ПРОЕКТИРОВАНИЕ АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ ОТРАЖЕНИЯ И ОРГАНИЗАЦИИ СОЦИАЛЬНЫХ ВЗАИМООТНОШЕНИЙ

пояснительная записка к проекту

|  |  |
| --- | --- |
|  | Заказчик:  В.О. Фатхулова  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г. |
|  | Выполнили:  И.А. Искужин,  В.М. Костусенко  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г. |

2021

АННОТАЦИЯ

Пояснительная записка к проекту содержит постановку и веб-приложение решения задачи «Проектирование автоматизированной информационной системы для отражения и организации социальных взаимоотношений».

Веб-приложение «Write Me» написана на языке C# в среде программирования Visual Studio 2019 с использованием сервера баз данных MySQL 8.0 и веб-технологии ASP Net. Core и предназначена для работы в браузерах любой операционной системы и отлажена на данных контрольного примера.

## СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 3](#_Toc90137122)

[ВВЕДЕНИЕ 5](#_Toc90137123)

[1 Описание предметной области 7](#_Toc90137124)

[2 Техническое задание 8](#_Toc90137125)

[2.1 Основание для разработки 8](#_Toc90137126)

[2.2 Назначение разработки 8](#_Toc90137127)

[2.3 Требования к программе 9](#_Toc90137128)

[2.4 Требования к надежности 9](#_Toc90137129)

[2.5 Требования к составу и параметрам технический средств 9](#_Toc90137130)

[2.6 Требования к информационной и программной совместимости 9](#_Toc90137131)

[2.7 Требования к транспортированию и хранению 9](#_Toc90137132)

[2.8 Специальные требования 10](#_Toc90137133)

[2.9 Требования к программной документации 10](#_Toc90137134)

[2.10 Технико-экономические обоснование 10](#_Toc90137135)

[2.11 Описание входной информации 11](#_Toc90137136)

[2.12 Описание выходных данных 11](#_Toc90137137)

[3 Диаграмма Ганта 13](#_Toc90137138)

[4 Проектирование ПО 16](#_Toc90137139)

[4.1 Контекстная диаграмма и декомпозиция 1-го уровня 16](#_Toc90137140)

[4.2 Диаграмма вариантов использования 17](#_Toc90137141)

[4.3 Диаграмма классов 18](#_Toc90137142)

[5 Описание структуры базы данных 18](#_Toc90137143)

[6 Описание программы 23](#_Toc90137144)

[6.1 Назначение программы 26](#_Toc90137145)

[6.2 Условия выполнения программы 26](#_Toc90137146)

[6.3 Выполнение программы 27](#_Toc90137147)

[7 Руководство пользователя 32](#_Toc90137148)

[8 Протокол тестирования ПО 35](#_Toc90137149)

[9 Экономические затраты 35](#_Toc90137150)

[ЗАКЛЮЧЕНИЕ 36](#_Toc90137151)

[ПРИЛОЖЕНИЕ А 37](#_Toc90137152)

[СПИСОК СОКРАЩЕНИЙ 45](#_Toc90137153)

## ВВЕДЕНИЕ

По мере развития интернета, позволяющего получить доступ из любой точки земного шара к огромным информационным и развлекательным ресурсам, у пользователей появилась новая возможность яркого самовыражения, рекламы своей фирмы или демонстрации миру своих идей, стиля, новаторских технологий – создания веб-сайтов.

Любая организация, коммерческая или не коммерческая в эпоху информационного общества зависит от информационных ресурсов, важнейшим из которых сейчас является веб-сайт. Для коммерческих компаний присутствие в Интернете — это возможность рассказать о своих товарах и услугах, найти потенциальных партнеров и клиентов, а также снизить издержки за счет Интернет-торговли, использования «облачных» сервисов, для некоммерческих компаний веб-сайт дает возможность быстрее находить волонтеров, расширять сферу деятельности, при этом не затрачивая больших средств.

При выполнении курсового проекта должны быть решены следующие задачи:

* изучить предметную область;
* описать входную и выходную информацию;
* изучить концептуальную модель;
* изучить логическую модель;
* описать структуру базы данных;
* описать контрольный пример и общие требования к программному продукту.

Во втором разделе курсового проекта должны быть решены следующие задачи:

* описать программу;
* описать руководство пользователя;
* описать протокол тестирования программного продукта и меры по обеспечению защиты информации.

В заключении подведены итоги работы над проектом, обобщены выводы и рекомендации.

## 1 Описание предметной области

Проектируемая информационная система предназначена для дистанционного взаимодействия между людьми при помощи инструментов автоматизированной информационной системы "Write Me", каждый ее пользователь сможет создать свой виртуальный портрет - сформировать аккаунт, в котором они смогут указать подробные данные о себе (ФИО, дату рождения, страну проживания, город и другое). По этой информации аккаунт пользователя смогут найти другие участники информационной системы. Наличие аккаунта уже позволяет использовать механизмы поиска людей, а также общения, для знакомств и создания социальных отношений между людьми, которые имеют схожие интересы.

Интернет-продукт должен быть разграничен функционально по уровню доступа из следующего списка:

* пользователь;
* администратор.

Для пользователя, информационная система должна обеспечивать возможность выполнения перечисленных ниже функций:

* отправление жалобы на пользователя или пост;
* поиск людей;
* просмотр каталога входящих заявок;
* просмотр каталога диалогов;
* просмотр каталога друзей;
* просмотр каталога исходящих заявок
* просмотр каталога личных постов;
* просмотр личного профиля;
* просмотр новостной ленты.

Для администратора, информационная система должна обеспечивать возможность выполнения перечисленных ниже функций:

* просмотр каталога жалоб;
* рассмотрение жалоб и принятие решения по ней.
* добавление новости проект.

А также иметь все функциональные составляющие пользователя и модератора.

## 2 Техническое задание

Техническое задание распространяется на проектирование информационный системы для построения, отражения и организации социальный взаимоотношений (далее – социальная сеть).

### 2.1 Основание для разработки

Основанием для проведения разработки социальной сети является договор на разработку проекта от 14.09.2021.

Наименование работы: Социальная сеть «Write Me».

### 2.2 Назначение разработки

Проектируемая информационная система предназначена для дистанционного взаимодействия между людьми при помощи инструментов социальной сети «Write Me», каждый ее пользователь сможет создать свой виртуальный портрет - сформировать аккаунт, в котором они смогут указать подробные данные о себе (ФИО, дату рождения, страну проживания, город и другое). По этой информации аккаунт пользователя смогут найти другие участники информационной системы. Наличие аккаунта уже позволяет использовать механизмы поиска людей, а также общения, для знакомств и создания социальных отношений между людьми, которые имеют схожие интересы.

Конечными пользователями системы будут являться пользователи, модераторы и администраторы

### 2.3 Требования к программе

#### 2.3.1 Требования к функциональный характеристикам

Интернет-продукт должен быть разграничен функционально по уровню доступа из следующего списка:

* пользователь;
* администратор.

#### 2.3.2 Требования к функционалу для пользователей

Информационная система должна обеспечивать возможность выполнения перечисленных ниже функций:

* отправление жалобы на пользователя или пост;
* поиск людей;
* просмотр каталога входящих заявок;
* просмотр каталога диалогов;
* просмотр каталога друзей;
* просмотр каталога исходящих заявок
* просмотр каталога личных постов;
* просмотр личного профиля;
* просмотр новостной ленты.

#### 2.3.3 Требования к функционалу для администраторов

Информационная система должна обеспечивать возможность выполнения перечисленных ниже функций:

* добавление новости проекта;
* просмотр каталога жалоб;
* рассмотрение жалоб и принятие решения по ней.

А также иметь все функциональные составляющие пользователя.

### 2.4 Требования к надежности

Разрабатываемая информационная система должна иметь:

* разграничение пользовательских прав;
* парольную защиту пользователей;
* контроль вводимой информации.

### 2.5 Требования к составу и параметрам технический средств

Системные требования для работы веб версии программного продукта должны быть следующими:

* браузер Firefox последней версии;
* разрешение экрана не менее 1024x768;
* стабильное подключение к сети интернет.

### 2.6 Требования к информационной и программной совместимости

Десктопное приложение должно уметь работать в операционных системах Windows 7 и выше.

### 2.7 Требования к транспортированию и хранению

Программа должна поставляться в электронном виде. Программная документация поставляется в электронном и печатном виде.

### 2.8 Специальные требования

Программное обеспечение должно иметь дружественный интерфейс, рассчитанный на пользователя средней квалификации (с точки зрения компьютерной грамотности). Ввиду объемности проекта задачи предполагается решать поэтапно. При этом модули программного обеспечения (ПО), созданные в разное время, должны предполагать возможность наращивания системы и быть совместимы друг с другом; поэтому документация на принятое эксплуатационное ПО должна содержать полную информацию, необходимую для работы с ним программистов. Язык программирования определяется выбором исполнителя.

### 2.9 Требования к программной документации

Сопровождающая документация к проектируемой информационной системы должны быть выполнены в соответствии с ГОСТ 19.106-78 и ГОСТами к каждому виду документа.

Программная документация должна включать в себя

* титульный лист;
* задание на курсовой проект;
* заключение на курсовой проект;
* техническое задание (ГОСТ 19.201-78);
* пояснительную записку к курсовому проекту (ГОСТ 7.32-2001);
* презентацию.

### 2.10 Технико-экономические обоснование

В рамках проектируемой информационной системы расчёт экономической эффективности предусмотрен в соответствии с диаграммой Ганта.

### 2.11 Описание входной информации

Основным источником информации для взаимодействия между пользователями является личная информация личных аккаунтов пользователей, а также новостная лента системы.

Описание входных документов представлено в таблице 2.11.1.

#### Таблица 2.11.1 – Описание входных данных

|  |  |  |
| --- | --- | --- |
| Наименование данных | Дата поступления данных | Откуда поступает данных |
| Личная информация пользователя | При регистрации | Пользователь |
| Новости | По мере необходимости | Администратор |

### 2.12 Описание выходных данных

Выходными данными являются:

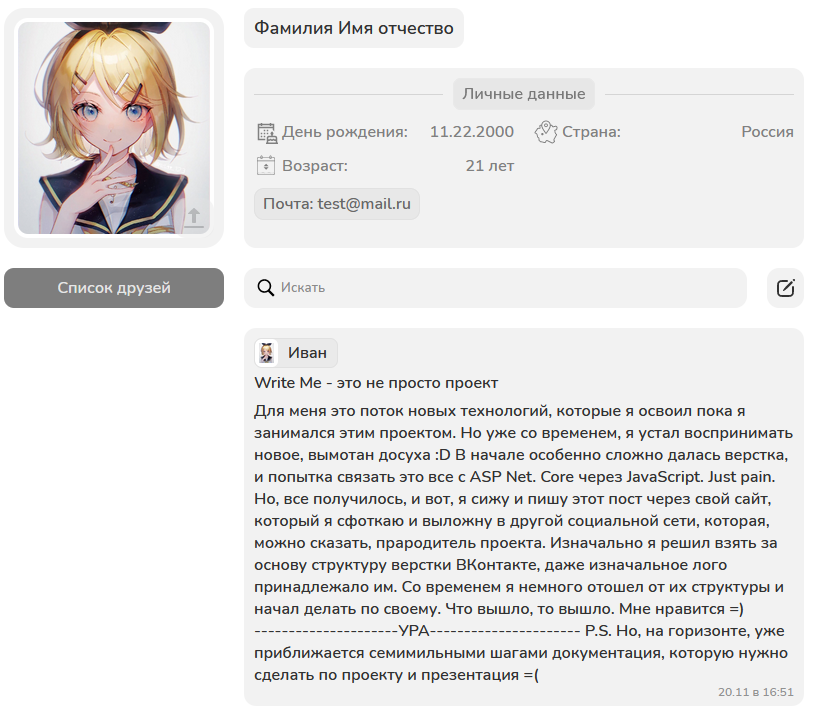
* личный профиль пользователя;
* новостная лента.

Описание выходных данных представлено в таблице 2.3.1.

#### Таблица 2.3.1 – Описание выходных данных

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Наименование данных | Периодичность выдачи данных | Кол-во экз. | Куда передаются | Поля сортировки | Поля группировки | Итоги |
| Личный профиль | Независимо от времени | 1 | Пользователю | - | - | - |
| Новостная лента | Независимо от времени | 1 | Пользователю | - | - | - |

На рисунке 2.3.1 представлен выходной документ «Личный профиль».

