

МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической теории  
упругости и биомеханики

**Разработка сервиса знакомств**

**АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

студента 4 курса 442 группы  
направления 09.03.03 Прикладная информатика

механико-математического факультета

Шишилина Даниила Игоревича

Научный руководитель  
к.ю.н., доцент

\_\_\_\_\_

Р. В. Амелин

Зав. кафедрой  
д.ф.-м.н., профессор

\_\_\_\_\_

Л. Ю. Коссович

Саратов 2023 год

**Введение.** Современный образ жизни заставляет людей искать новые способы знакомства и общения. Один из таких способов - использование сервисов знакомств, которые позволяют людям находить партнеров для знакомства и создания отношений в онлайн-режиме. Сервисы знакомств - это веб-приложения, разработанные для удобного и быстрого поиска подходящих кандидатов для знакомства.

Однако, несмотря на широкий выбор существующих сервисов знакомств, многие из них имеют ограниченный функционал и неудобный интерфейс, что может отталкивать пользователей. Некоторые сервисы знакомств не предоставляют возможность поиска по разным критериям, что затрудняет выбор подходящего партнера. Другие сервисы знакомств не предоставляют достаточно информации о кандидатах, что может приводить к неэффективности поиска и разочарованию в использовании таких сервисов.

Цель данной дипломной работы является разработка сервиса знакомств, позволяющий пользователям быстро и легко находить подходящего партнера для знакомства.

Для достижения цели работы были поставлены следующие задачи:

1. Анализ рынка сервисов знакомств.
2. Определение требований к разрабатываемому сервису
3. Разработка архитектуры и функционала сервиса.
4. Реализация сервиса.

В качестве объекта исследования выступает веб-сервис знакомств, позволяющий пользователям быстро и легко находить оппонента для знакомства, а также язык программирования C#.

Выпускная квалификационная работа состоит из введения, 3 глав, заключения, списка использованных источников, включающего 24 наименования, работа изложена на 44 листах машинописного текста, содержит 16 рисунков и 2 листинга. Далее приведены наименования глав:

1. Анализ предметной области;
2. Проектирование сервиса;
3. Реализация.

**Основное содержание работы.** В первой главе рассматривается предметная область сервиса знакомств "Duo". Глава разделена на четыре раздела:

1. Описание предметной области. В данной главе рассказывается о сервисе знакомств "Duo". Описывается основная цель сервиса, состоящая в помощи пользователям в поиске знакомств, рассмотрены основные направления деятельности сервиса, а также приведена организационная структура сервиса.
2. Обоснование необходимости создания сервиса знакомств. В этой главе обсуждается необходимость разработки сервиса знакомств "Duo" в связи с современным образом жизни, который характеризуется высокой степенью занятости, частыми переездами и уменьшением времени на личную жизнь. В главе также представлен анализ двух существующих сервисов знакомств: «LovePlanet.ru» и «Teamo». Описывается основной функционал каждого сервиса, используемые технологии и некоторые недостатки.
3. Итог сравнительного анализа. Анализ двух рассмотренных сервисов знакомств, «LovePlanet.ru» и «Teamo», выявил их преимущества и недостатки.
4. Постановка задачи. В этой главе говорится о постановке задач для разработки сервиса знакомств.

Вторая глава посвящена проектированию веб-сервиса и разделяется на 3 раздела:

В первом разделе первой главы рассматривается роль и важность архитектур информационных систем, особенно в контексте разработки сервиса знакомств "Duo". Затем рассматриваются различные типы архитектур информационных систем, и в контексте разработки сервиса знакомств "Duo" указывается, что многоуровневая архитектура является подходящим выбором. Такая архитектура, продемонстрированная на рисунке 1, позволяет явно

разделять бизнес-логику от пользовательского интерфейса, что обеспечивает гибкость, масштабируемость и легкость внесения изменений.

