

План тестирования

1 Введение

Данный план тестирования разработан для проверки работоспособности всех модулей разработанного проекта, а так же для выявления ошибок и неисправностей в работе приложения при выполнении определенных сценариев действий пользователя. На основании выводов, представленных в данном документе, разработчик может определить, какие изменения необходимы в новой версии ПО и какие корректировки требуется осуществить для улучшения работоспособности и производительности приложения. В данном документе описаны подходы и аспекты тестирования данного программного обеспечения, по результатам которого возможно устранение каких-либо логических ошибок и избежание некорректного исполнения задач.

2 Объект тестирования

2.1 Описание проекта и его компонентов

Приложение "Easy To Study" нацелено на облегчение учебного процесса студента. Пользователю предоставляется возможность удобного контроля таких показателей, как:

- 1) дедлайны.
- 2) отметки.
- 3) пропуски занятий.
- 4) заметки любого характера.

Также есть возможность изменения внесенной информации, просмотр статистики по пропускам и сданным лабораторным работам, а также хранение всей информации в текстовых файлах.

2.2 Атрибуты качества (по ISO 9126)

Функциональность	Пригодность для применения
	Корректность
Надежность	Завершенность
	Восстанавливаемость
Практичность	Понятность
	Простота использования
Эффективность	Временная эффективность
	Стабильность
Мобильность	Мультиплатформенность
	Сосуществование

3 Риски

В качестве начальной инициализации списка предметов, с которым в дальнейшем пользователь будет осуществлять работу, используется текстовый файл. При первом использовании приложения пользователь должен создать дерево соответствующих каталогов и текстовых документов (рис. 1). В корневом каталоге Files требуется создать два текстовых файла subjects.txt и note.txt соответственно. В файле subject.txt необходимо организовать структуры типа <кол-во лаб.раб., предмет>, пример – рис. 2. В каталогах absences и marks требуется создать файлы, названия которых соответствуют названиям предметов, перечисленных в файле subjects.txt. Остальные файлы оставить пустыми.

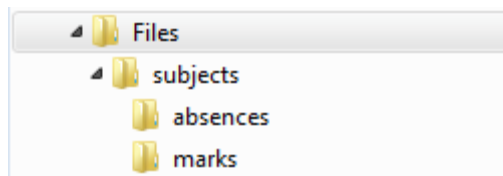


Рис.1 – дерево каталогов.

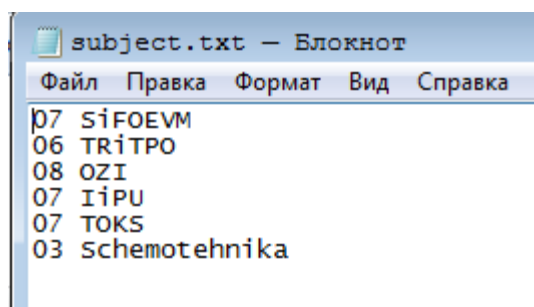


Рис.2 – организация файла subjects.txt

Данная инициализация непосредственно влияет на качество продукта, т.к. несоблюдение инструкций описанных выше ведет к некорректному исполнению функций разработанного приложения.

4 Аспекты тестирования

4.1 Пригодность для применения

Необходимо удостовериться, может ли в целом приложения выполнять главную задачу, поставленную в документе требований к ПО – облегчение учебного процесса.

4.2 Корректность

Корректность исполнения всех заявленных и реализованных функций гарантируют целостную успешность приложения.

4.3 Завершенность

Конечно же, в приложении должны быть реализованы все функции, заявленные в требованиях к проекту.

4.4 Восстанавливаемость

В случае каких-либо непредвиденных ошибок системы, либо обычного выхода из приложения, все данные занесенные пользователем должны каким-то образом храниться и перезаписываться в случае изменения их пользователем. Если же, файлы каким-то образом были утеряны, пользователь должен иметь возможность самостоятельного создания их собственными усилиями.

4.5 Понятность и простота использования

Имея цель, достижение которой возможно средствами данного ПО, пользователь должен интуитивно понимать, в каком направлении ему следует двигаться и какие действия совершить для удовлетворения своих потребностей.

4.6 Временная эффективность

Переходы от соответствующих окон и выполнения каких-либо операций должны занимать достаточно малое время ($<1с$) в целях достижения хорошей эффективности в работе.

4.7 Стабильность

Не должно возникать никаких непредвиденных ситуаций и логических ошибок при использовании, разработанного ПО.

4.8 Мультиплатформенность

Требуется установить – имеется ли возможность запуска приложения на платформе Windows/UNIX, чего необходимо было достичь в соответствии с документом требований.

4.9 Сосуществование

В процессе тестирования, требуется удостовериться, что приложение не вступает в конфликты с иным ПО и ни каким образом не оказывает влияние на систему.

5 Подходы к тестированию

Каждый из пунктов аспектов тестирования (п.п. 4.1 – 4.9) будет протестирован с помощью пользовательских тестов.

6 Представления результатов

Тестовые сценарии являются аналогом исполнения вариантов использования, описанных в документации к проекту.

П.п. 4.1 - 4.3, 4.5 – 4.7 возможно протестировать, используя выше описанный тестовый сценарий.

Пункт 4.4 проверяется путем удаления некоторых (или всех) файлов инициализации и попытки создания тестируемым лицом данных файлов самостоятельно.

Для проверки п.4.8 (мультиплатформенность) пользователю

необходимо доп. инструменты тестирования (платформа Windows и UNIX). В случае корректного запуска на обеих системах можно сделать вывод о соблюдении данного аспекта тестирования.

7 Заключение

При выполнении тестовых сценариев (п. 6), тестируемый продукт должен полностью подтвердить каждый из аспектов тестирования описанных в пункте 4. При полном соблюдении тестового плана и в случае успешного прохождения каждого из тестов, пользователю гарантируется надежность и полная работоспособность разработанного программного обеспечения.