Рисунок 2.3.1 – «Личный профиль»

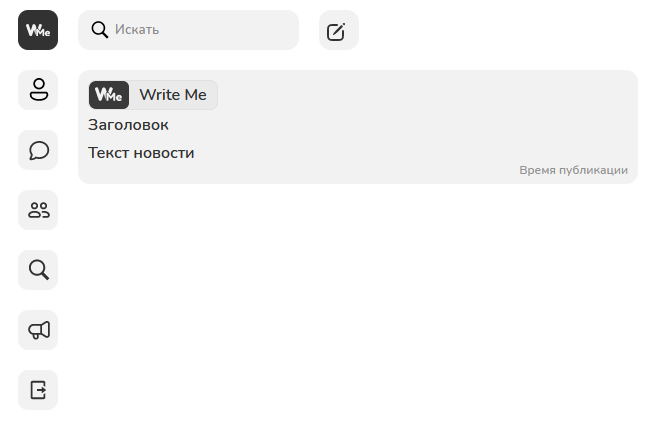


Рисунок 2.3.2 – «Новостная лента»

## 3 Диаграмма Ганта

Диаграмма Ганта используется для иллюстрации плана, графика работ по проекту.

Диаграмма Ганта изображена на рисунках 3.0.1 и 3.0.2

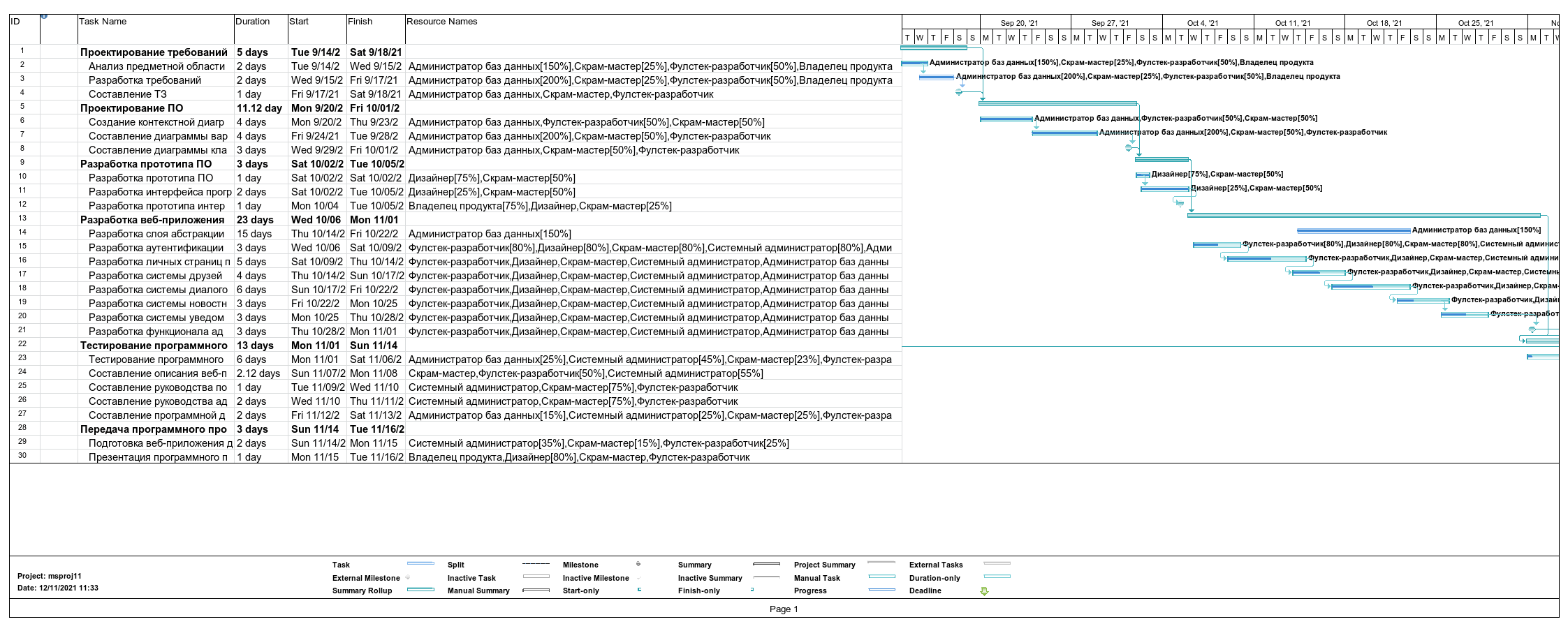


Рисунок 3.0.1 – Диаграмма Ганта (сентябрь-октябрь)

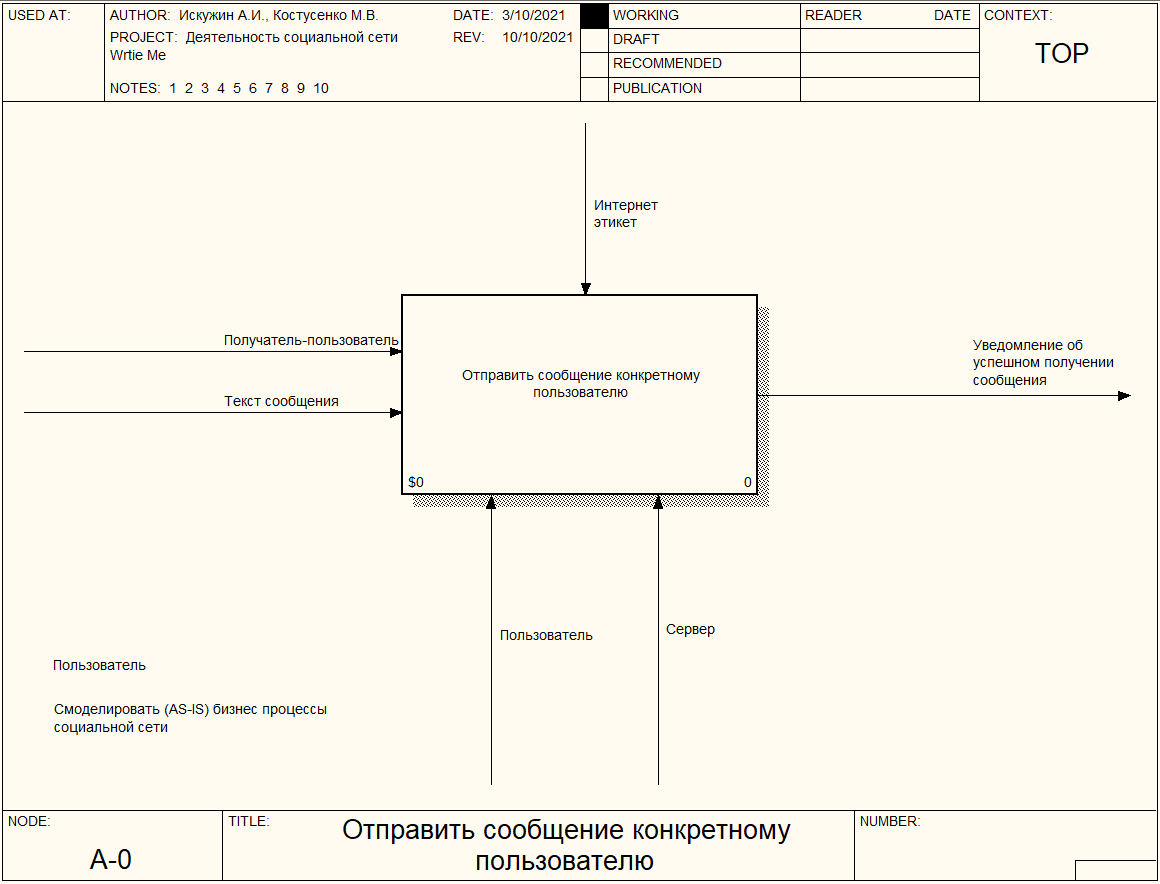


Рисунок 3.0.2 – Диаграмма Ганта (ноябрь-декабрь)

## 4 Проектирование ПО

Контекстная диаграмма представляет собой самое общее описание системы и ее взаимодействие с внешней средой.

### 4.1 Контекстная диаграмма и декомпозиция 1-го уровня

Рисунок 4.1.1 – Контекстная диаграмма

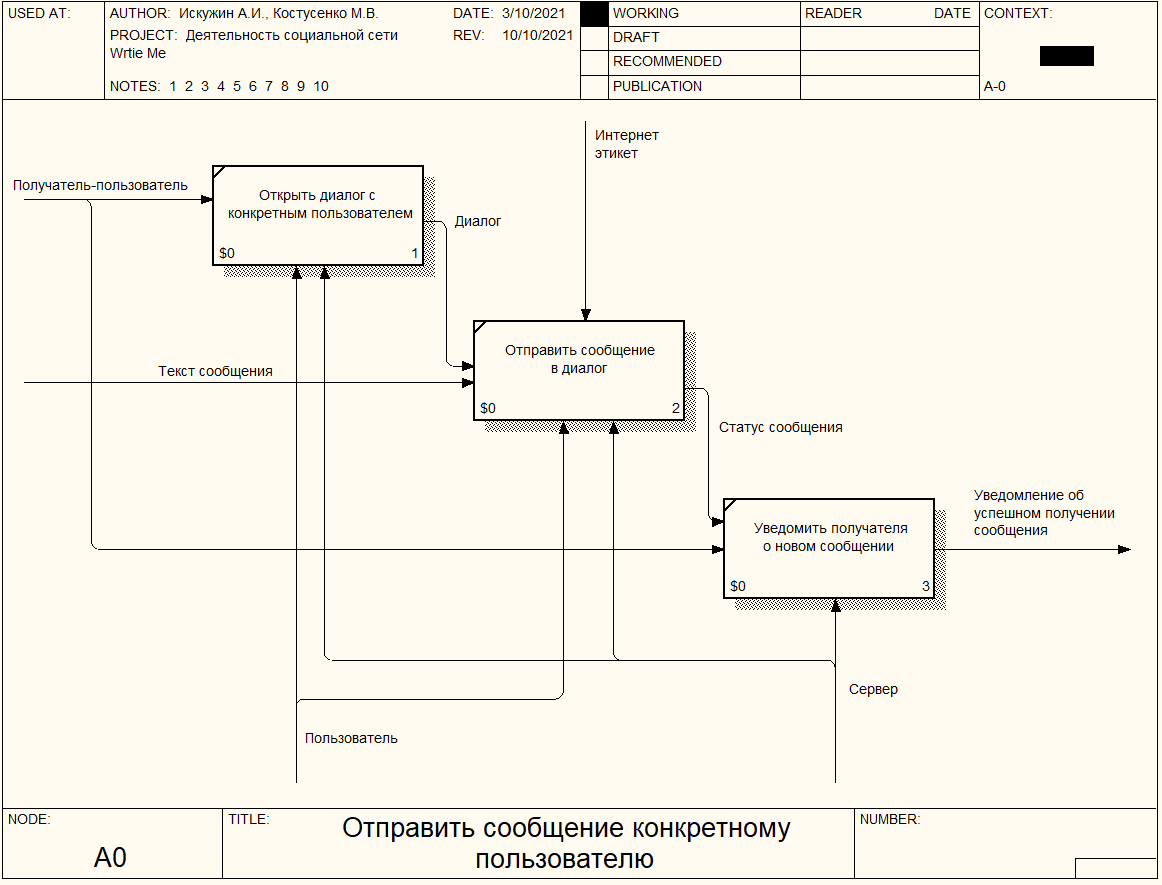


Рисунок 4.1.2 – Декомпозиция 1-го уровня

### 4.2 Диаграмма вариантов использования

Концептуальная модель – это модель предметной области. Компонентами модели являются объекты и взаимосвязи. Концептуальная модель включает описание объектов и их взаимосвязей, представляющих интерес в рассматриваемой предметной области. Взаимосвязи между объектами являются частью концептуальной модели и должны отображаться в базе данных. Взаимосвязь может охватывать любое число объектов.

Основными элементами модели являются сущности, связи между ними и их свойства (атрибуты).

Сущность – это класс однотипных объектов, информация о которых должна быть учтена в модели.

Каждая сущность должна иметь наименование, выраженное существительным в единственном числе. Каждая сущность в модели изображается в виде прямоугольника с наименованием.

У сущностей выделяются ключевые атрибуты – ключ сущности – это один или более атрибутов, уникально определяющих данную сущность.

Вместе эти элементы образует диаграмму вариантов использования в соответствие с приложением А1.

### 4.3 Диаграмма классов

Диаграмма классов демонстрирует общую иерархию классов системы, их коопераций, атрибутов, методов, интерфейсов и взаимосвязей между ними.

Диаграмма классов представлена в приложениях А2-А7.

## 5 Описание структуры базы данных

Для работы с базами данных используется СУБД MySQL

Для работы с СУБД MySQL в режиме графического пользовательского интерфейса используется среда визуального проектирования баз данных MySQL Workbench 8.0 CE.