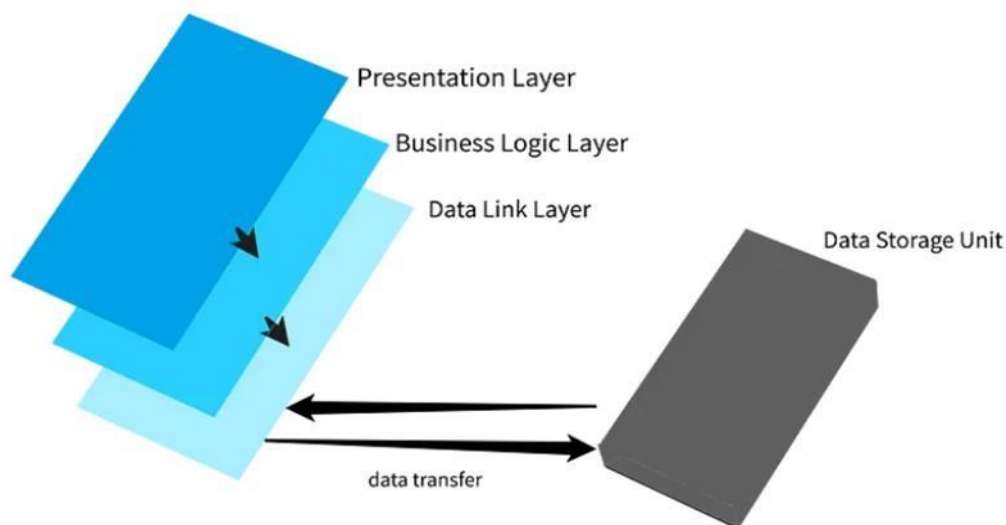


Рисунок 1 - Многоуровневая архитектура информационной системы

Во втором разделе первой главы рассматриваются теоретические аспекты реляционных баз данных. Приводится краткая история возникновения таких баз данных и выделяются их особенности. Затем описывается процесс проектирования базы данных для целевого веб-сервиса, который разделен на логический и физический уровни. На логическом уровне проектирования выполняется выделение сущностей, их атрибутов и взаимосвязей между ними. Для этого используется ER-моделирование, и строится диаграмма "сущность-связь", продемонстрированная на рисунке 2. Эта диаграмма представляет структуру базы данных, отражает взаимосвязи между сущностями и их атрибутами.

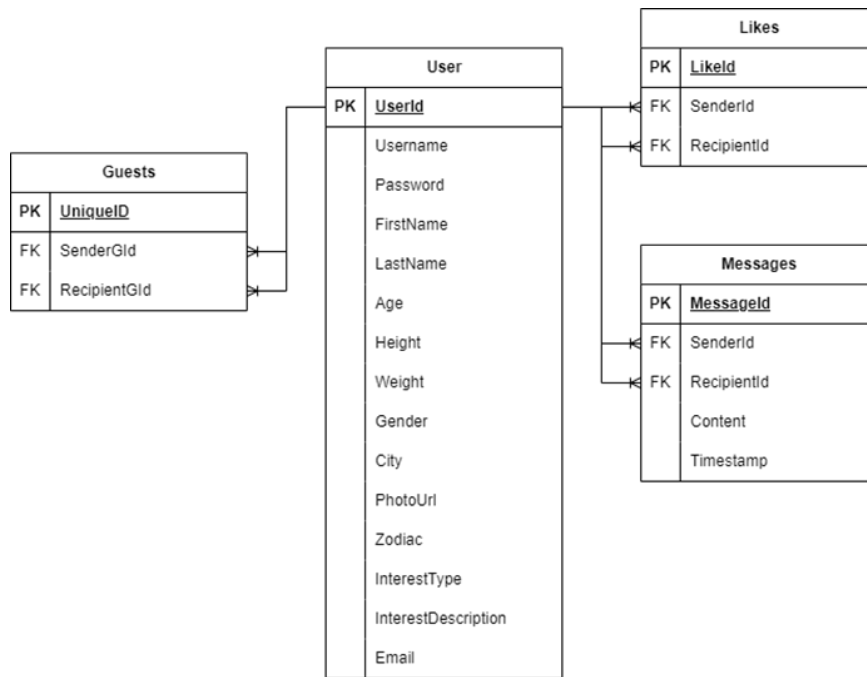


Рисунок 2 - ER модель базы данных

На физическом уровне проектирования создается физическая модель базы данных, которая будет реализована с использованием конкретной системы управления базами данных (СУБД). В данном случае используется СУБД MSSQL. После этого строится физическая диаграмма, указанная на рисунке 3, которая показывает детали физической реализации базы данных, такие как таблицы, поля, индексы и связи между ними.

