Федеральное государственное бюджетное образовательное учреждение

высшего образования (ФГБОУ ВО)

«Дальневосточный государственный университет путей сообщения»

Кафедра: «Информационные технологии и

системы»

Лабораторная работа № 3

Тема: «Разработка диаграммы классов»

Выполнил: Козеко М.В.

гр.СО251КОБ

Проверил: Анисимов В.В.

Сазанова Е.В.

Хабаровск, 2020

**Задание на выполнение лабораторной работы**

Разработать диаграммы классов по индивидуальному заданию:

- диаграмму классов анализа, описывающую информационную систему в целом;

- логическую (по-русски) и физическую (с учетом выбранной СУБД) диаграммы классов, описывающие структуру БД;

- логическую (по-русски) и физическую (с учетом выбранного языка программирования) диаграммы классов, описывающие структуру клиентского ПО.

**Ход выполнения работы**

***Диаграмма классов анализа***

Класс анализа – это укрупненная абстракция, которая на концептуальном уровне (без точного определения атрибутов и операций) описывает некоторый фрагмент системы.

Существует три вида классов анализа:

* граничный;
* управляющий;
* сущности.

Диаграмма классов анализа является прообразом классической диаграммы классов. Элементами, отображаемыми на диаграмме, являются классы и отношения между ними.

Назначение классов анализа:

* граничный класс – используется для моделирования взаимодействия между системой и актерами (пользователями, внешними системами или устройствами);
* управляющий класс – отвечает за координацию, взаимодействие и управление другими объектами, выполняет сложные вычисления, управляет безопасностью, транзакциями и т. п.;
* класс сущности – используется для моделирования долгоживущей, нередко сохраняемой информации. Классы сущности являются абстракциями основных понятий предметной области – людей, объектов, документов и т. д., как правило, хранимых в табличном или ином виде.

Связи между классами анализа отображаются с использованием отношений пяти видов:

* ассоциаций – показывает, что объекты одного класса содержат информацию о существовании (наличии в памяти) объектов другого класса и между ними имеется некоторая логическая или семантическая связь;
* агрегаций – указывает на отношение «часть–целое» и отображается сплошной линией с не закрашенным ромбиком со стороны «целого»;
* композиций – аналогично агрегации, в которой «части» не могут существовать отдельно от «целого»;
* обобщения – является обычным таксонометрическим отношением между более общим (абстрактным) классом (родителем или предком) и его частным случаем (дочерним классом или потомком);
* зависимостей – означает, что в спецификации или теле методов объектов одного класса (зависимого) выполняется обращение к атрибутам, методам или непосредственно к объектам другого класса.

Диаграмма классов анализа изображена на рисунке 1.

На диаграмме в виде классов отображено клиент-серверное взаимодействие между приложением и сервером. Соединение обеспечивается некоторым управляющим классом, который позволяет всем граничным классам приложения отображать информацию, которая содержится в таблицах базы данных и наоборот, вносить изменения в эти же таблицы. Все граничные классы представляют собой веб-страницы сайта, тогда как все сущности представляют собой таблицы в базе данных.