Для работы с СУБД MySQL из программного кода используется набор технологий Entity Framework Core.

Таблица 5.0.1 – Описание структуры базы данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Физическое имя таблицы (Логическое имя таблицы) | | | | |
| countries (страны) | | | | |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа[[1]](#footnote-1) |
| Id | Код страны | INT | 4 | PK |
| CountryCode | Идентификационный код страны | INT | 4 | - |
| Продолжение таблицы 5.0.1 | | | | |
| Name | Наименование страны | VARCHAR | 256 | - |
| roles (роли) | | | | |
| Id | Код роли | INT | 4 | PK |
| Name | Наименование роди | VARCHAR | 256 | - |
| Code | Короткий код | LONGTEXT | <4294967295 | - |
| users (пользователи) | | | | |
| Id | Код пользователя | INT | 4 | PK |
| MaillAddress | Почта пользователя | LONGTEXT | <4294967295 | - |
| Password | Пароль пользователя | LONGTEXT | <4294967295 | - |
| RoleId | Код роли | INT | 4 | FK |
| IsNew | Статус нового пользователя | TINYINT | 1 | - |
| IsDeleted | Статус удаленного пользователя | TINYINT | 1 | - |
| CreatedDateTime | Штамп создания профиля | DATETIME | 8 | - |
| UpdatedDateTime | Штамп обновления профиля | DATETIME | 8 | - |
| Name | Имя пользователя | LONGTEXT | <4294967295 | - |
| Surname | Фамилия пользователя | LONGTEXT | <4294967295 | - |
| Patronymic | Отчество пользователя | LONGTEXT | <4294967295 | - |
| Birthday | Дата рождения пользователя | DATETIME | 8 | - |
| AvatarPath | Путь к иконке профиля | LONGTEXT | <4294967295 | - |
| CountryId | Код страны | INT | 4 | - |
| chatParticipants (Участники чата) | | | | |
| Продолжение таблицы 5.0.1 | | | | |
| Id | Код участника | INT | 4 | PK |
| ChatId | Код чата | INT | 4 | FK |
| UserId | Код пользователя | INT | 4 | FK |
| CreatedDateTime | Штамп создания/вхождения в чат | DATETIME | 8 | - |
| LeftDateTime | Штамп выхода из чата | DATETIME | 8 | - |
| chats (чаты) | | | | |
| Id | Код чата | INT | 4 | PK |
| Title | Заголовок чата | LONGTEXT | <4294967295 | - |
| MaximumChatParticipants | Максимальное количество участников в чате | INT | 4 | - |
| IsPrivateChat | Является сущность приватным чатом | TINYINT | 1 | - |
| CreatedDateTime | Штамп создания чата | DATETIME | 8 | - |
| UpdatedDateTime | Штамп обновления чата | DATETIME | 8 | - |
| generatedChatMessaged (сгенерированные сообщения чата) | | | | |
| Id | Код сообщения | INT | 4 | PK |
| ChatId | Код чата | INT | 4 | FK |
| Text | Текст сообщения | LONGTEXT | <4294967295 | - |
| CreatedDateTime | Штамп создания сообщения | DATETIME | 8 | - |
| participantChatMessages (сообщения участников чата) | | | | |
| Id | Код сообщения | INT | 4 | PK |
| ChatParticipantSenderId | Код отправителя, участника чата | INT | 4 | FK |
| ChatId | Код чата | INT | 4 | FK |
| Text | Текст сообщения | LONGTEXT | <4294967295 | - |
| Продолжение таблицы 5.0.1 | | | | |
| CreatedDateTime | Штамп отправки сообщения | DATETIME | 8 | - |
| UpdatedDateTime | Штамп изменения сообщения | DATETIME | 8 | - |
| userPosts (посты пользователей | | | | |
| Id | Код поста | INT | 4 | PK |
| OwnerId | Код пользователя, создателя | INT | 4 | FK |
| Title | Заголовок поста | LONGTEXT | <4294967295 | - |
| Description | Текст поста | LONGTEXT | <4294967295 | - |
| CreatedDateTime | Штамп создания поста | DATETIME | 8 | - |
| UpdatedDateTime | Штамп обновления поста | DATETIME | 8 | - |
| systemPosts (системные посты) | | | | |
| Id | Код поста | INT | 4 | PK |
| Title | Заголовок поста | LONGTEXT | <4294967295 | - |
| Description | Текст поста | LONGTEXT | <4294967295 | - |
| CreatedDateTime | Штамп создания поста | DATETIME | 8 | - |
| UpdatedDateTime | Штамп обновления поста | DATETIME | 8 | - |
| postReports (Жалобы на посты) | | | | |
| Id | Код жалобы | INT | 4 | PK |
| PostId | Код поста | INT | 4 | FK |
| SenderId | Код отправителя | INT | 4 | FK |
| Commentary | Комментарий к жалобе | LONGTEXT | <4294967295 | - |
| ReportTypeId | Код типа жалобы | INT | 4 | FK |
| ReportStateId | Состояния жалобы | INT | 4 | FK |
| CreatedDateTime | Дата поступления жалобы | DATETIME | 8 | - |
| Продолжение таблицы 5.0.1 | | | | |
| UpdatedDateTime | Дата изменения жалобы | DATETIME | 8 | - |
| reportsStates (Состояния жалоб) | | | | |
| Id | Код состояния жалобы | INT | 4 | PK |
| Name | Наименование состояния | VARCHAR | 256 | - |
| reportTypes (Типы жалоб) | | | | |
| Id | Код типа жалобы | INT | 4 | PK |
| Name | Наименование типа жалобы | VARCHAR | 256 | - |
| friendshipApplications (Дружественные связи) | | | | |
| Id | Код связи | INT | 4 | PK |
| UserOneID | Код первого пользователя | INT | 4 | FK |
| UserTwoId | Код второго пользователя | INT | 4 | FK |
| ApplicationStateUserOne | Состояние заявки первого пользователя | TINYINT | 1 | - |
| ApplicationStateUserTwo | Состояние заявки второго пользователя | TINYINT | 1 | - |
| UserOneFrendshipTypeId | Код типа типа связи первого пользователя | INT | 4 | FK |
| UserTwoFriendshipTypeID | Код типа связи первого пользователя | INT | 4 | FK |

Версия концептуальной модели, которая может быть обеспечена конкретной СУБД, называется логической моделью. Процесс построения логической модели базы данных должен опираться на определённую модель данных (реляционная, сетевая, иерархическая). В нашем случае информационная система создаётся в среде Workbench 8.0 CE и будет представлять собой реляционную базу данных.

В реляционных моделях данных объекты и взаимосвязи между ними представляются с помощью таблиц.

Таблица в реляционной модели называется отношением.

Атрибут (поле) – любой столбец в таблице.

Кортежи (записи) – строки таблицы.

Таблицы связаны между собой при помощи ключевых полей.

Ключ – это поле, позволяющее однозначно идентифицировать запись в таблице. Ключ может быть простым (состоит из одного поля) или составным (из нескольких полей).

В реляционных базах данных логическое проектирование приводит к разработке схемы данных в соответствии приложением А8.

## 6 Описание программы

Описание программы содержит: описание контроллеров, сервисов, классов, методов.

Описание программы представлено на таблице 6.0.1.

Таблица 6.0.1 – Описание модулей и процедур программы

|  |  |
| --- | --- |
| Процедуры | Назначение |
| Сервис AuthenticateService – сервис аутентификации | |
| ConfirmUserAsync | Авторизация пользователя |
| RegisterAsync | Регистрация пользователя |
| IsUserExist | Проверка существования пользователя |
| GetViewModel (User) | Получить сущность пользователя в виде модели представления |
| Сервис FileService – сервис работы с файлами | |
| Продолжение таблицы 6.0.1 | |
| Upload | Загрузить фото |
| Сервис FriendsService – сервис взаимотношений | |
| GetViewModel (User) | Получить сущность пользователя в виде модели представления |
| GetViewModel (FriendshipApplication) | Получить сущность связи отношения между двумя пользователями в виде модели представления |
| GetFriendViewModel | Получить сущность пользователя в виде модели представления |
| GetFriendFromApplication | Получить сущность друга из связи отношений в виде модели представления |
| GetUserFriends (int) | Получить список друзей указанного пользователя |
| GetUserFriends (int, string) | Получить список друзей указанного пользователя, соответствующие фильтру |
| GetIncomingApplications | Получить список входящих заявок на дружбу указанного пользователя |
| GetOutgoingApplications | Получить список исходящих заявок на дружбу указанного пользователя |
| TryRemoveOutgoingFriendship | Отменить исходящую заявку на дружбу |
| TryRemoveFriendship | Удалить пользователя из друзей |
| TryResponseIncomingFriendship | Изменить статус входящей заявки |
| TryAllowIncomingFriendship | Принять входящую заявку |
| TryDenyIncomingFriendship | Отказать во входящей заявке |
| TrySendFriendship | Отправить заявку на дружбу |
| Сервис MessengerService – сервис мессенджер | |
| Продолжение таблицы 6.0.1 | |
| NewMessagesOnChat | Событие, реагирует на новые сообщения в чате |
| GetUserChatsPreviews | Получить список моделей представления превью чатов |
| GetPrivateChatHistory | Получить историю указанного чата |
| GetChat | Получить модель представления указанного чата |
| GetPrivateChatWithUser | Получить приватный чат с указанным пользователем |
| GetNewChatWithUser | Создать приватный чат с указанным пользователем |
| SendMessageToChat | Отправить сообщение в чат |
| GetNewMessagesFromLast | Получить новые сообщения после указанного сообщения |
| GetChatParticipantsIds | Получить список участников чата |
| GetChatViewModel | Получить сущность чата в виде модели представления |
| Сервис PostingService – сервис записей | |
| GetSystemPostViewModel | Получить сущность записи системы в виде модели представления |
| IsAdmin | Является ли указанный пользователь админом |
| GetSystemPosts | Получить список системных записей |
| GetSystemPostsWithFilter | Получить список системных записей, соответствующих фильтру |
| UploadPost | Выложить запись |
| EditPost | Отредактировать запись |
| RemovePost | Удалить запись |
| SendReportToPost | Отправить жалобу на пост |
| GetPostsReports | Получить список жалоб на посты |
| Продолжение таблицы 6.0.1 | |
| GetReportTypes | Получить список вариантов жалоб на пост |
| CloseReport | Закрыть жалобу |
| CloseReportAndDeletePost | Закрыть жалобу и удалить пост |
| Сервис ProfileService – сервис профиля | |
| GetUsersPostsWithFilter | Получить список постов указанного пользователя, соответствующие фильтру |
| GetUsersPosts | Получить список постов указанного пользователя |
| GetUserAsync | Получить пользователя |
| UploadPost | Выложить запись |
| EditPost | Отредактировать запись |
| RemovePost | Удалить запись |
| GetViewModel (UserPost) | Получить модель представления записи пользователя |
| GetViewModel (User) | Получить модель представления пользователя |

### 6.1 Назначение программы

Веб-сайт «Write Me» работает в браузере Firefox. Основными функциями веб-сайта «Write Me» являются создание дружественных связей между пользователями и общение между ними посредством отправления сообщений в диалоги.

### 6.2 Условия выполнения программы

Для возможности использования веб-сайта требуется:

* операционная система Windows 7 или новее;
* процессор Pentium 4 или новее, с поддержкой SSE2;
* оперативная память 2048 Мб или более;
* дисковое пространство не менее 200 Мб.

### 6.3 Выполнение программы

Для выполнения веб-приложения необходимо запустить браузер и перейти по ссылке localhost.

Откроется окно в соответствии с рисунком 6.3.1.