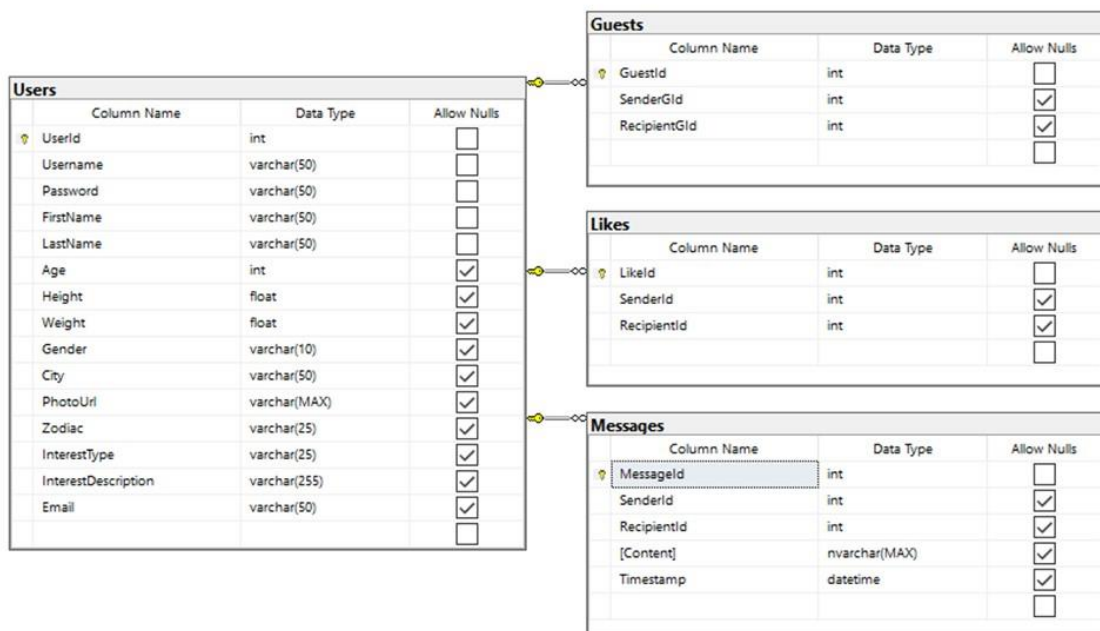


Рисунок 3 - Физическая модель базы данных

В третьем разделе второй главы описываются используемые, которые были использованы для разработки как клиентской, так серверной части сервиса.

Для серверной части использовались:

- Платформа .Net Core
- Entity Framework Core (EF Core)
- Язык программирования C#
- IDE «Visual Studio»

Для клиентской части использовались:

- Язык HTML
- Язык CSS
- Язык JavaScript

Третья глава посвящена реализации и разделяется на 4 раздела:

В первом разделе третьей главы рассматривается использование пакета "Microsoft.EntityFrameworkCore" для удобного взаимодействия с базой данных, предоставляющего классы и методы для определения моделей данных и выполнения запросов. Для работы с базой данных SQL Server используется пакет "Microsoft.EntityFrameworkCore.SqlServer", обеспечивающий провайдер данных и интеграцию с SQL Server.

Во втором разделе третьей главы рассматривается использование метода "Database First" для работы с уже созданной базой данных. Строка подключения продемонстрирована на листинге 1.

Листинг 1 – Команда для подключения к базе данных

```
Scaffold-DbContext  
"Server=POKEMON\SQLEXPRESS;Database=Krop;TrustServerCertificate=True;  
Trusted_Connection=True;" Microsoft.EntityFrameworkCore.SqlServer -project  
DAL
```

После подключения к базе данных произойдет генерация классов моделей данных. На основе анализа существующей базы данных, «Entity Framework Core» сгенерирует классы моделей данных.

В третьем разделе третьей главы рассматривается реализация регистрации пользователей и их авторизация с использованием кода, который указан в приложении. После успешной регистрации пользователю предоставляется возможность заполнить данные, включая загрузку фотографии профиля. Для загрузки фотографии используется код, который сохраняет файл в определенной папке и сохраняет путь к файлу в базе данных. Для каждой фотографии, загружаемой пользователем, присваивается уникальное именование. Файлы располагаются в каталоге «wwwroot/css» и структура этого каталога продемонстрирована на рисунке 4.

