Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Проектирование и разработка баз данных интернет-приложений

Студент: Трубач Д.С.

ФИТ 3 курс 5 группа

Преподаватель: Нистюк О.А.

Минск 2024

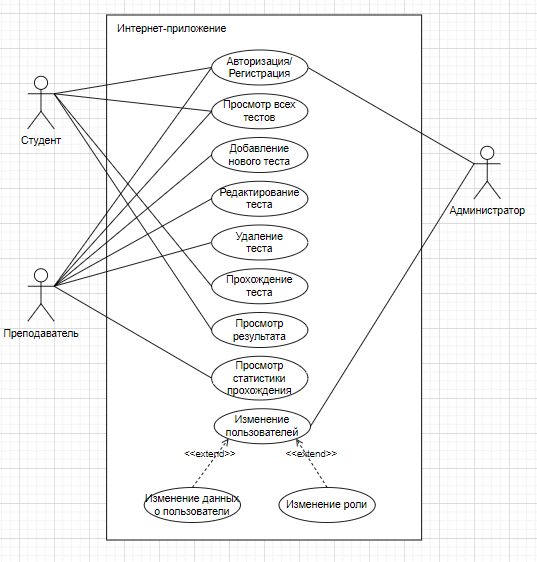
# **Лабораторная работа № 1**

1. Описать целевую аудиторию и стратегии использования интернет-приложения.

Целевая аудитория: студенты и учащиеся, преподаватели.

Стратегия использования: обучение и последующее оценивание, самостоятельное обучение, аналитика и отчетность.

1. Разработать UML диаграммы для каждой роли приложения.



1. Выделить основные сущности, описывающие предметную область.

Пользователь, Тест, Вопрос, Ответ, Результат теста

1. Определить и уточнить атрибуты для каждой сущности.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблица Тест  ID PK  Название  Описание  Предмет  ID\_пользователя FK | Таблица Вопрос  ID PK  Текст  ID\_теста FK | Таблица Ответ  ID\_ответа PK  Текст  Правильный  ID\_вопроса FK | Таблица Пользователь  ID PK  Имя  Фамилия  Пароль  Роль  Данные | Таблица Результат теста  ID PK  ID\_пользователя FK  ID\_теста FK  Дата и время  Итоговый балл |

1. Определить связи между сущностями.
2. Преобразовать сущности в таблицы базы данных согласно выбранной модели.

Тест и Вопрос: Отношение “один ко многим”. Один тест может содержать множество вопросов.

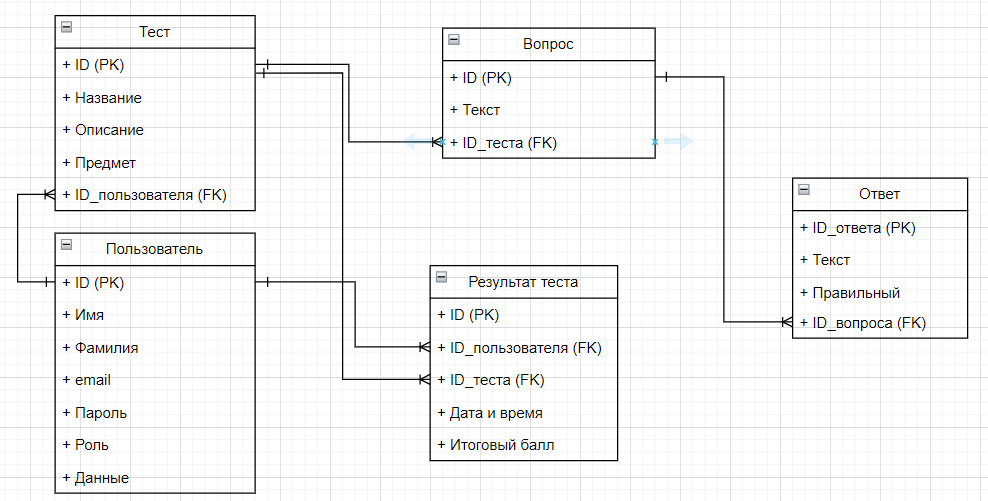
Вопрос и Ответ: Отношение “один ко многим”. Один вопрос может иметь множество ответов.

Пользователь и Результат\_теста: Отношение “один ко многим”. Один пользователь (студент) может иметь множество результатов тестов.