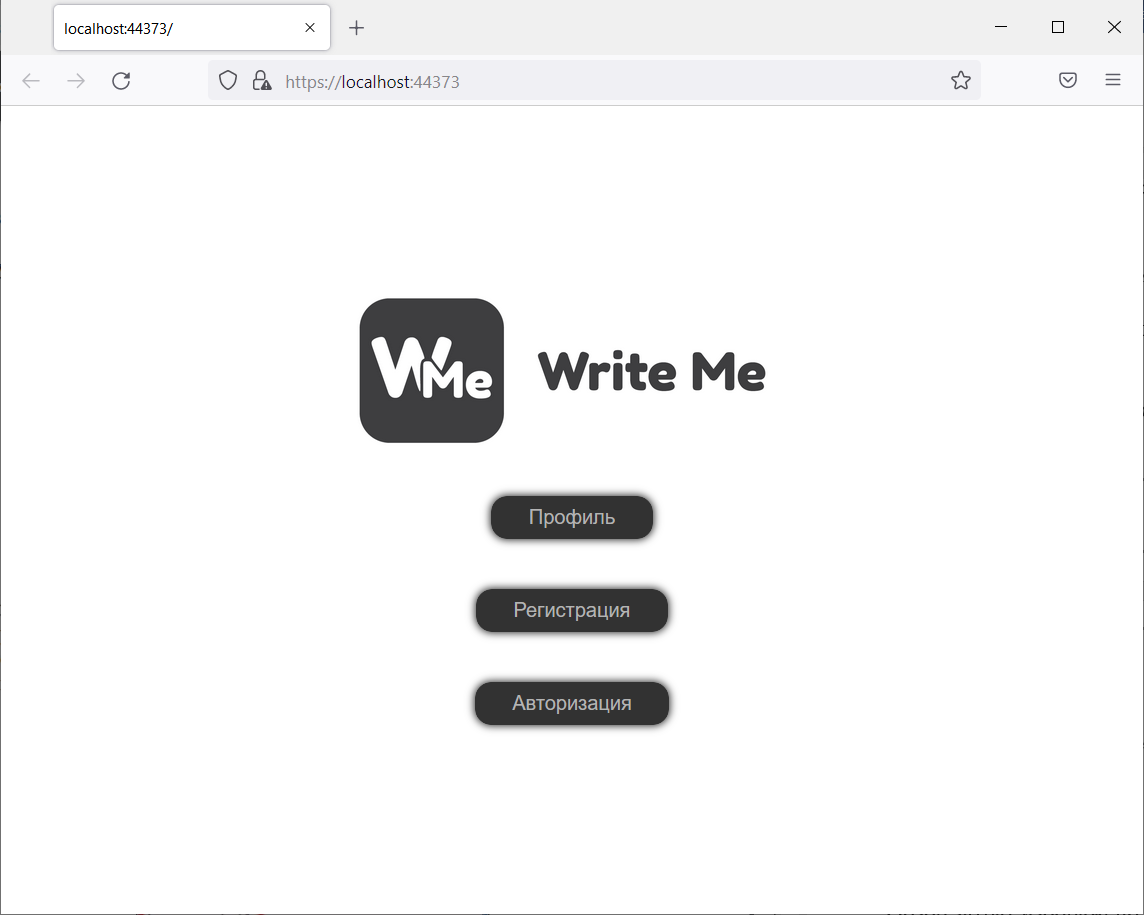


Рисунок 6.3.1 – Окно «Приветствие»

В появившемся окне можно перейти к авторизации и регистрации в соответствии с рисунками 6.3.2 и 6.3.3.

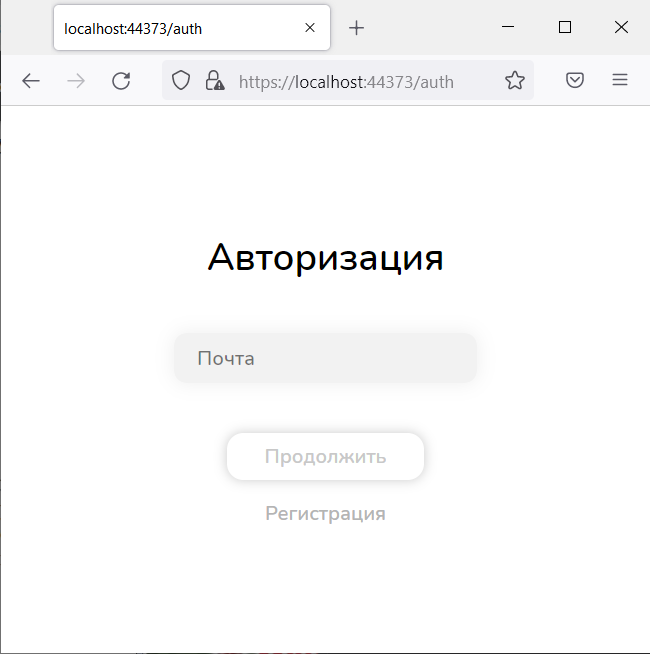


Рисунок 6.3.2 – Окно «Авторизация»

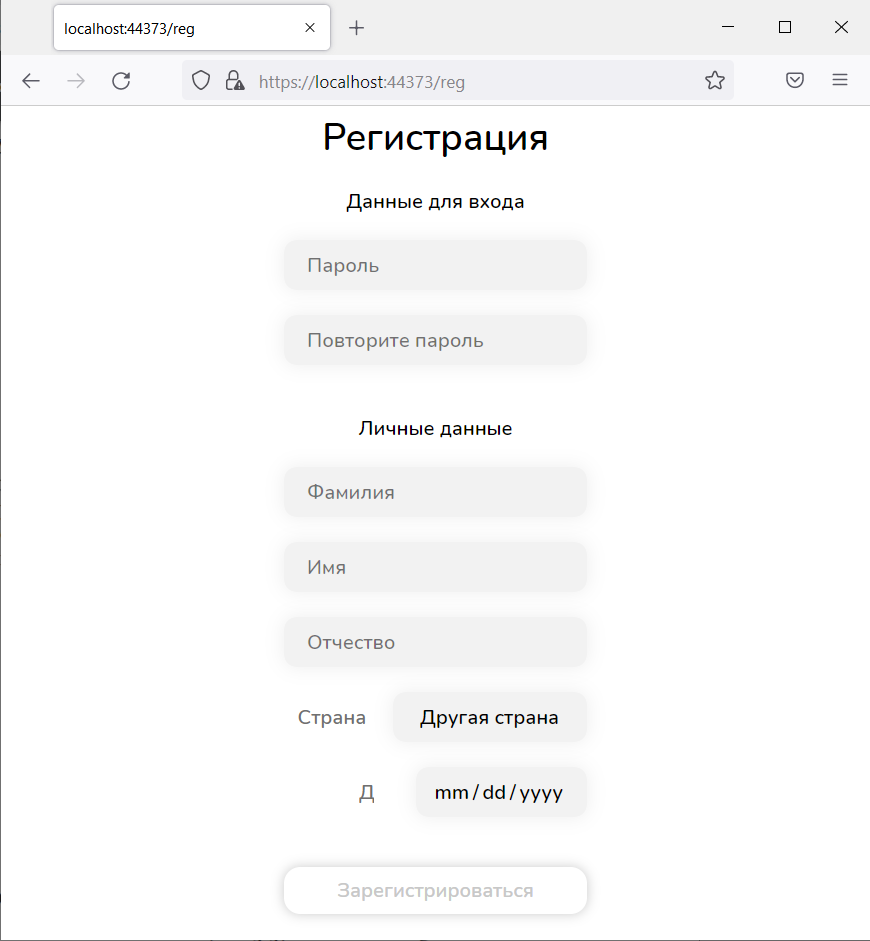


Рисунок 6.3.3 – Окно «Регистрация»

После того как пользователь зарегистрируется или авторизуется откроется окно в соответствии с рисунком 6.3.4.

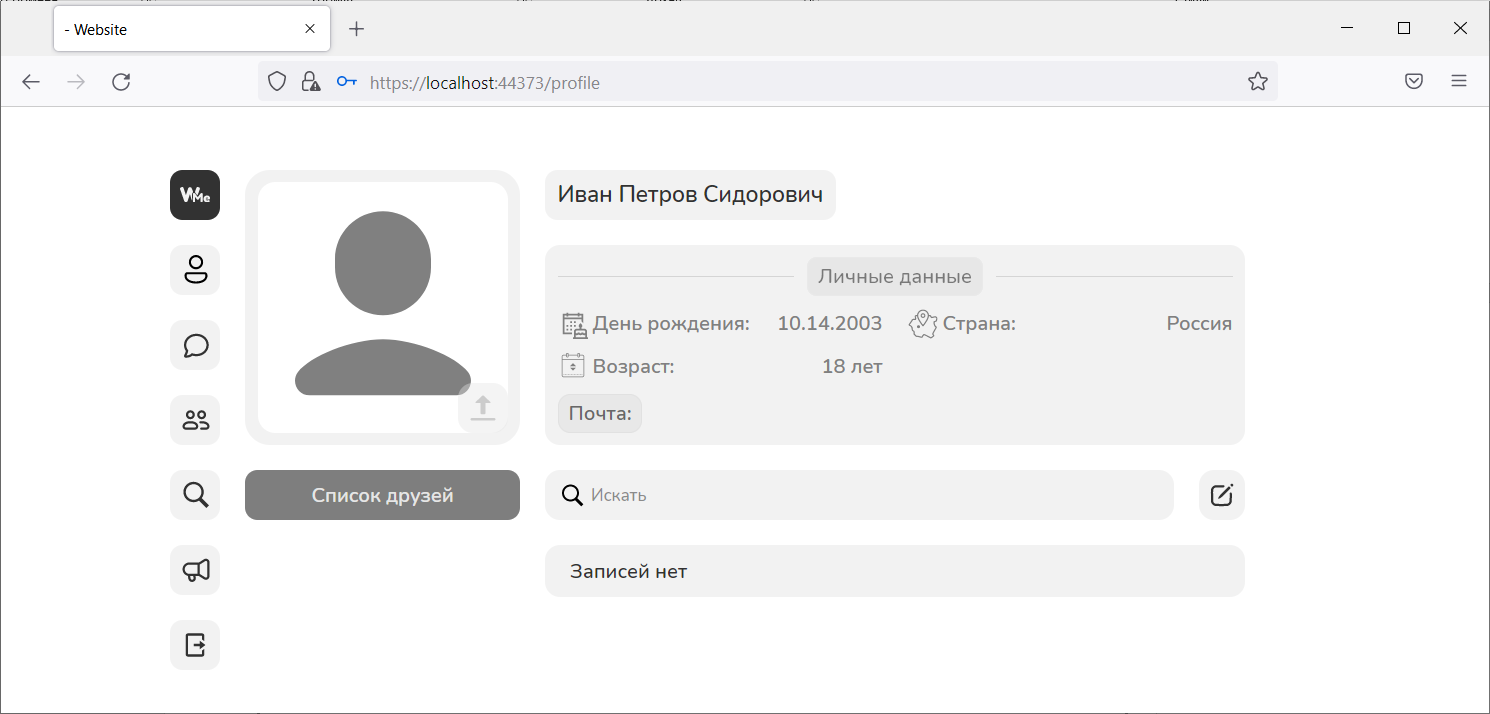


Рисунок 6.3.4 – Окно «Профиль»

В профиле пользователя отображается его личная информация, посты и боковое меню навигации.

У пользователя есть возможность добавлять посты в соответствии с рисунком 6.3.5.

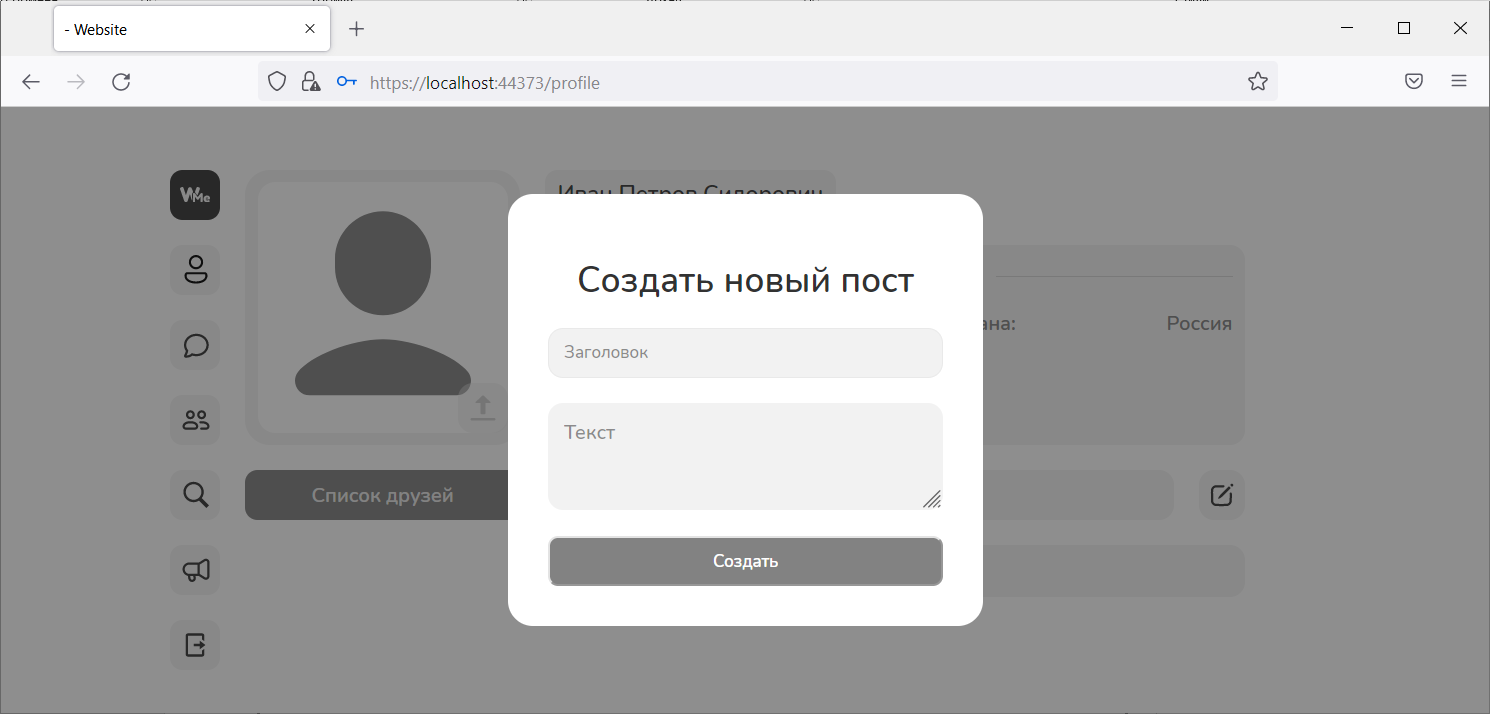


Рисунок 6.3.5 – Форма добавления поста в профиль

Боковое меню навигации позволяет переместиться в окна «Сообщения», «Друзья», «Пользователи», «Новости проекта» и выйти из авторизованного аккаунта.