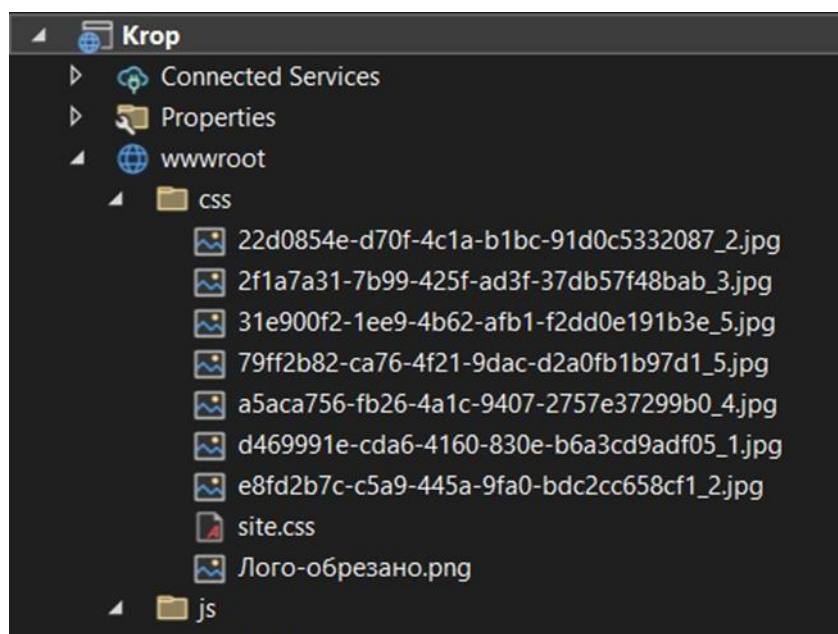


Рисунок 4 - Структура каталога wwwroot/css

Заключительный раздел третьей главы посвящен реализации и демонстрации пользовательского интерфейса. Для этого используется паттерн MVC (Model-View-Controller) в сервисе знакомств. MVC разделяет приложение на три основных компонента: модель, представление и контроллер.



1. Модель (Model): Модель отвечает за бизнес-логику и доступ к данным. В контексте "Duo" модель включает компоненты, связанные с базой данных.
2. Представление (View): Представление отображает данные пользовательского интерфейса и взаимодействует с пользователем.
3. Контроллер (Controller): Контроллер является посредником между моделью и представлением.

Для работы с пользовательским интерфейсом в сервисе используются файлы с расширением "cshtml". Эти файлы, известные как Razor View, позволяют создавать пользовательский интерфейс веб-приложений с использованием серверного кода, такого как C#.

Пользователь может зарегистрироваться, заполнив необходимые данные, и после успешной регистрации будет перенаправлен в раздел "Мой профиль", где видит информацию о себе. Окно регистрации нового пользователя продемонстрировано на рисунке 5.

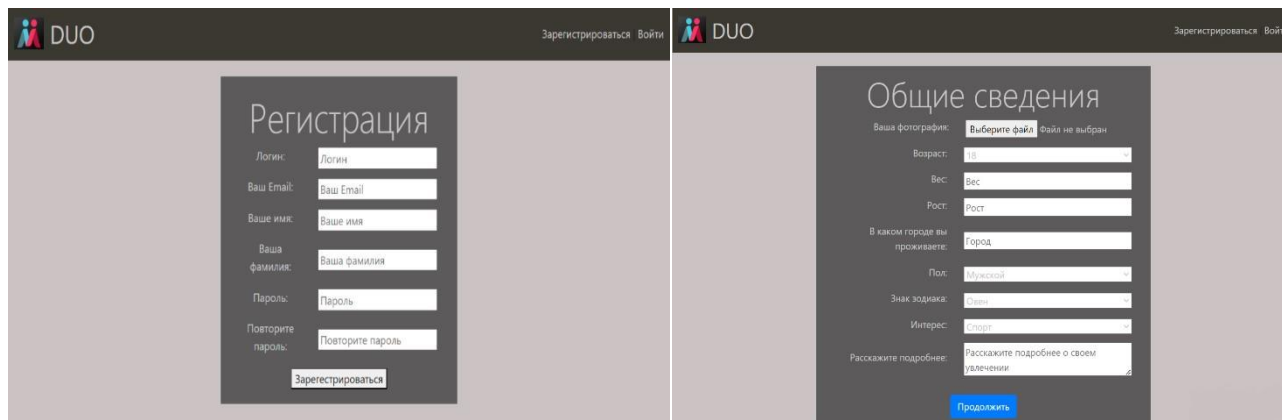


Рисунок 5 - Регистрация нового пользователя

В случае, когда пользователь уже зарегистрирован, он может ввести свой логин и пароль, тем самым авторизоваться. Если информация устарела, можно её отредактировать в разделе «Редактировать профиль». Далее пользователь может просмотреть анкеты других пользователей в разделе "Просмотр анкет". Можно использовать фильтр для уточнения результатов поиска в соответствии с предпочтениями. Возможность находить анкеты по определённым предпочтениям, указанным в фильтрах, показано на рисунке 6.

