

Android приложение состоит из двух видов тестов: текстового и графического. После прохождения теста пользователю будет дана характеристика психологического состояния. Кроме того, пользователь сможет просматривать результаты своих предыдущих тестов и результаты других пользователей.

Для создания приложения использован объектно-ориентированный язык программирования Java, с использованием IDE Eclipse и Android SDK.

Основной сложностью в работе является подбор необходимых материалов для тестов и разработка методик обработки картинок.

Результаты данной дипломной работы могут быть использованы в качестве экспресс-анализа психологического состояния человека.

Список использованных источников:

1. Цехнер М. Проектирование и разработка Android приложений // Уч. метод. пособие для студентов. – Минск, 2009. – 898 с.
2. Rogers R. Android Application Development Programming with the Google SDK / Lombardo J. // Минск, 2008. – 1060 с.

ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС ДЛЯ ПРОВЕДЕНИЯ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ СЕРВИСОВ ИНТЕРНЕТ-ПОЧТЫ

УО «Белорусский государственный университет информатики и радиоэлектроники»
г. Минск, Республика Беларусь

Ковалев Н.К.

Черимисинов Д.И. – доцент, к. т. н.

Целью работы является программно-аппаратный комплекс для проведения автоматизированного тестирования сервисов интернет-почты, который обусловлено необходимостью в возрастании информационной нагрузки на пользователя

Автоматизация позволяет проводить тестирование посредством программно-аппаратных средств. Применение автоматизации позволяет существенно сократить временные затраты на тестирование и уменьшить объемы обрабатываемой инженерами информации.

Для решения рассматриваемой проблемы разработано программное решение. Данное решение работает напрямую с веб-интерфейсом почтового сервиса, для того, чтобы протекающие автоматизированные процессы можно было наблюдать непосредственно, в настоящем времени.

Можно выделить следующие части разрабатываемого комплекса:

- Подсистема работы с веб-интерфейсом. Служит для предоставления эмуляции веб-браузера, а так же объектного описания тестируемого приложения и его компонентов. Определяет возможности по взаимодействию с тестируемой средой.
- Тестовые классы, описывающие процесс тестирования. Здесь описывается иерархия и структура проводимых тестов, их взаимозависимости, порядок выполнения, а так же их содержание.
- Инструменты для обработки тестовых данных. Тестовыми данными могут быть информация, необходимая для работы тестируемого приложения, средства локации определенных элементов на интерфейсе и многое другое. Они могут быть представлены в виде электронных таблиц, xml-файлов, баз данных и прочих информационных структур.
- Средства логирования процесса тестирования и его результатов. Предоставляют информацию о процессе тестирования с задаваемой точностью и подробностью. Подобная информация может быть представлена в виде текстовых логов, снимков экранов, файлов отчетов и т.д.

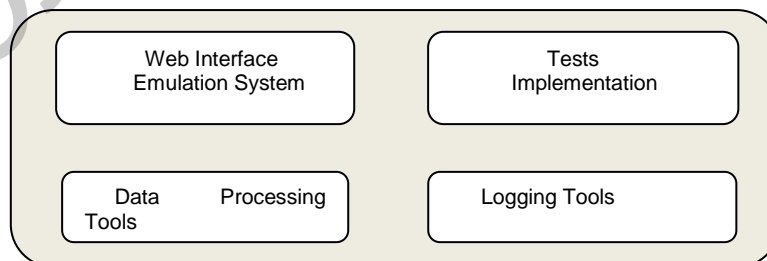


Рис 1 – Структурная схема приложения в первом приближении.

Дополнительно к разрабатываемому проекту можно подключить средства обеспечения непрерывной интеграции разработки, позволяющие проводить тестирование на удаленных компьютерах и на регулярной основе, а также средства распараллеливания процесса тестирования, которые позволяют выполнять группы тестов одновременно на нескольких компьютерах.

Основная цель разработанного продукта – предоставление инженеру-тестировщику фреймворка, обеспечивающего дополнительный слой абстракции при написании тестов, с возможностью расширения

данного решения и наличием дополнительных опций для повышения эффективности и упрощения работы инженера.

Реализован данный продукт на объектно-ориентированном языке Java с применением интегрированной среды разработки «Eclipse».