Окно «Сообщения», изображенное на рисунке 6.3.6, позволяет выбрать диалог и перейти к окну в соответствии с рисунком 6.3.7.

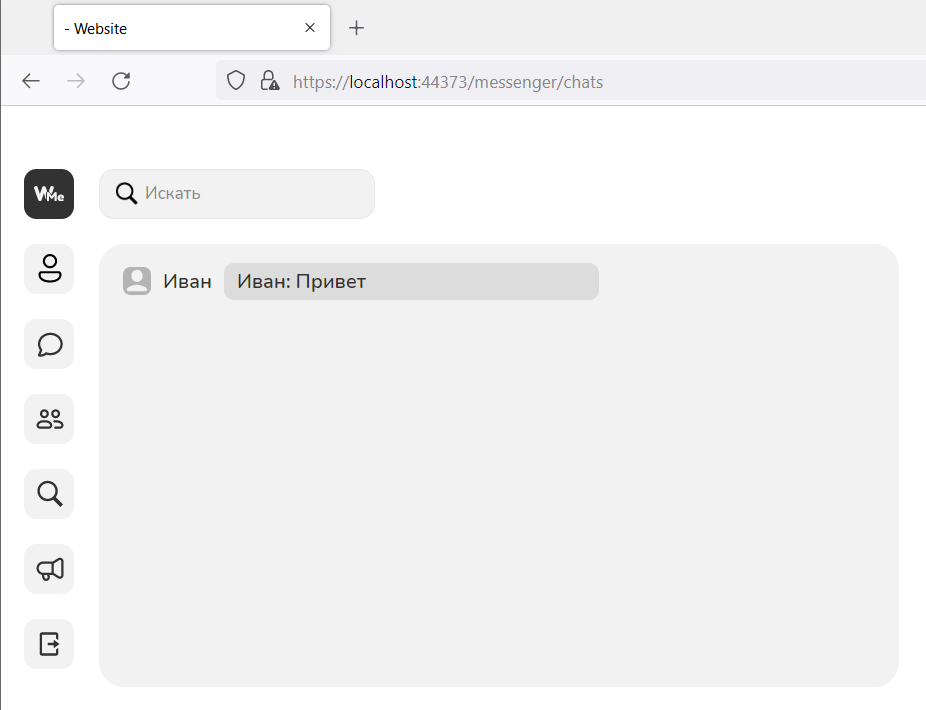


Рисунок 6.3.6 – Окно «Сообщения»

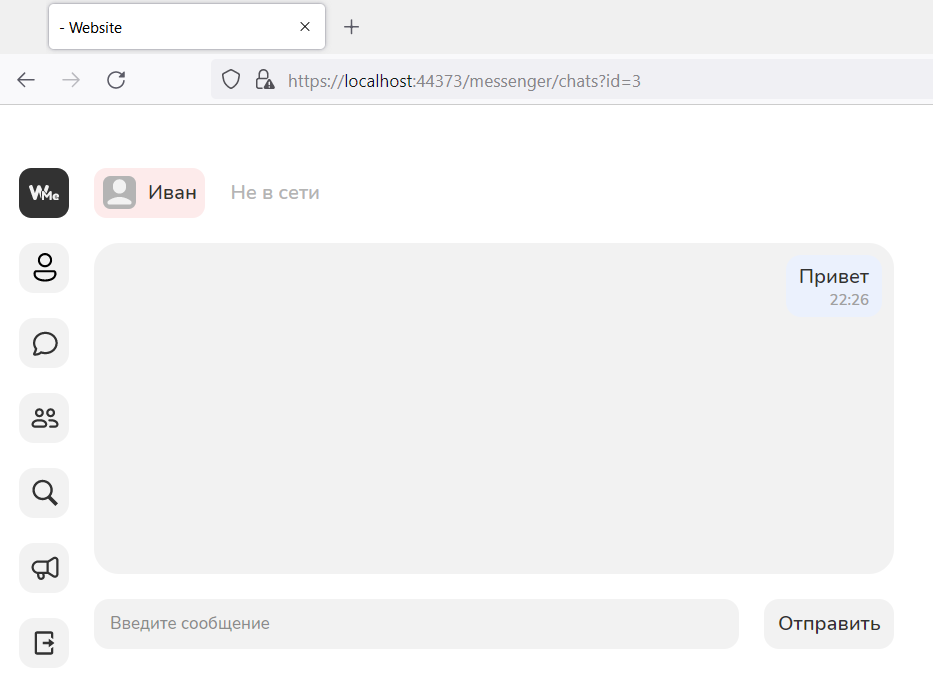


Рисунок 6.3.7 – Окно «Диалог»

Окно «Друзья» отображает друзей, исходящие или входящие заявки на дружбу, а также позволяет принимать, отклонять или отменять их, в соответствии с рисунком 6.3.8.

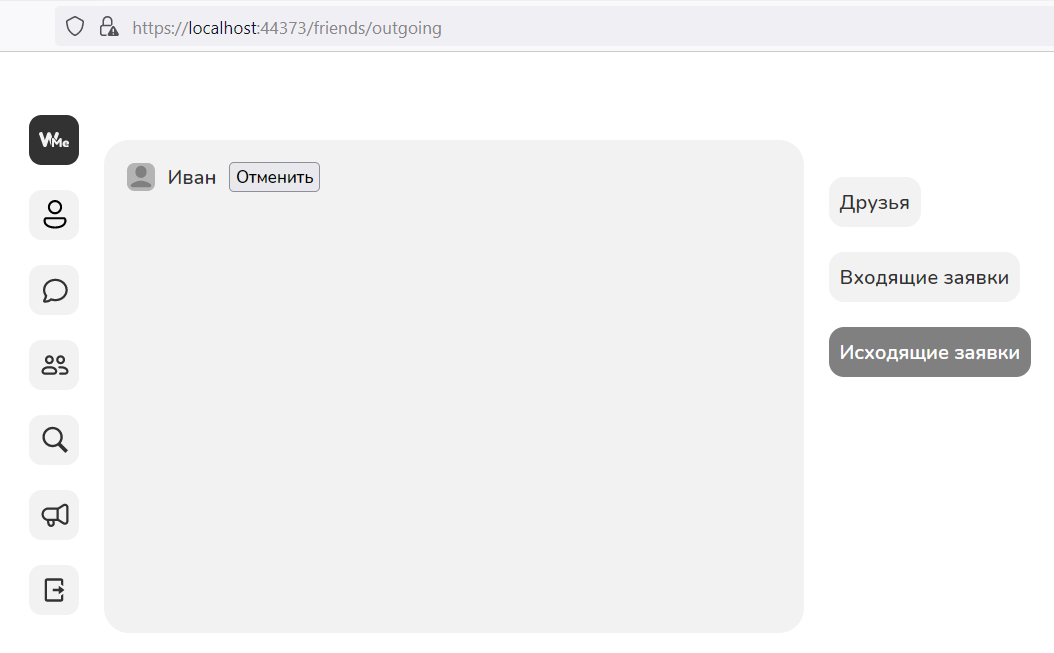


Рисунок 6.3.8 – Окно «Друзья»

Окно «Пользователи» отображает всех пользователей системы в соответствии с рисунком 6.3.9.

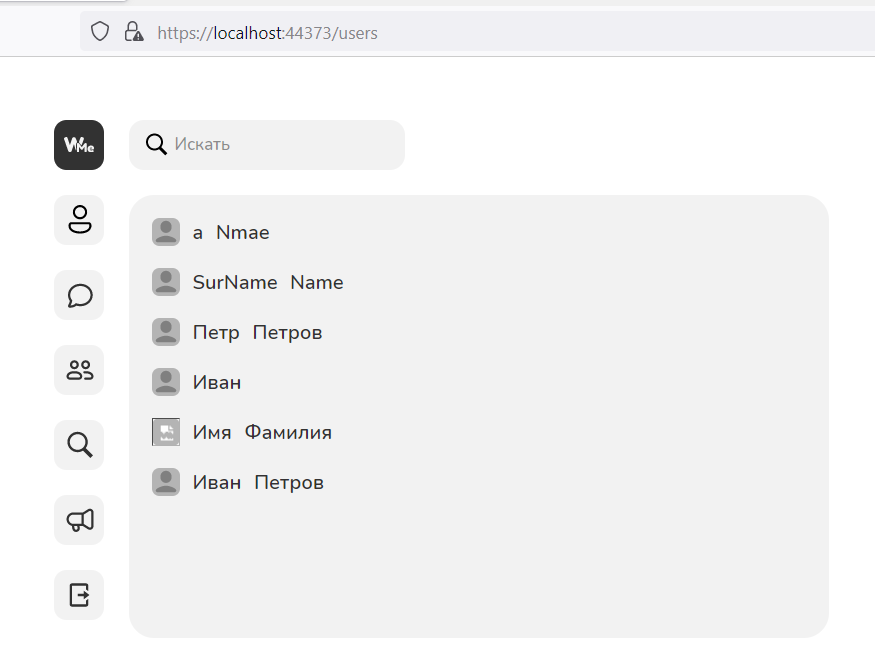


Рисунок 6.3.9 – Окно «Пользователи»

Окно «Новости проекта», изображенное на рисунке 6.3.10 отображает посты, сделанные модераторами проекта.

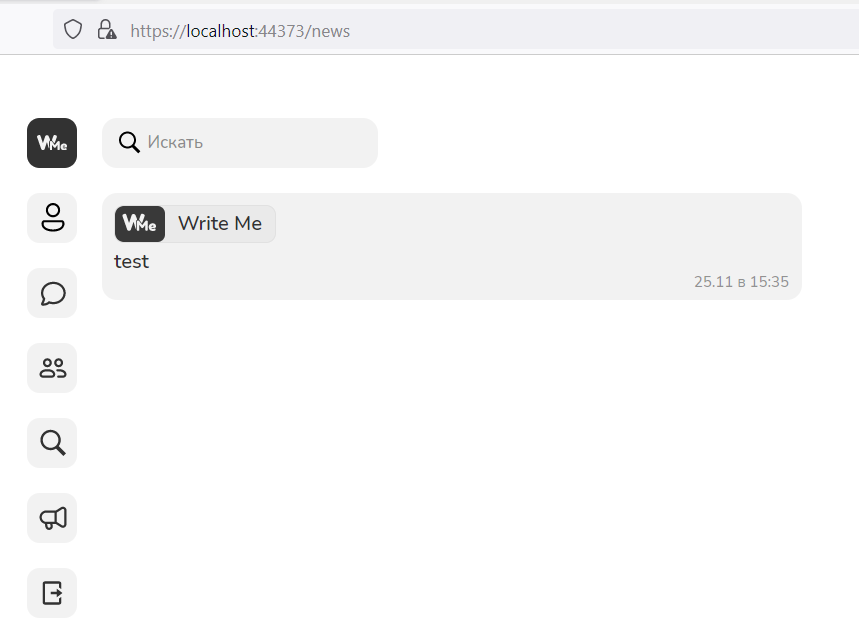


Рисунок 6.3.10 – Окно «Новости проекта»

Кнопка «Выйти» изображенная на рисунке 6.3.11 выполнит выход из авторизованного аккаунта и откроет окно «Приветствие», изображенное на рисунке 6.3.1.