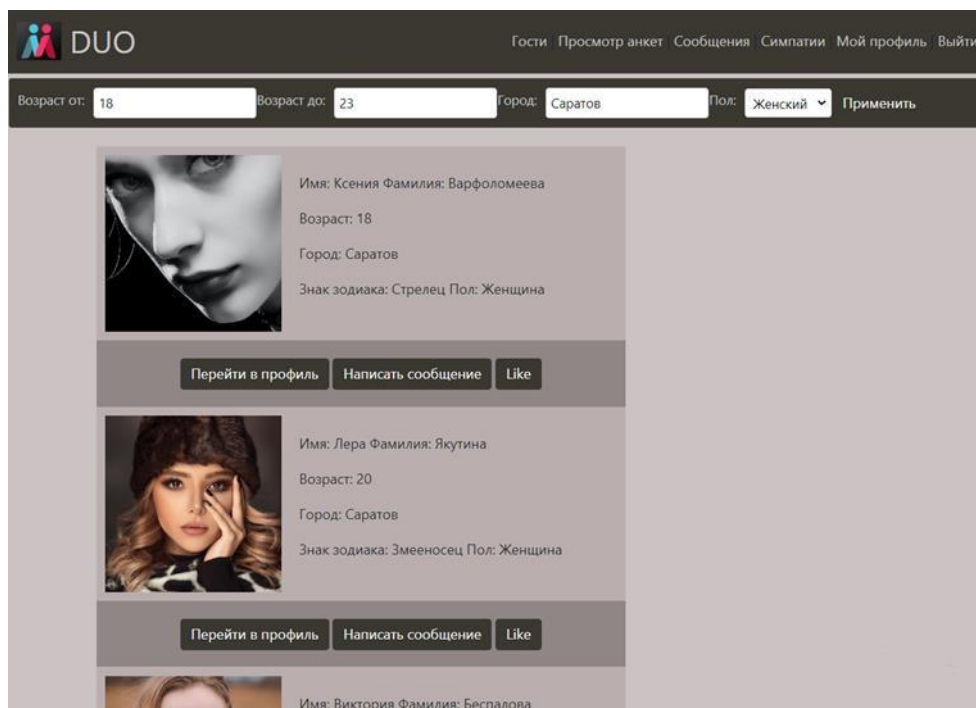


Рисунок 6 - Просмотр анкет пользователей

Если понравилась анкета, можно перейти в профиль пользователя, отправить сообщение или добавить пользователя в раздел "Симпатии". Раздел "Сообщения" предоставляет возможность создавать чаты с другими пользователями, общаться и договариваться о встрече.

**Заключение.** В современном обществе, где люди сталкиваются с ограничениями по времени и расстоянию, сервисы знакомств становятся все более актуальными и востребованными. Они предоставляют возможность людям расширить свои круги общения, найти партнера для знакомства и создания отношений, не ограничиваясь только своим географическим положением или сферой деятельности. Сервисы знакомств облегчают процесс поиска подходящего партнера, предоставляя разнообразные возможности для выбора и общения.

В ходе разработки данного сервиса знакомств было проведено аналитическое исследование существующих решений на рынке. Были выявлены недостатки и ограничения, с которыми пользователи сталкиваются при использовании подобных сервисов. Также были определены требования к разрабатываемому сервису, чтобы обеспечить удобство и эффективность его использования.

Архитектура и функционал сервиса были разработаны с учетом современных технологий и требований. Были выбраны оптимальные инструменты и технологии, которые обеспечили гибкость и масштабируемость разработки. В ходе работы над сервисом знакомств "Duo" были использованы актуальные технологии, такие как Entity Framework (EF) Core и многоуровневая архитектура информационной системы. В заключении стоит подвести итоги проделанной работы:

1. Был проанализирован рынок сервисов знакомств;
2. Были определены требования к информационной системе;
3. Были разработаны архитектура и функционал сервиса;
4. Реализован базового функционала сервиса.

Цель, поставленная в начале работы, достигнута, задачи для её достижения выполнены.