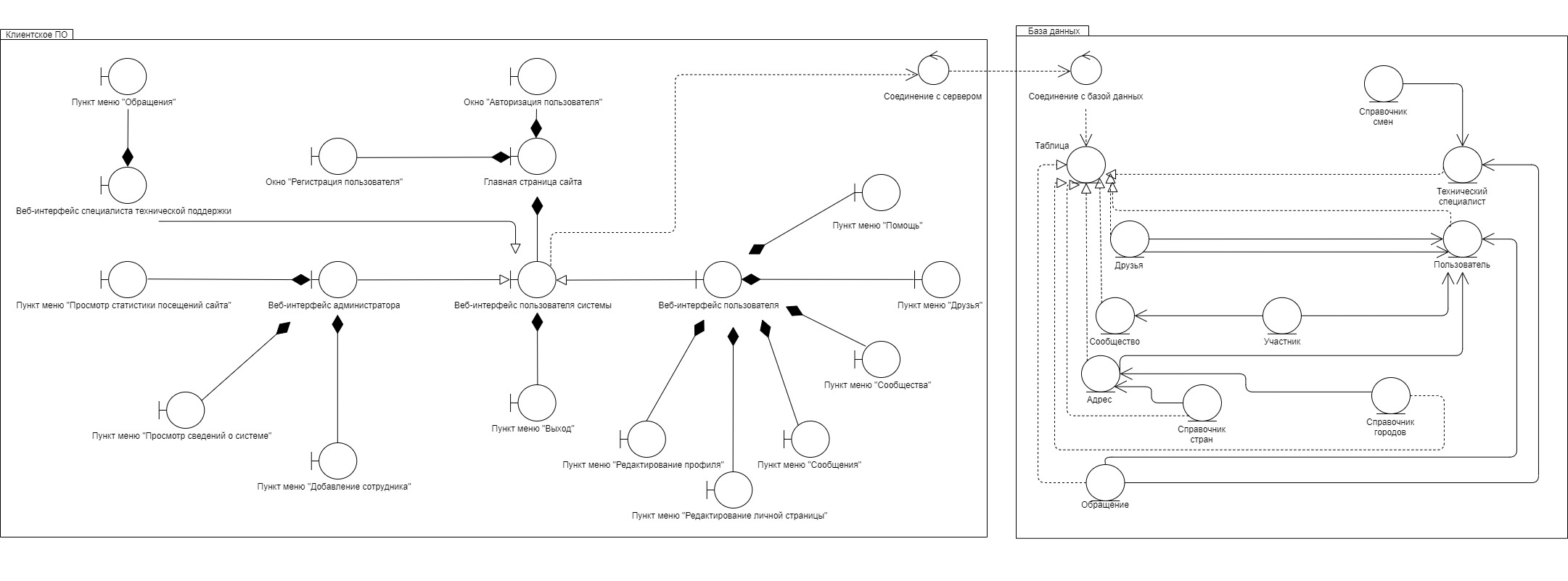


Рисунок 1 Диаграмма класса анализа

**Диаграммы классов**

Диаграммы классов используются при моделировании информационных систем наиболее часто. Они являются одной из форм статического описания системы с точки зрения ее проектирования, показывая ее структуру. Диаграмма классов не отображает динамическое поведение объектов, изображенных на ней классов.

На диаграммах классов показываются классы, интерфейсы и отношения между ними.

Диаграмма классов представляет собой граф, вершинами которого являются элементы типа «классификатор», связанные различными типами структурных отношений.

Существуют разные точки зрения на построение диаграмм классов в зависимости от целей их применения:

* концептуальная точка зрения – диаграмма классов описывает модель предметной области, в ней присутствуют только классы прикладных объектов;
* точка зрения спецификации – диаграмма классов применяется при проектировании информационных систем;
* точка зрения реализации – диаграмма классов содержит классы, используемые непосредственно в программном коде.

Классы могут иметь логическую и физическую реализации. Логические диаграммы классов в отличие от физических, строятся без привязки к языкам программирования.

В ходе выполнения курсового проекта было разработано два типа диаграмм классов: для клиентского приложения и для серверного приложения.

Прежде чем перейти к разработке этих двух моделей, опишем структуру таблиц базы данных:

Таблица 1. Адрес (Address)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер адреса | AddressID | Int | int | Уникальный идентификатор адреса |
| Страна | Country | Directory\_Country | int | Страна проживания |
| Город | City | Directory\_City | int | Город проживания |

Таблица 2. Участник (Member)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер участника | MemberID | int | int | Уникальный идентификатор участника |
| Номер пользователя | UserID | int | int | Уникальный идентификатор пользователя |
| Номер сообщества | CommunityID | int | int | Уникальный идентификатор сообщества |
| Роль | Role | string | nvarchar(15) | Роль пользователя в сообществе |

Таблица 3. Сообщество(Community)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер сообщества | CommunityID | int | int | Уникальный идентификатор сообщества |
| Название | Name | string | Nvarchar(30) | Название сообщества |
| Тема | Theme | string | Nvarchar(30) | Тема сообщества |
| Описание | Discription | string | Text | Описание сообщества |
| Фото | Photo | image | image | Фотография сообщества |

Таблица 4. Пользователь(User)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер пользователя | UserID | int | int | Уникальный идентификатор пользователя |
| Логин | Login | string | Nvarchar(20) | Логин пользователя |
| Пароль | Password | string | Nvarchar(30) | Пароль пользователя |
| Фамилия | Surname | String | Nvarchar(20) | Фамилия пользователя |
| Имя | Name | String | Nvarchar(20) | Имя пользователя |
| Отчество | Patronymic | String | Nvarchar(20) | Отчество пользователя |
| Эл. Почта | Email | String | Nvarchar(20) | Электронная почта пользователя |
| Фото профиля | UserPhoto | Image | Image | Фото профиля пользователя |