Рисунок 6.3.11 – Кнопка «Выйти»

## 7 Руководство пользователя

Для начала работы необходимо открыть браузер и в нем открыть «Write Me».

После запуска веб-приложения появится приветственное окно. Для перехода в окно личного профиля необходимо авторизоваться или зарегистрироваться. Для регистрации нужно нажать кнопку «Регистрация».

Откроется окно «Регистрация», изображенное на рисунке 7.0.1, с полем для ввода почты. Почта должна быть не короче 2х символов до знака «@» и иметь хотя бы по одному знаку после с разделителем в виде точки. Для продолжения регистрации нужно нажать кнопку «Продолжить».

В соответствии с рисунком 7.0.2, откроется окно «Регистрация» с полями для ввода данных для входа и личных данных. После введения данных и нажатия на кнопку «Зарегистрироваться» откроется окно «Профиль», изображенное на рисунке 7.0.3.

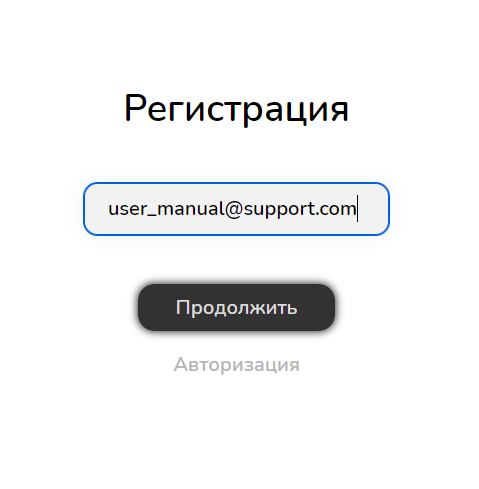


Рисунок 7.0.1 – Окно «Регистрация» для ввода почты

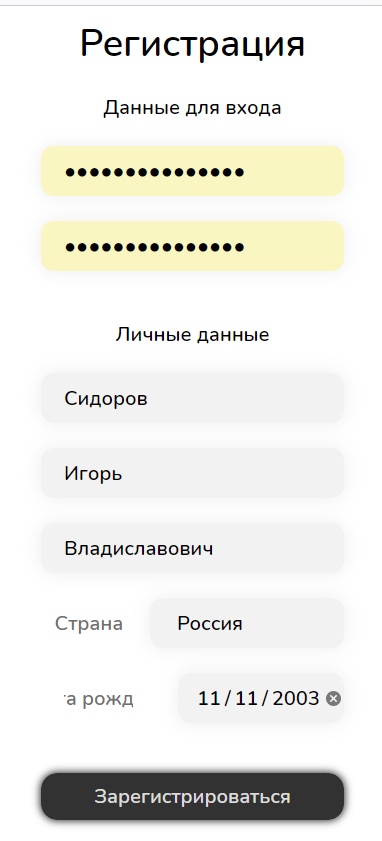


Рисунок 7.0.2 – Окно «Регистрация» для заполнения данных для входа и личных данных

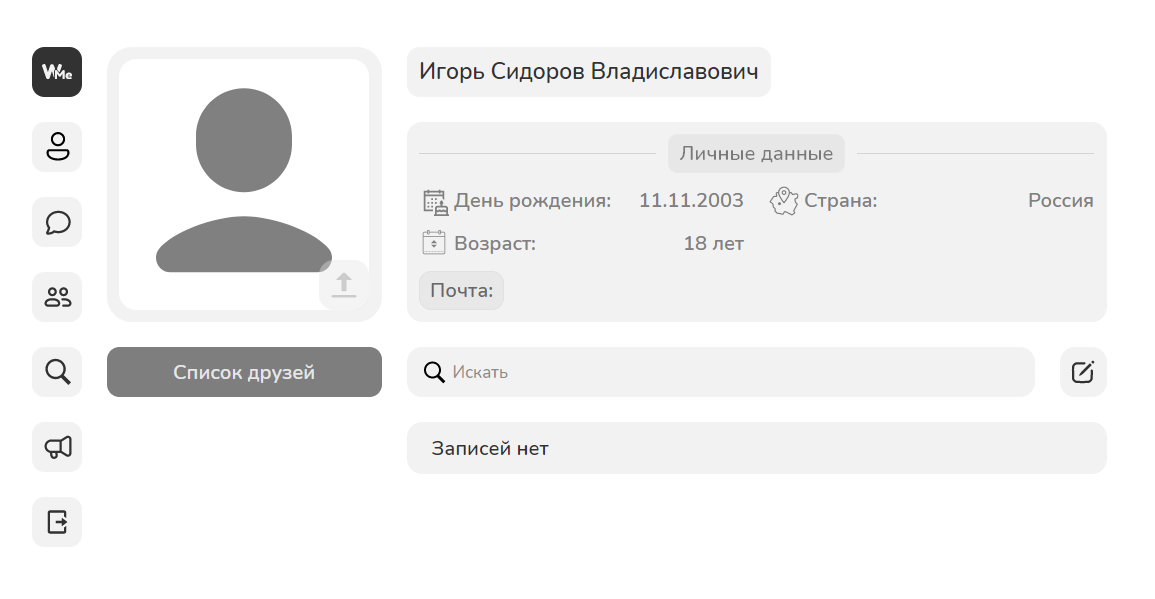


Рисунок 7.0.3 – Окно «Профиль»

## 8 Протокол тестирования ПО

## 9 Экономические затраты

Обзор затрат изображено на рисунке 9.0.1

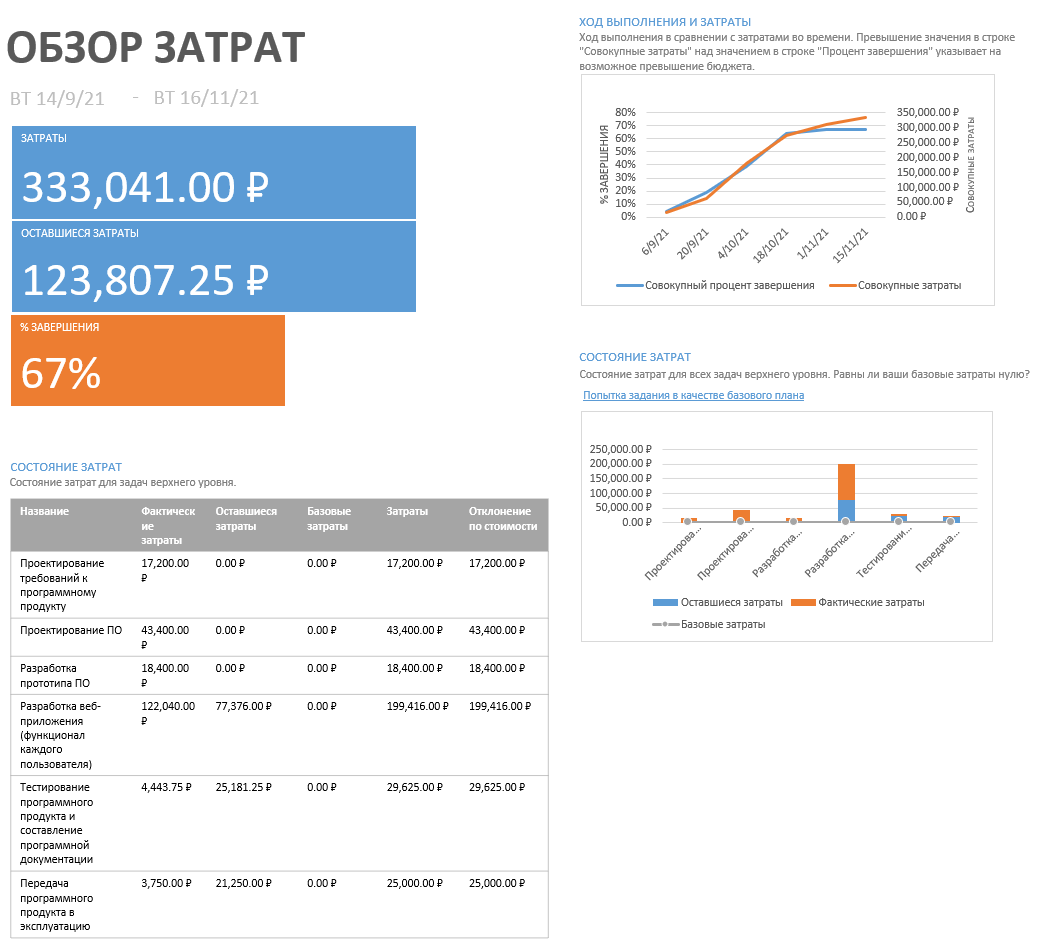


Рисунок 9.0.1 – экономические затраты

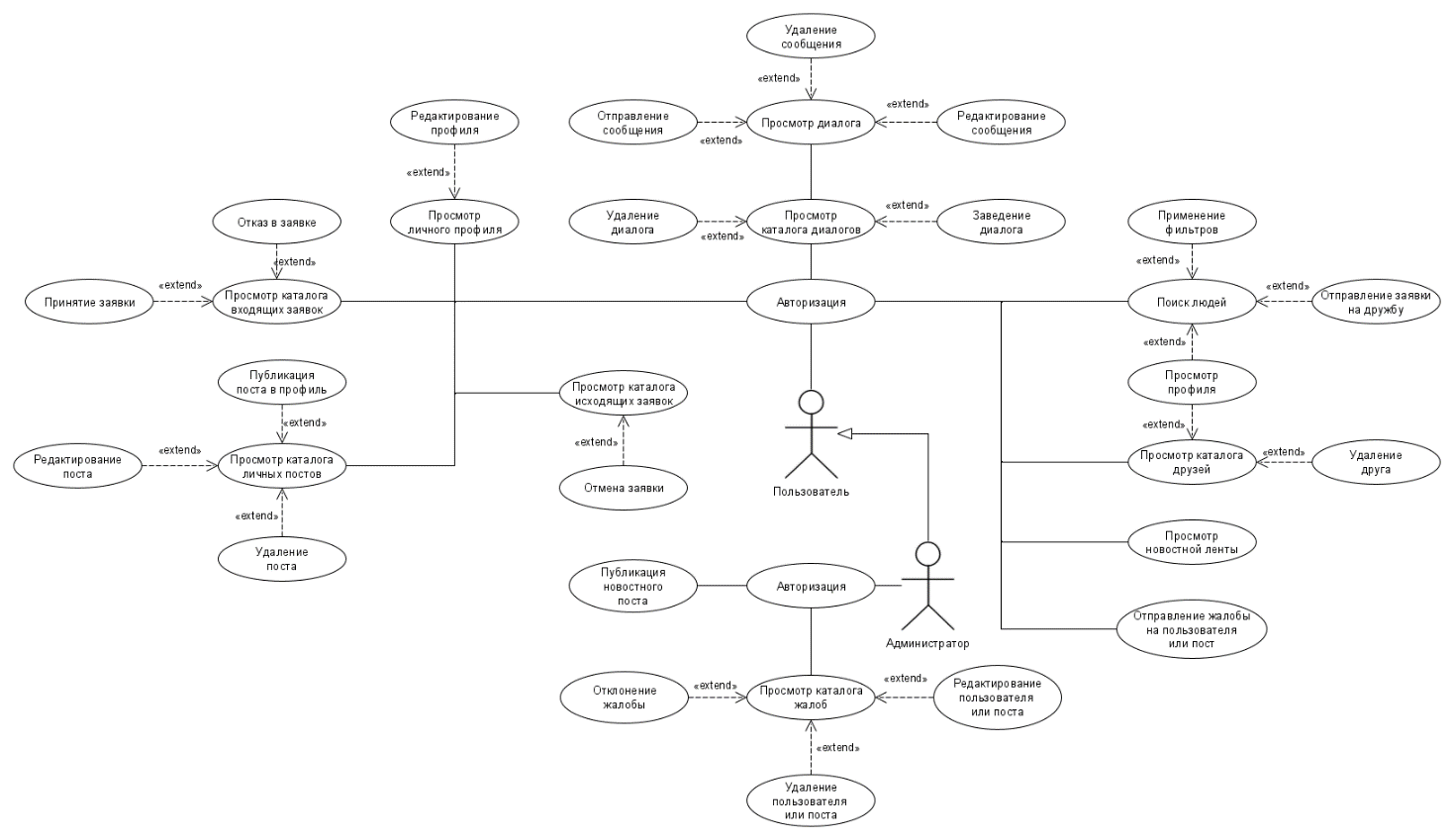
# ЗАКЛЮЧЕНИЕ

В процессе выполнения курсового проекта были разработаны структура и алгоритм работы приложения «Write Me». При этом были изучены особенности реализации компонентов информационной системы. Результатом работы стало создание программного продукта.

