**Содержание**

[Введение 2](#_Toc449196913)

[1 Аналитическая часть 3](#_Toc449196914)

[1.1 Описание приложения 3](#_Toc449196915)

[1.2 Анализ требований 3](#_Toc449196916)

[1.3 Предварительная оценка трудоемкости разработки 3](#_Toc449196917)

[2 Технологическая часть 4](#_Toc449196918)

[2.1 Описание проектирования логики и данных 4](#_Toc449196919)

[2.2 Определение функциональных типов по данным 4](#_Toc449196920)

[2.3 Описание проекта интерфейса приложения 4](#_Toc449196921)

[2.4 Определение транзакционных функциональных типов 4](#_Toc449196922)

[2.5 Расчет количества функциональных точек 4](#_Toc449196923)

[2.6 Определение основных технико-экономических показателей 5](#_Toc449196924)

[Заключение 6](#_Toc449196925)

[Использованные источники информации 7](#_Toc449196926)

**Введение**

В связи с распространением смартфонов среди студентов и преподавателей появляется возможность проверять знания студентов в удобном формате и без использования компьютерных классов.

Тема актуальна в связи с распространением смартфонов и доступностью интернета, образуется возможность проводить тестирование на предмет знаний студентов по выбранным дисциплинам.

Целью проектирования курсового проекта является создание мобильного приложения для составления и проведения тестирования среди студентов.

Предметом проетирования курсового проекта является разработка мобильного приложения.

Объектом исследования курсового проекта является тестирование студентов.

Для достижения цели курсового проекта нужно выполнить следующие задачи:

1. Анализировать предметную область;

2. Реализовать пользовательский интерфейс;

3. Реализовать возможности тестирования;

При разработке мобильного приложения были выявлены следующие

функциональные требования:

1. Регистрация и авторизация пользователя;

2. Возможность присваивания роли пользователю;

3. Возможность составлять тестирования;

4. Возможность прохождение тестирования;

5. Возможность получения результата.

Основанием для разработки проекта является учебный план специальности 09.02.07 «Информационные системы и программирование»

## 1. Аналитическая часть

## 1.1 Описание приложения

Сейчас трудно представить человека без смартфона. Но, не во всех учебных заведениях есть возможность оборудовать достаточное количество компьютерных классов. На помощь приходит смартфон. Он не требует использовать его в компьютерном классе, тем самым сможет упростить этапы обучения как и для студента, так и для преподавателя.

Мобильное приложение представляющее собой максимально удобный инструмент для тестирования преподавателями студентов по разным преподаваемым дисциплинам не используя компьютерный класс, но зато используя смартфон.

Мобильное приложение тестирования дает возможность преподавателю составлять тесты для дальнейшего прохождения их студентами, также дает возможность выбора тестов для дальнейшего прохождения, по завершению прохождения тестирования система выдает результаты тестирования студентам и преподавателю.

Также данное мобильное приложение имеет функции регистрации/авторизации с выбором роли (студент или преподаватель), возможность сохранения результатов тестирования уже пройденных тестов.

## 1.2 Анализ требований

Исходя из выявленных функциональных требований была построена Use-case диаграмма, представленная на рисунке 1:

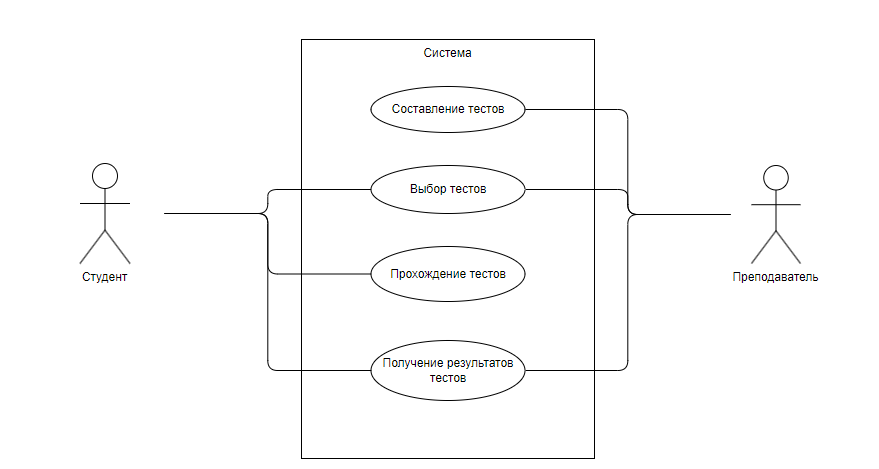


Рисунок 1. Use-case диаграмма.