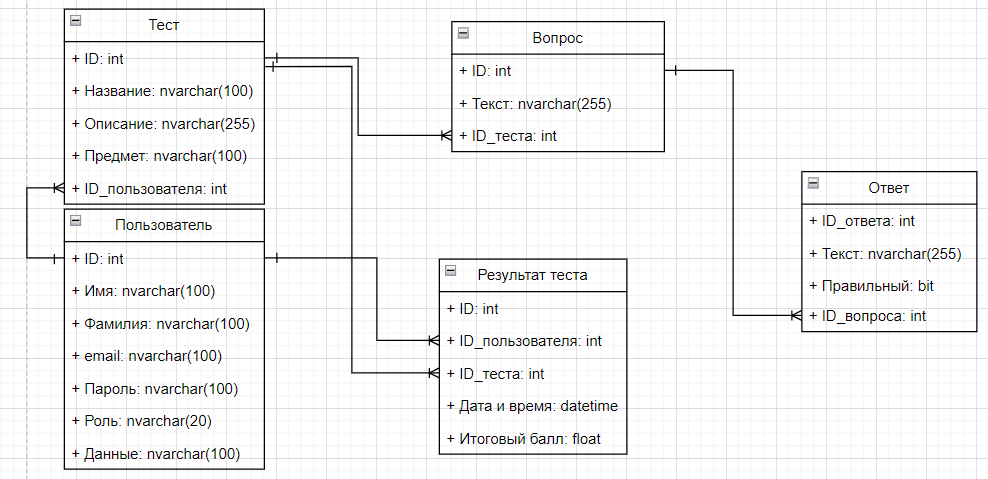
Тест и Результат\_теста: Отношение “один ко многим”. Один тест может иметь множество результатов тестов.

1. Разработать логическую схему базы данных.

Логическая схема БД:



Физическая схема БД:



1. Проверить логическую схему базы данных на:

* соответствие нормальным формам до 4 НФ включительно;
  1. Чтобы таблица соответствовала 1-й нормальной форме (1NF), необходимо, чтобы все значения ее полей были неделимыми и не вычисляемыми, а все записи – уникальными (не должно быть полностью совпадающих строк). **ПРИМЕР:** в таблице Пользователь, каждое поле (ID, Имя, Роль, Данные) содержит атомарные значения, то есть каждое поле содержит только одно значение.
  2. Чтобы таблица соответствовала 2-й нормальной форме (2NF), необходимо, чтобы она находилась в 1-й нормальной форме и все не ключевые поля полностью зависели от ключевого. **ПРИМЕР:** В таблице Результат\_теста, все не ключевые атрибуты (Дата\_и\_время, Итоговый\_балл) полностью зависят от первичного ключа (ID\_пользователя, ID\_теста).
  3. Для перехода к 3-й нормальной форме (3NF), необходимо обеспечить, чтобы все таблицы находились во 2-й нормальной форме и все не ключевые поля в таблицах не зависели взаимно друг от друга. **ПРИМЕР:** в таблице Ответ, все не ключевые атрибуты (Текст, Правильный) взаимно независимы. Например, текст ответа не зависит от того, является ли ответ правильным, и наоборот.
  4. Для перехода к 4-ой нормальной форме (4NF), требуется отсутствие многозначных зависимостей. **ПРИМЕР:** в таблице Тест, нет многозначных зависимостей. Например, предмет теста не зависит от названия теста, и наоборот.
* присутствие денормализованных атрибутов;

**Денормализация** в базах данных обычно используется для улучшения производительности за счет добавления избыточных данных или группирования данных. В моей базе данных системы тестирования не видно явных денормализованных атрибутов. Все атрибуты в каждой таблице кажутся уникальными и необходимыми для этой таблицы.   
Например, в таблице Тест атрибуты Название, Описание и Предмет уникальны для каждого теста и не дублируются в других таблицах. Однако Описание и Предмет могут дублироватся.

1. Оценить сроки проекта и требуемые ресурсы.
2. **Оценка сроков и ресурсов** для разработки базы данных может сильно варьироваться в зависимости от многих факторов, включая сложность проекта, опыт команды и доступные ресурсы. Вот примерная оценка:
3. **Анализ и проектирование (1-2 недели)**: Этот этап включает в себя определение требований, проектирование схемы базы данных и создание UML-диаграмм.
4. **Разработка (2-4 недели)**: На этом этапе происходит непосредственное создание базы данных, включая определение таблиц, связей и ограничений.
5. **Тестирование (1-2 недели)**: После разработки базы данных следует этап тестирования, на котором проверяется корректность работы базы данных и соответствие ее требованиям.
6. **Документация (1 неделя)**: На этом этапе создается документация, которая описывает структуру базы данных, ее функции и использование.

Таким образом, общий срок разработки может составить от 5 до 9 недель.

Что касается ресурсов, то для разработки такой базы данных потребуется, как минимум, один опытный разработчик баз данных, а также доступ к соответствующему программному обеспечению для создания и управления базой данных. Кроме того, может потребоваться время и ресурсы на обучение пользователей работе с новой системой.