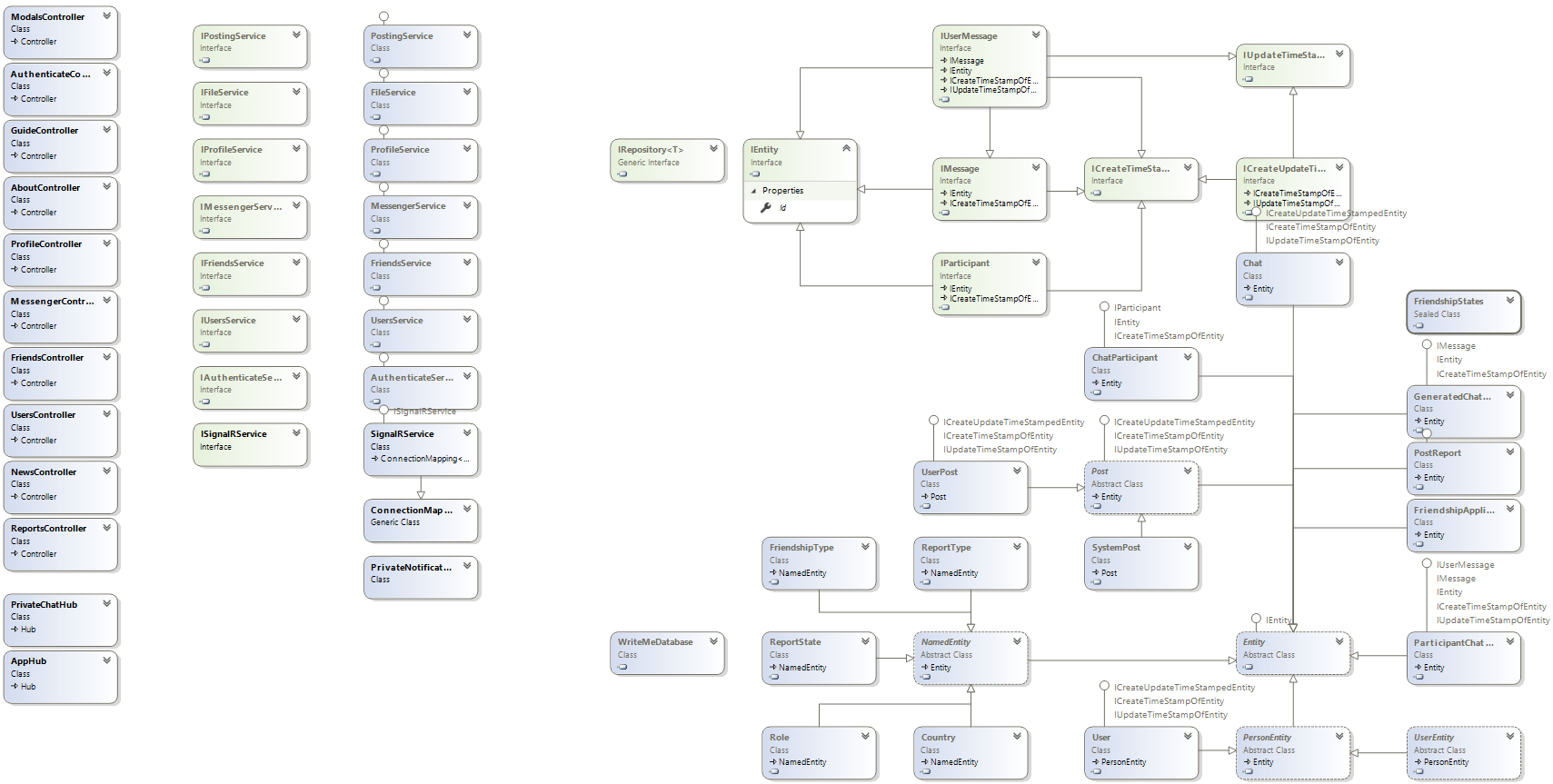
Программное обеспечение написано на языке C# с использованием среды Visual Studio 2019 и СУБД MY SQL SERVER 8.0.

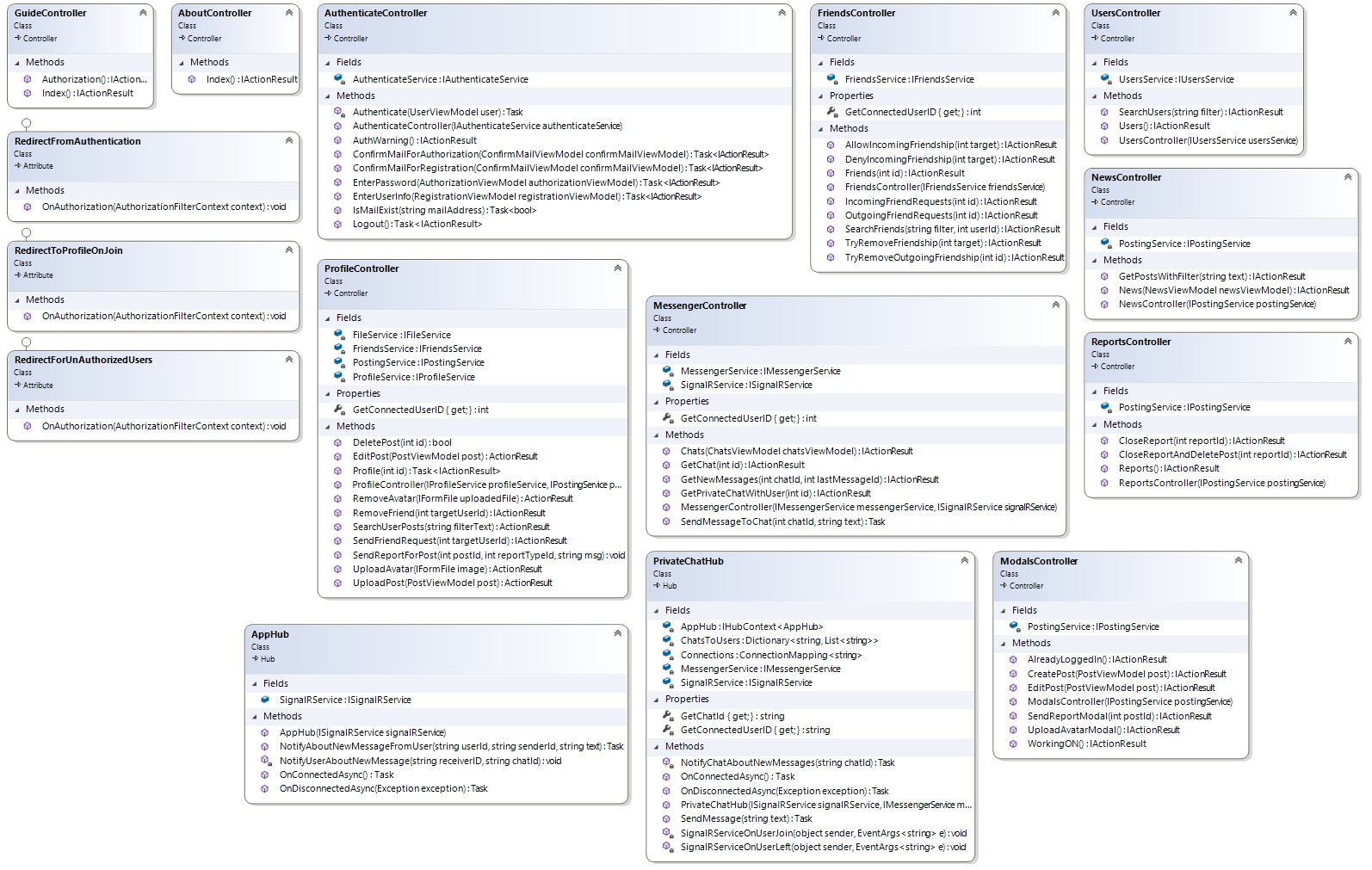
Была проведена эксплуатация и отладочное тестирование программного обеспечения. По результатам отладочного тестирования были устранены некоторые недостатки, в частности были обнаружены и исправлены неточности в реализации алгоритма. После этого было написано руководство пользователя.

