Справочник «Виды ремонтов»

Test Plan

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 21/07/2015 | 1.0 | Создал | Филонов Е.С. |
| 22/07/2015 | 1.0 | Редактирование | Филонов Е.С. |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Background 4

1.3 Scope 4

1.4 Project Identification 4

2. Requirements for Test 4

3. Test Strategy 4

3.1 Testing Types 5

3.1.1 Smoke Testing 5

3.1.2 Function Testing 5

3.1.3 User Interface Testing 6

4. Resources 6

5. Deliverables 6

Test Plan

# Introduction

## Purpose

Целью составления данного Тест Плана является описание процесса тестирования приложения «Справочник «Виды ремонтов». Документ позволяет получить представление о мероприятиях по тестированию проекта.

## Background

Справочник «Виды ремонтов» - приложение представляет собой справочник видов ремонта.

## Scope

Целью тестирования приложения «Справочник «Виды ремонтов» является проверка корректной работы его функциональностей, обнаружение ошибок в интерфейсе, проверка работоспособности приложения на различных платформах. Оценить способность системы правильно функционировать при большом количестве данных.

Итогом процесса тестирования должен стать отчет о результатах тестирования, включающий в себя описание найденных дефектов и недочетов.

Тестирование будет производиться вручную.

## Project Identification

Справочник «Виды ремонтов»*,* версия 1.0

# Requirements for Test

Приложение должно удовлетворить потребность пользователя в справочной информации по видам ремонта.

Для этого приложение должно позволять пользователю:

- создавать строки в справочнике;

- изменять строки в справочнике;

- сохранять новые и измененные строки в файл;

- удалять строки в справочнике;

- отменять произведенные изменения до последнего сохраненного состояния;

- экспортировать справочник в xml-файл.

# Test Strategy

В процессе тестирования приложения «Справочник «Виды ремонтов» будет применена стратегия тестирования «черным ящиком (серым тоже можно, не уверен)», ввиду отсутствия доступа к коду приложения. Тестирование приложения будет разделено на несколько этапов.

На первом этапе будет проведено дымовое тестирование для подтверждения того, что приложение стартует и выполняет основные функции.

На втором этапе, в ходе проведения функционального тестирования, планируется выявить ошибки, основанные на реализации стандартных, а также нетривиальных тестовых сценариев.

Третий этап – обнаружение ошибок в интерфейсе.

Четвертый этап – проведение нагрузочного тестирования, для оценки способности системы правильно функционировать при большом количестве данных.

## Testing Types

### **Smoke Testing**

**Goal:**

Убедиться, что приложение запускается и пользователь может использовать основной функционал программы в процессе реального использования.

**Process Description:**

1. Эмуляция использования приложения: запуск приложения, добавление нескольких строк, сохранение, проверка созданного файла, экспорт в файл, выход. (TestCase\_SmokeTesting#1.docx)
2. Эмуляция использования приложения: запуск приложения, удаление строк, сохранение, проверка файла, экспорт в файл, выход. (TestCase\_SmokeTesting#2.docx)

**Completion Criteria:**

Приложение запускается, в приложении работает основной функционал. Можно перейти к следующей стадии тестирования.

### **Function Testing**

**Goal:**

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

**Process Description:**

1. Добавление строк, сохранение, проверка сохраненных данных в файле. (TestCase\_FunctionTesting#1.docx)
2. Добавление строк, отмена произведенных изменений, проверка данных в файле. (TestCase\_FunctionTesting#2.docx)
3. Редактирование строк, сохранение произведенных изменений, проверка данных в файле. (TestCase\_FunctionTesting#3.docx)
4. Удаление строк, сохранение произведенных изменений, проверка данных в файле. (TestCase\_FunctionTesting#4.docx)
5. Удаление строк, сохранение произведенных изменений, добавление новых строк, сохранение, проверка данных в файле. (TestCase\_FunctionTesting#5.docx)
6. Добавить несколько одинаковых строк. Проверить уникальность. (TestCase\_FunctionTesting#6.docx)
7. Выгрузка справочника в XML-файл, проверка данных в файле. (TestCase\_FunctionTesting#7.docx)
8. Удаление файла с данными, запуск приложения, проверка данных в файле. (TestCase\_FunctionTesting#8.docx)
9. Изменение имени файла, запуск приложения, проверка данных в справочнике. (TestCase\_FunctionTesting#9.docx)
10. Изменение данных в файле, запуск приложения, проверка данных в справочнике. (TestCase\_FunctionTesting#10.docx)
11. Провести сортировку, проверить данные в справочнике, проверка данных в файле. (TestCase\_FunctionTesting#11.docx)
12. Переместить столбцы, проверить данные в справочнике, проверка данных в файле. (TestCase\_FunctionTesting#11.docx)

**Completion Criteria:**

Запланированные области протестированы, найденные ошибки зафиксированы.

### **User Interface Testing**

**Goal:**

Тестирование элементов приложения, выявление ошибок пользовательского интерфейса.

**Process Description:**

1. Ввод данных в поле «Идентификатор» более 5 символов, сохранить, проверить данные в файле. (TestCase\_User\_Interface\_Testing#1.docx)
2. Ввод данных в поле «Идентификатор» 5 символов, сохранить, проверить данные в файле. (TestCase\_User\_Interface\_Testing#2.docx)
3. Не вводить данные в поле «Идентификатор», сохранить, проверить данные в файле. (TestCase\_User\_Interface\_Testing#3.docx)
4. Ввод данных в поле «Название» более 50 символов, сохранить, проверить данные в файле. (TestCase\_User\_Interface\_Testing#4.docx)
5. Ввод символов в поле «№», проверить данные в файле. (TestCase\_User\_Interface\_Testing#5.docx)
6. Добавить строку, проверить подсветку строки, сохранить. (TestCase\_User\_Interface\_Testing#6.docx)
7. Отредактировать строку, проверить подсветку строки, сохранить. (TestCase\_User\_Interface\_Testing#7.docx)
8. Удалить строку, проверить подсветку строки, сохранить. (TestCase\_User\_Interface\_Testing#8.docx)

**Completion Criteria:**

Запланированные тесты проведены, найденные ошибки зафиксированы.

# Resources

|  |  |
| --- | --- |
| System Resources | |
| Resource | Name / Type |
| Client Test PC's | Test PC (Windows 7 x64 SP1, Intel(R) Core(TM) i3-3240 CPU @ 3.40GHz, 8Gb) |
| Test Repository | C:\testPlan\ |

# Deliverables

Конечным результатом проведения тестирования должен стать оформленный конечный результат процесса тестирования с описанными дефектами, а также рекомендациями по улучшению продукта.