Глава 1

В первой главе были рассмотрены теоретические аспекты процесса тестирования. Определено понятие тестирование программного продукта. Были выделены основные классификации видов тестирования:

1. По объекту тестирования
2. Тестирование, связанное с изменениями
3. По уровню тестирования
4. По исполнению кода
5. По субъекту тестирования
6. По позитивности сценария
7. По степени автоматизации

Так же были изучены различные методологии тестирования:

1) метод черного ящика;

2) метод белого ящика;

3) метод серого ящика.

Были изучены аспекты разработки и выполнения различных тест-кейсов, были проанализированы основные атрибуты, которые включает в себя любой тест-кейс. Были изучены требования, которые применяются при составлении тест-кейсов. Так же били описаны и проанализированные различные виды дефектов, которые могут присутствовать в любом ПО. Был разобран жизненный цикл любого дефекта.

Был проведен анализ популярных CI платформ таких как CircleCI, 1.3.3 Travis CI, 1.3.4 Jenkins. Были выявлены достоинства и недостатки данных систем непрерывной интеграции. Была представлена схема непрерывной интеграции при использовании Ci платформ.

Были проанализированы различные фреймворки тестирования xUnit, NUnit, Visual Studio Unit Testing Framework так же были выявлены достоинства и недостатки данных фреймворков.

Далее были изучены системы контроля версий такие как Subversion, Git, Mercurial были проанализированы их достоинства и недостатки, а также были проанализированы способы работы с ними.

После этого была рассмотрена система TestLink. Так как основная часть тестов для ручного тестирования на предприятии размещено в данной системе. Были изучены способы работы с данной системой ее достоинства и недостатки.

Глава 2

Во второй главе была проанализирована программа Alpha.Alarms.Именно для этой программы в дальнейшем планируется написание автоматизированных тестов. За основу автоматизированных тестов будут взяты тесты для ручного тестирования расположенные в системе TestLink. Так же были изучены различные аспекты работы программы Alpha.Alarms. Были изучены режимы работы данной программы. Ее функциональные возможности.

Так же были изучены аспекты автоматизации процесса тестирования. Были выявлены достоинства и недостатки автоматизированных тестов перед ручными тестами. Были выявлены критерии, при которых следует применять автоматизированные тесты, а не ручных. Были изучены уровни автоматизации тестирования и была описана архитектура автоматизированных тестов.

После анализа автоматизированного тестирования. Был проведен расчет экономической и временно рентабельности замены набора ручных скриптов на автоматизированное тестирование. Данные расчёты показали, что автоматизированное тестирование в несколько раз эффективней ручного тестирования.

Глава3

В третьей главе рассматривается финальная версия реализованной библиотеки для тестирования программного продукта Alpha.Alarms. Был подробно описан алгоритм запуска тестов. И их поведение в процессе выполнения. Была составлена Uml диаграмма последовательности, которая в графическом виде показывает жизненный цикл процесса выполнения автоматизированных тестов.

Были описаны различные группы автоматизированных тестов, для которых необходимо выполнять тестирование, а именно:

1. получения оперативных событий;
2. получения исторических событий;
3. проверки корректности отображения всех элементов;
4. загрузки различных конфигураций;
5. совместимости различных конфигураций;
6. применения различных фильтров в историческом режиме;
7. применения различных фильтров в оперативном режиме
8. использования различных некорректных значений;
9. сохранения настроек после перезагрузки программы;
10. имитации различных действий пользователя.

Так же были описаны два способа запуска автоматизированных тестов с помощью Microsoft Visual Studio и при сборке продукта с помощью Jenkins.Так же были описаны достоинства и недостатки каждого вида запуска.

Так же была произведена оценка тестового покрытия приложения Alpha.Alarms на текущий момент. Из 113 элементов доступных для работы пользователю на текущий момент проверяется 73 что составляет 64.5% общего функционала приложения Alpha.Alarms