |
| **Role** | **Minimum Resources Recommended  (number of full-time roles allocated)** | **Specific Responsbilities or Comments** |
| Test Manager | 1 | Обеспечивает управление процессом тестирования.  Обязанности:   1. Планирование и логистика; 2. Определение мотивации к началу тестирования; 3. Подбор ресурсов; 4. Отчеты; 5. Оценка эффективности. |
| Test Designer | 0,5 | Определяет технические подходы к разработке тестов.  Обязанности:   1. Определяет подходы; 2. Определяет архитектуру для автоматизации тестов; 3. Проверяет тестовые техники; 4. Определяет тестируемые элементы; 5. Структурирует тестовую разработку. |
| Tester | 1 | Разрабатывает и выполняет тесты.  Обязанности:   1. Разработать тесты и тестовые наборы; 2. Выполнить тестовые наборы; 3. Вести лог результатов; 4. Анализировать и перепроверять тестовые ошибки; 5. Документировать инциденты. |

# Risks, Dependencies, Assumptions, and Constraints

Список любых рисков, которые могут влиять на успешность выполнения тестового плана:

|  |  |  |
| --- | --- | --- |
| **Risk** | **Mitigation Strategy** | **Contingency  (Risk is realized)** |
| Тестовые сценарии содержат ошибки | Тестер должен создавать тестовые сценарии и давать их на проверку другому тестеру или аналитику для проверки. | * Переписывание тестового сценария; * Переписывание всех связанных тестовых сценариев или всего тестового набора; * Переписывание тестового плана и переоценка тестовой активности |
| Необходимость пересоздания базы данных при внесении обновлений | Оценка и учет такого риска в тестовом плане с самого начала | * Потеря тестовых данных и необходимость их воссоздания; * Переустановка базы данных или ее обновление. |
| Выход за рамки отведенного на тестирование времени | Планирование вспомогательных ресурсов для тестирования и промежуточная корректировка плана в соответствии с текущим состоянием тестирования | * Не проверена большая часть системы * Не найдены критические ошибки в непроверенных функциях * Установка системы заказчику с нерабочими функциями * Исправление ошибок на месте у заказчика * Потеря заказчика * Денежные взыскания при выявлении поставки нерабочего продукта |

# Management Process and Procedures

## Measuring and Assessing the Extent of Testing

Найденные во время тестирования ошибки должны быть сохранены в одном документе и переданы команде разработки для приоритезации работы над ними менеджером команды. По окончании работы документ должен быть возращен команде тестирования с измененными статусами.

## Approval and Signoff

При проведении тестирования должен вестись список тестов и статус их выполнения, которые в дальнейшем послужат базой для документа Test Evaluation Summaries. Списки должны быть доступны тестовому менеджеру.

Процесс тестирования должен быть утвержден заказчиком.