На основе составленной Use-case диаграммы было выделено 4 прецедента. Рассмотрим прецедент “Выбор тестов”:

Таблица 1 — Прецедент “Выбор тестов”

|  |  |
| --- | --- |
| Название прецедента | Выбор тестов. |
| Цель | Выбор теста студентом составленного преподавателем. |
| Актеры | Пользователь студент(инициатор), Пользователь преподаватель, Система. |
| Ссылки (предусловия) | Реализованы прецеденты: Получение результатов тестирования. |
| Результаты (постусловия) | Пользователь студент заканчивает тестирование. Система определяет количество правильных и неправильных ответов. После этого система отправляет результат студенту и преподавателю для выставления оценки |

На основе выявленных прецедентов в Use-case диаграмме был составлен основной успешный сценарий, представленный в таблице 2:

Таблица 2 — Основной успешный сценарий

|  |  |
| --- | --- |
| Действия актера | Отклик системы |
| 1. Пользователь студент выбирает тест составленный преподавателем.  2. Пользователь студент проходит тестирование.  4. Пользователь студент завершает тестирование. | 3. Система ведет подсчет верных и неверных ответов студента и отправляет результат пользователю студенту и пользователю преподавателю.  5. Система отправляет результат тестирования. |

На основе успешного варианта событий можно составить альтернативный поток событий:

1. Произошла ошибка в системе. Система выдает сообщение об ошибке.

3. Данные отсутствуют в системе. Система выдает сообщение об ошибке.

4. Данные отсутствуют в системе. Система выдает сообщение об ошибке.

## 1.3 Предварительная оценка трудоемкости разработки

На основании перечня функций, выявленных на этапе анализа требований была выполнена экспертная оценка трудоемкости разработки приложения, с учетом его функций, представленная в таблице 3.

Таблица 3 – Экспертная оценка трудоемкости разработки приложения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Функция** | **Лучший случай** | **Наиболее вероятный** | **Худший случай** | **Ожидаемый случай** |
| Составление тестов | 4 | 2 | 26 | 9 |
| Выбор тестов | 2 | 4 | 20 | 6,3 |
| Прохождение тестов | 2 | 4 | 20 | 6,3 |
| Получение результатов тестов | 4 | 2 | 26 | 9 |
| **Итого** | **12** | **20** | **92** | **30,6** |

1. Составление тестов.

Oi = 4

Pi = 26

Mi = 6

Средняя трудоемкость рассчитана по формуле:

Еi = (Pi+4\*Mi + Oi)/6 = (4+4\*6+26) /6=9 чел. час

2. Выбор тестов.

Oi = 2

Pi = 20

Mi = 4

Средняя трудоемкость рассчитана по формуле:

Еi = (Pi+4\*Mi + Oi)/6 = (2+4\*4+20) /6=6.3 чел. час

3. Прохождение тестов.

Oi = 2

Pi = 20

Mi = 4

Средняя трудоемкость рассчитана по формуле:

Еi = (Pi+4\*Mi + Oi)/6 = (2+4\*4+20) /6=6.3 чел. час

4. Получение результатов тестов.

Oi = 4

Pi = 26

Mi = 6

Средняя трудоемкость рассчитана по формуле:

Еi = (Pi+4\*Mi + Oi)/6 = (4+4\*6+26) /6=9 чел. час

На основе экспертная оценки трудоемкости разработки мобильного приложения приложения было выявлено, что ожидаемый случай выполнения разработки программного продукта займет 30,6 дней.

# 2. Технологическая часть

# Описание проектирования логики и данных

По результатам анализа требований была составлена диаграмма классов проекта, представленная на рисунке 2.

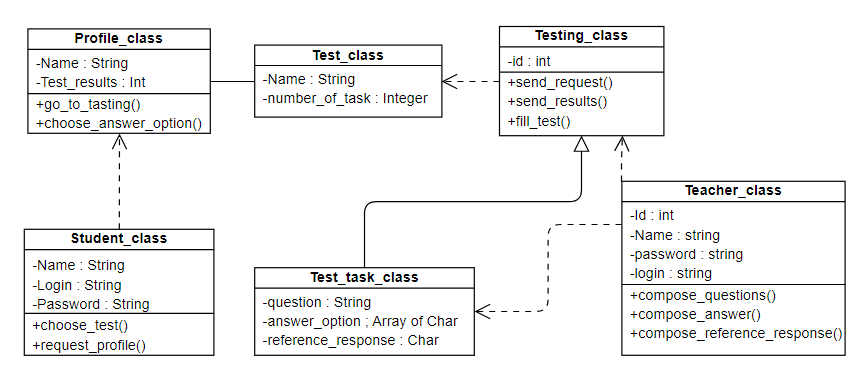
**

Рисунок 2. Диаграмма классов.

Следующим шагом проектирования является составление ER-диаграммы, показанная на рисунке 3. Диаграмма отображает состав таблиц базы данных проекта, а также связи между ними.