Таблица 5. Технический специалист(Specialist)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер технического специалиста | SpecialistID | int | int | Уникальный идентификатор технического специалиста |
| Логин | Login | string | Nvarchar(20) | Логин технического специалиста |
| Пароль | Password | string | Nvarchar(30) | Пароль технического специалиста |
| Фамилия | Surname | String | Nvarchar(20) | Фамилия технического специалиста |
| Имя | Name | String | Nvarchar(20) | Имя технического специалиста |
| Отчество | Patronymic | String | Nvarchar(20) | Отчество технического специалиста |
| Должность | Position | String | Nvarchar(20) | Должность технического специалиста |
| Номер смены | WorkTime | Directory\_WorkTime | int | Номер рабочей смены специалиста |

Таблица 6. Друзья(Friends)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер записи | RecordID | int | int | Уникальный идентификатор записи о дружбе пользователей |
| Номер пользователя | UserID | Int | Int | Уникальный идентификатор пользователя |
| Номер пользователя | UserID | Int | Int | Уникальный идентификатор пользователя |

Таблица 7. Обращение (Request)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер обращения | RequestID | int | int | Уникальный идентификатор обращения |
| Номер пользователя | UserID | Int | Int | Уникальный идентификатор пользователя |
| Номер технического специалиста | SpecialistID | int | int | Уникальный идентификатор технического специалиста |
| Дата создания | CreationDate | DateTime | DateTime | Дата создания обращения |
| Дата закрытия | ClosingDate | DateTime | DateTime | Дата закрытия обращения |
| Текст обращения | Message | String | Text | Текст обращения пользователя |

Таблица 8. Справочник смен (Directory\_WorkTime)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер смены | WorkTimeID | int | int | Номер рабочей смены специалиста |
| Смена | WorkTime | string | Nvarchar(20) | Смена специалиста(дневная, вечерняя, ночная) |

Таблица 9. Справочник городов (Directory\_City)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер города | CityID | int | int | Номер города |
| Город | City | string | Nvarchar(20) | Название города |

Таблица 10. Справочник стран (Directory\_Country)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название поля | Название поля на английском | Тип данных в с# | Тип данных в SQL Server | Описание |
| Номер страны | CountryID | int | int | Номер страны |
| Страна | Country | string | Nvarchar(20) | Название страны |

Далее приведены диаграммы классов для приложения (рис. 2 и рисунок 3) и диаграммы классов для БД (рис. 4 и рис. 5). Каждый вид диаграмм представлен в двух экземплярах: логическом (на русском языке) и физическом (с учетом языка программирования c#).