Список использованных источников:

1. Эккель Б. [Eckel B.] Философия Java. Библиотека программиста. 4-е изд. – СПб: Питер, 2012 – 640 с.
2. <http://testng.org/doc/documentation-main.html> - документация TestNG.

МНОГОФУНКЦИОНАЛЬНЫЙ СЕТЕВОЙ ФАЙЛОВЫЙ МЕНЕДЖЕР

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Кондратюк А. А.

Давыдовский А. Г. – доцент, к. биолог. н.

Целью работы является разработка многофункционального сетевого файлового менеджера. При создании приложений для облегчения работы с электронными документами основной задачей является предоставление пользователю инструмента для взаимодействия с файловой системой, важным аспектом является обеспечение сохранности данных.

Разработанное приложение может использоваться для более продуктивной работы с файлами и сетью, а также для повышения эффективности электронного документооборота. В процессе разработки приложения необходимо учитывать специфику взаимодействия как с локальной файловой системой, так и передачей данных посредством сети. Исходя из требований к функционалу такого рода программ, было принято решение реализовать следующие функции: работа с файлами (просмотр, правка, копирование, перемещение, добавление, удаление, назначение прав доступа, редактирование атрибутов), прозрачная работа не только с локальной файловой системой, но и с виртуальными или удаленными файловыми системами, присутствует возможность гибкой настройки поддерживаемых файловых систем (ftp, sftp и т. д.), работа с архивами (упаковка, распаковка и просмотр), работа в консоли для выполнения команд ОС вручную. Открытая архитектура, позволяющая гибко добавлять дополнительные модули, возможность работы как в режиме однопанельного проводника в стиле Windows Explorer, так и в режиме двухпанельного файлового менеджера в стиле Midnight Commander, Norton Commander, Total Commander и т.д.

Потому как файловый менеджер является интерфейсом для взаимодействия с файловой системой, приведем следующую схему (рис. 1) взаимодействия, применимую как для локальной файловой системы так и для удаленной.

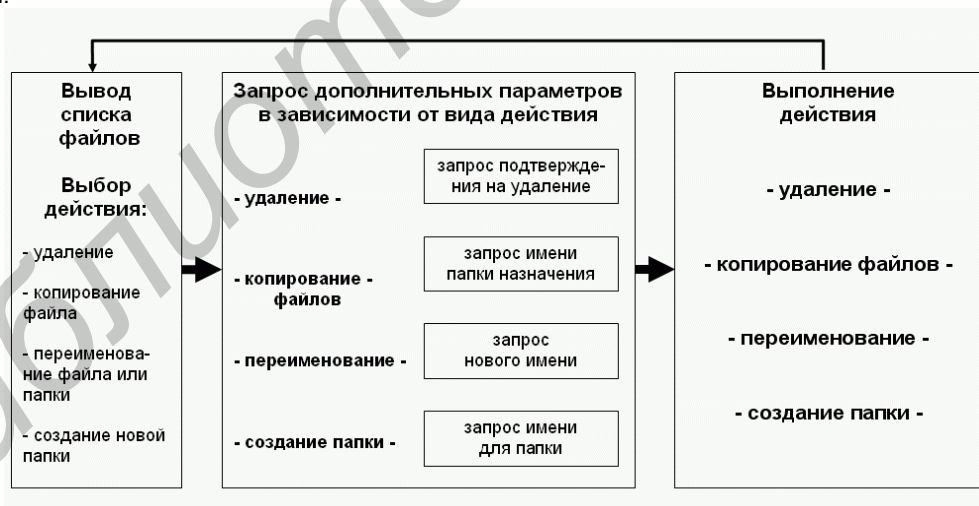


Рис. 1 –Схема работы файлового менеджера.

Основной сложностью при проектировании программ для работы с файлами является реализация доступа к файлам находящимся в файловых системах различного типа. В процессе разработки этот факт должен быть учтен. Также необходимо предусмотреть возможность гибкого расширения функционала посредством подключения независимых модулей.

Для интерфейса принято решение применить двухпанельную схему размещения списков файлов, с возможностью переключения на однопанельное отображение, таким образом используя преимущества обеих схем компоновки визуальных элементов.

Для реализации приложения выбрана среда Microsoft Visual Studio 2010, платформа Microsoft .NET версии 4.0 и объектно-ориентированный язык программирования C#.

Таким образом, была разработана архитектура многофункционального сетевого файлового менеджера,