# ПРИЛОЖЕНИЕ А

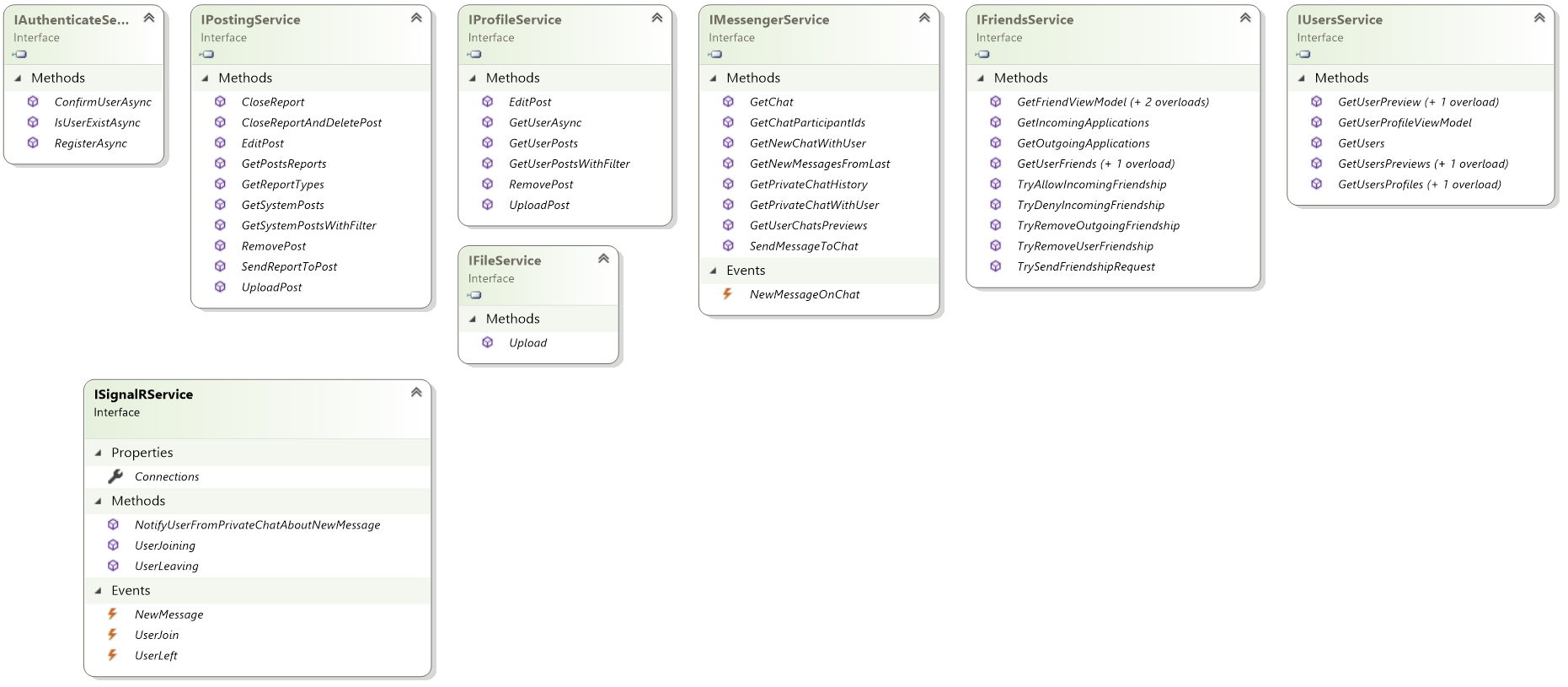


Приложение А1 – Диаграмма вариантов использования

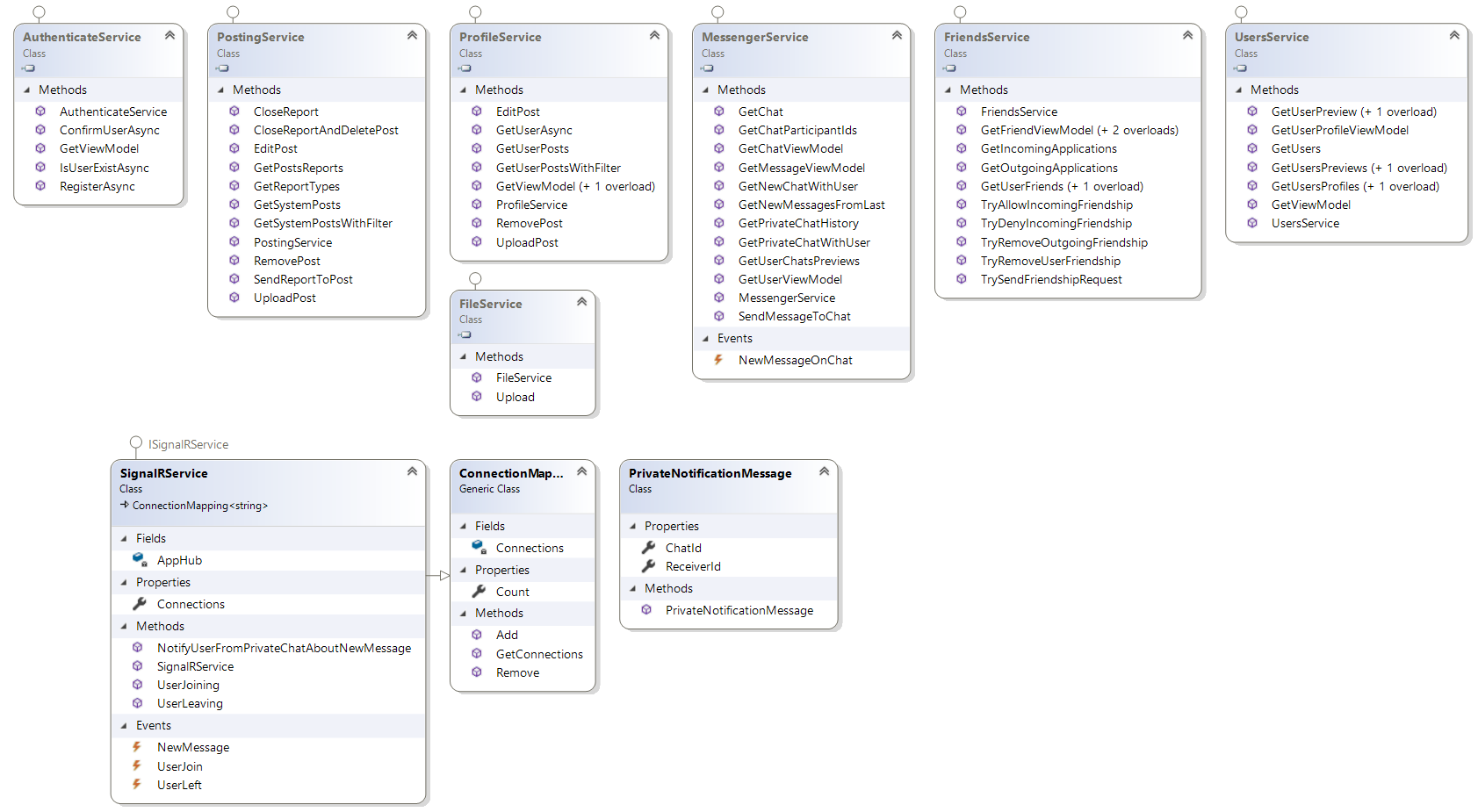
 Приложение А2 – Обобщенная диаграмма классов



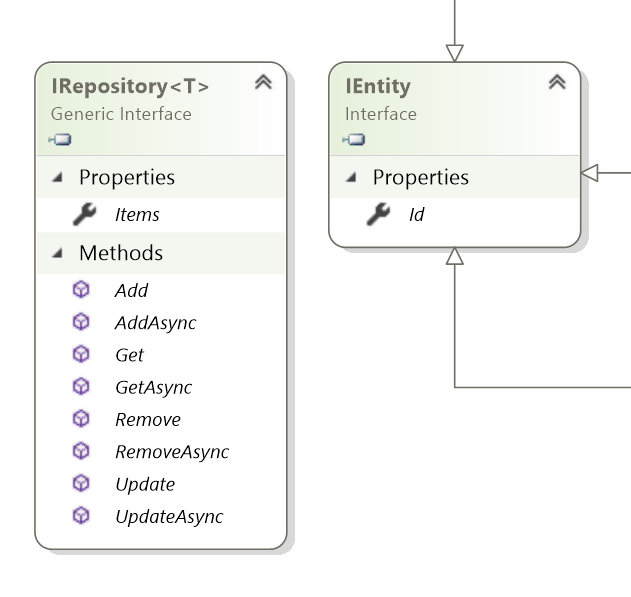
Приложение А3 – Диаграмма классов контроллеров



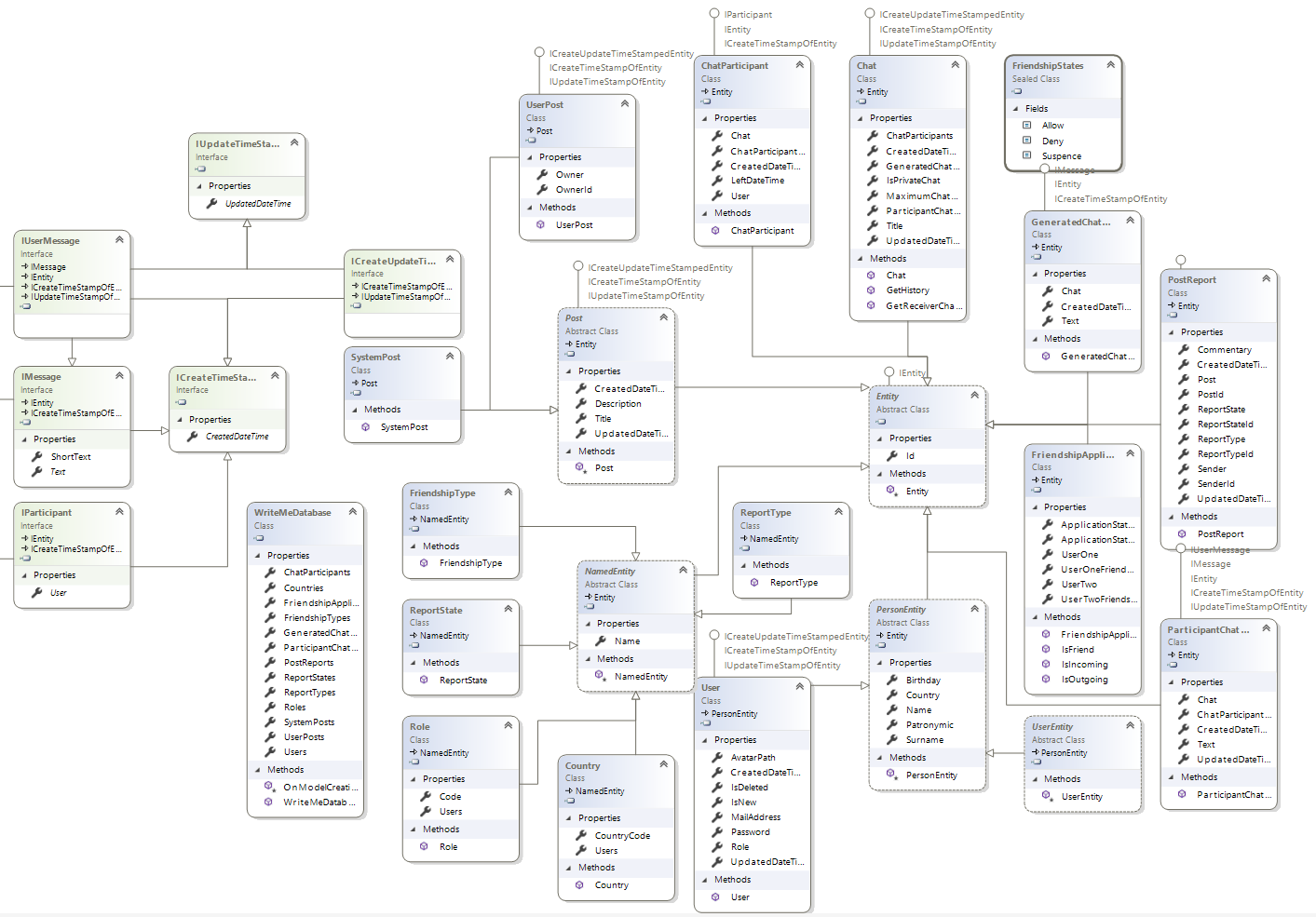
Приложение А4 – Диаграмма классов интерфейсов сервисов



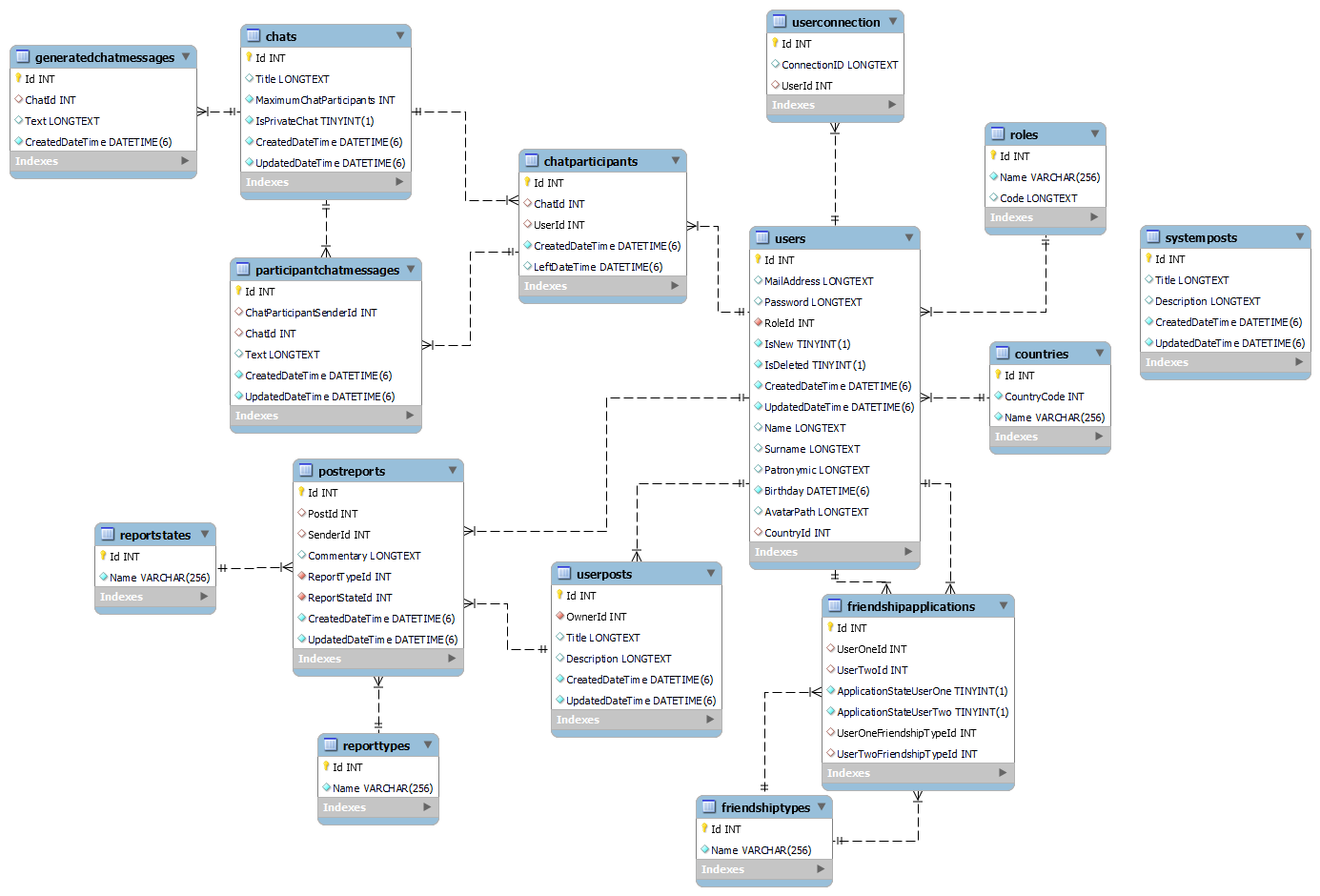
Приложение А5 – Диаграмма классов сервисов



Приложение А6 – Диаграмма классов репозитория



Приложение А7 – Диаграмма классов слоя базы данных



Приложение А8 – EER диаграмма

# СПИСОК СОКРАЩЕНИЙ

MS – Microsoft;

MySQL – MySQL Server;

PK – Первичный ключ;

FK – Внешний ключ;

СУБД – Система управления базами данных;

UCD (Use Case Diagram) – Диаграмма последовательности действий;

EER (Enhanced entity-relationship model) – Расширенная модель «сущность-связь».

1. PK – первичный ключ (primary key)  
   FK – внешний ключ (foreign key) [↑](#footnote-ref-1)