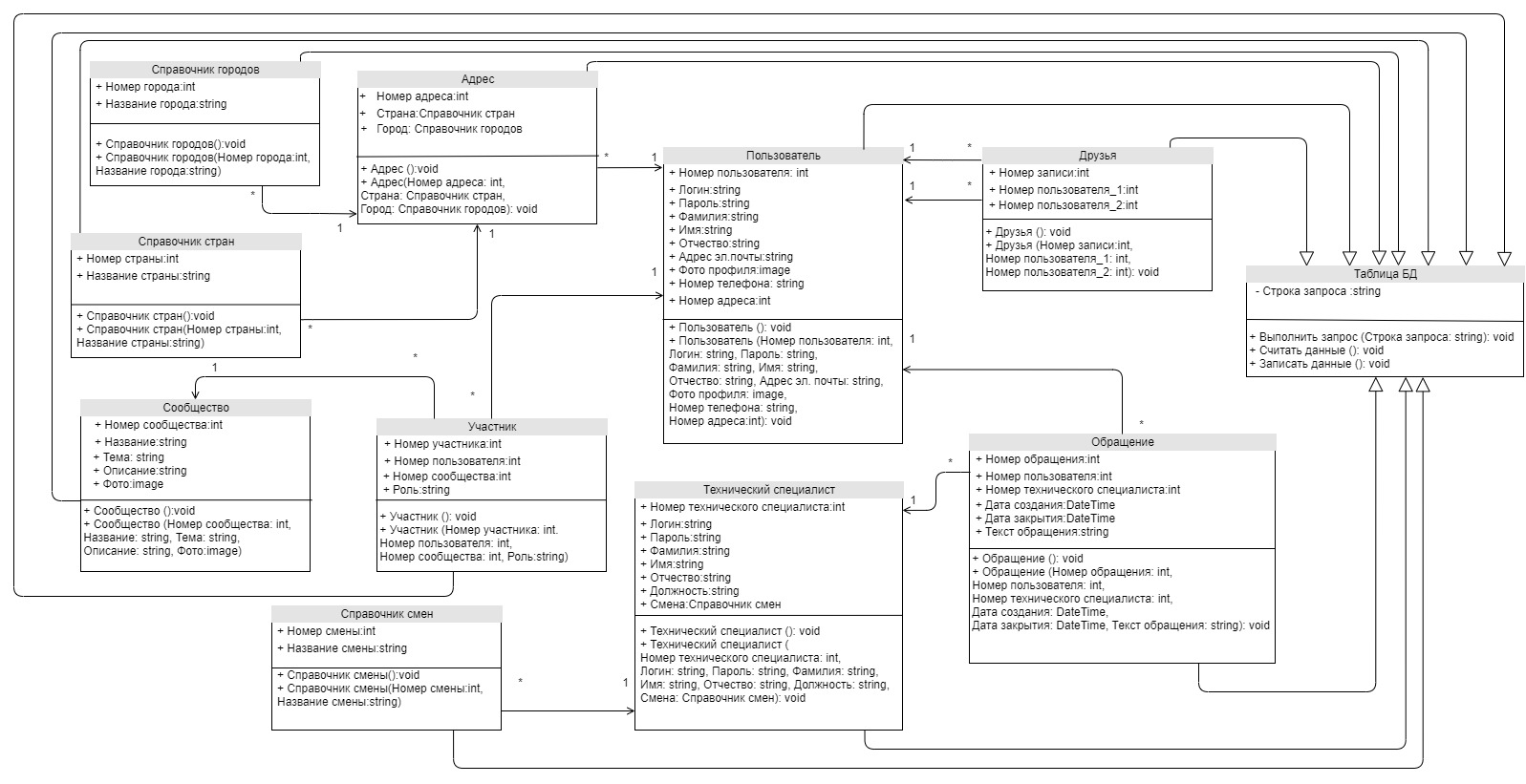


Рисунок 2 Логическая диаграмма классов приложения

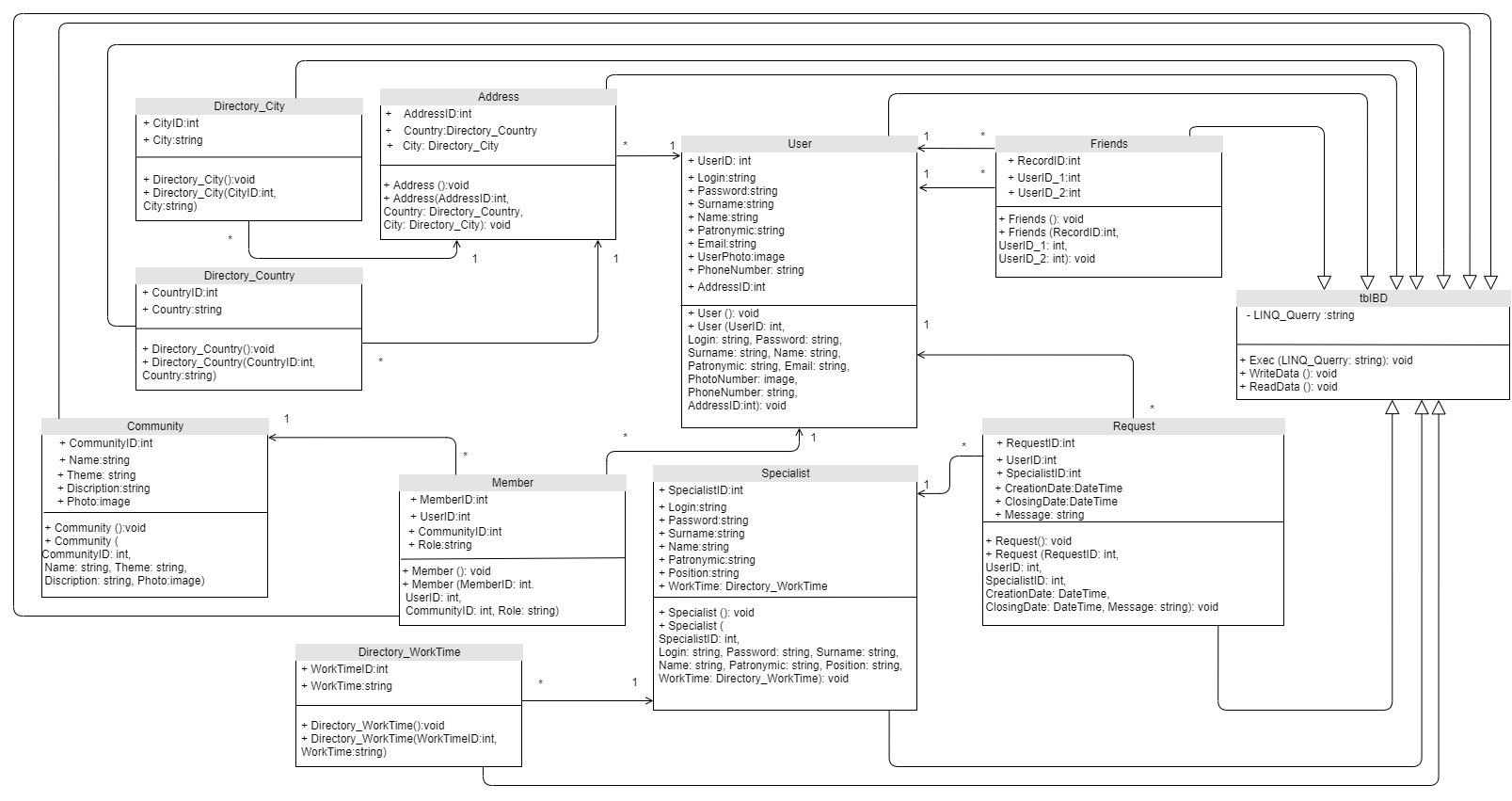


Рисунок 3 Физическая диаграмма классов приложения