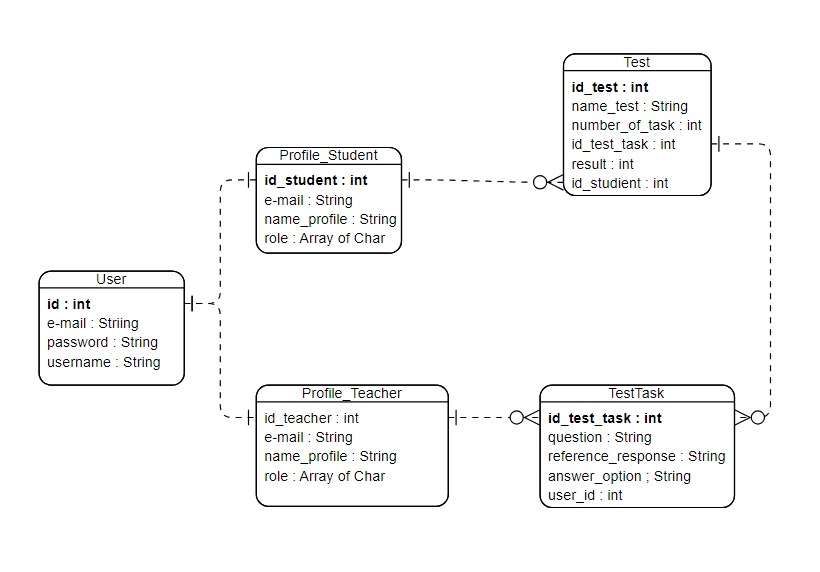
**

Рисунок 3. ER – Диаграмма

# Определение функциональных типов по данным

**Таблица 2 Оценка функциональных типов по данным**

На основании ER-диаграммы выполнили оценку функциональных типов по данным. Сначала определили тип данных системы: EIF или ILF. Затем произвели подсчет функциональных точек, связанных с данными, определили сложность данных.

Таблица 4 – Экспертная оценка трудоемкости разработки приложения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Информационный объект** | **Тип** | **RET** | **DET** | **Сложность** | **Коэффициент** |
| User | ELF | 1 | 4 | Простая | 7 |
| Profile\_Student | ELF | 1 | 4 | Простая | 7 |
| Profile\_Teacher | ELF | 1 | 4 | Простая | 7 |
| TestTask | ELF | 1 | 5 | Простая | 7 |
| Test | ILF | 1 | 7 | Простая | 7 |

Коэффициент определяется в зависимости от принадлежности объекта к ILF или EIF.

# Описание проекта интерфейса приложения

*Эскизы форм приложения*

*Схема перемещений по приложению*

# Определение транзакционных функциональных типов

*(Для каждой формы из проекта интерфейса приложения)*

Таблица 5 – Оценка транзакционных функциональных типов для формы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Транзакция** | **Тип** | **FTR** | **DET** | **Сложность** | **Коэффициент** |
| *наименование 1* | *EI* | *2* | *5* | *средняя* | *4* |
| . . . |  |  |  |  |  |
| *наименование N* |  |  |  |  |  |

# Расчет количества функциональных точек

Таблица 5 — Основной успешный сценарий

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Категория функциональных типов** | **Простые** | **Средние** | **Сложные** | **Кол-во точек** |
| Количество внутренних логических файлов | *7\*1* | *10\*4* | *0* | *47* |
| Количество внешних интерфейсных файлов |  |  |  |  |
| Количество входных элементов | *0* | *4\*2* | *0* | *8* |
| Количество выходных элементов |  |  |  |  |
| Количество внешних запросов |  |  |  |  |
| **Количество функциональных точек (UFP)** | | | |  |

Таблица 6 — Основной успешный сценарий

|  |  |  |
| --- | --- | --- |
| **№** | **Наименование фактора** | **Значение** |
| 1 | Обмен данными |  |
| 2 | Распределенная обработка данных |  |
| 3 | Производительность |  |
| 4 | Эксплуатационные ограничения |  |
| 5 | Частота транзакций |  |
| 6 | Ввод данных в режиме «онлайн» |  |
| 7 | Эффективность работы конечных пользователей |  |
| 8 | Онлайновое обновление |  |
| 9 | Сложная обработка |  |
| 10 | Повторное использование |  |
| 11 | Простота установки |  |
| 12 | Простота эксплуатации |  |
| 13 | Количество возможных установок на различных платформах |  |
| **Суммарное значение коэффициентов (М)** | |  |

VAF = (0,65 + (sum М \* 0,01)) = конкретные значения

AFP = UFP \* VAF.

*(формулы оформляются по правилам оформления формул в текстовом документе)*

# Определение основных технико-экономических показателей

*Расчет трудозатрат на разработку системы и необходимых людских ресурсов при реализации системы (описание в файле Методы оценки \ Дополнение к расчету ТЭО)*

# Заключение

# Использованные источники информации