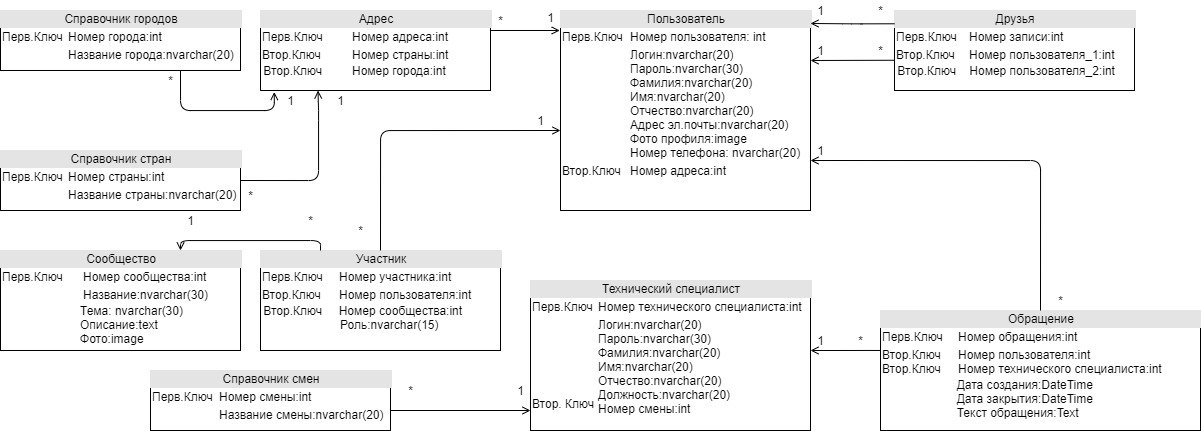


Рисунок 4 Логическая диаграмма базы данных

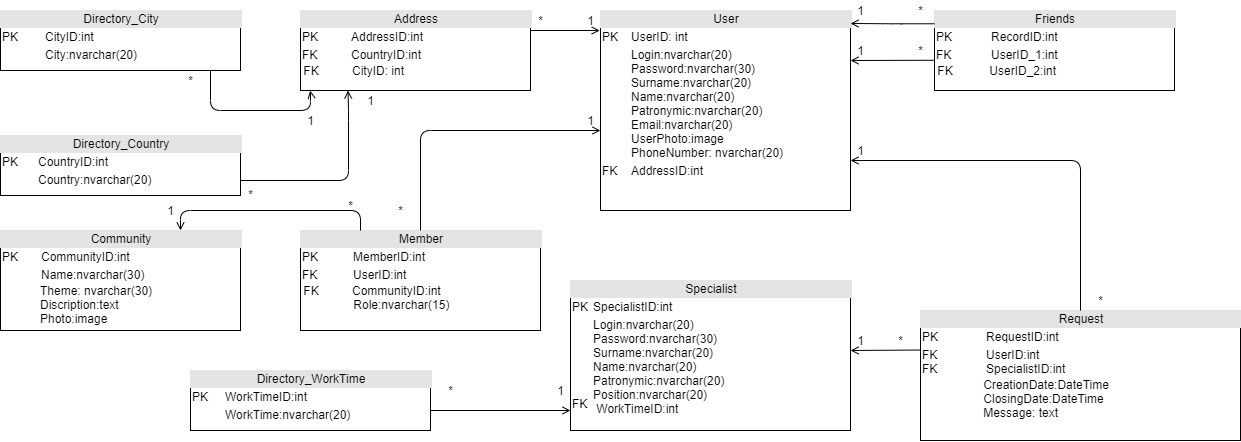


Рисунок 5 Физическая диаграмма базы данных