

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**  
**к автоматизированной системе**  
Вариант № 7 «Экологический фонд»

**Составил:**

студент гр. ИСТд-31

*Желепов Алексей Сергеевич*

«19» мая 2014 г.

**Проверил:**

доцент каф. ИВК, к.т.н., доцент

*Родионов Виктор Викторович*

«\_\_\_\_\_» \_\_\_\_\_ 2014 г.

**Ульяновск, 2013**

## Содержание

Список использованных сокращений и обозначений .....	3
Введение.....	4
1 Техническое задание.....	7
1.1 Общие сведения.....	7
2.1 Назначение и цели создания системы .....	7
2.2.1 Назначение системы .....	7
2.2.1 Цели создания системы .....	7
1.3 Характеристика объекта автоматизации.....	8
1.4 Требования к системе .....	8
1.4.1 Требования к системе в целом .....	8
1.4.2 Требования к функциям, выполняемым системой .....	9
1.4.3 Требования к видам обеспечения .....	10
1.5 Состав и содержание работ по созданию системы .....	11
1.6 Порядок контроля и приёмки системы .....	11
1.7 Требования к документированию.....	11
2 Информационное обеспечение системы.....	12
2.1.1 Диаграмма «сущность-связь» .....	12
2.1.2 Сущности и их атрибуты .....	1
2.1.3 Связи между сущностями.....	2
2.1.4 Инструментальное средство моделирования .....	4
2.2.1 Анализ инфологической модели.....	6
2.2.1 База данных системы .....	7
3 Математическое обеспечение системы.....	19
4 Прикладное программное обеспечение системы .....	24
5 Руководство пользователя.....	41
Заключение .....	50
Список использованных источников .....	51
Приложение А. Исходные тексты программных модулей .....	52

## **Список использованных сокращений и обозначений**

1НФ – первая нормальная форма

2НФ – вторая нормальная форма

3НФ – третья нормальная форма

НФБК – нормальная форма Бойса-Кодда

Список должен содержать используемые в тексте пояснительной записки сокращения и обозначения, а также их расшифровку. Общеизвестные обозначения, такие как СУБД, САПР и другие, в список можно не включать и в тексте записки не расшифровывать. Если в списке отсутствуют условные сокращения, то упоминание о них из заголовка должно быть удалено.

## **Введение**

Экологические фонды – это внебюджетные организации, занимающиеся сбором средств на решения экологических проблем.

Источниками поступающих в фонд средств являются средства предприятий в виде платы за нормативные и сверхнормативные выбросы загрязняющих веществ в окружающую среду, также штрафы могут взиматься с браконьеров и прочих лиц, подвергающих опасности дикую живую природу.

Размеры платы за вред окружающей природной среды установлены постановлением Совета Министров - Правительства Российской Федерации от 28.08.92 № 632 «Об утверждении Порядка определения платы и ее предельных размеров за загрязнение окружающей природной среды, размещение отходов, другие виды вредного воздействия», а также в соответствии с другими инструктивно-нормативными документами; сумм, полученных по искам о возмещении вреда, и штрафов за экологические правонарушения.

Штрафы налагаются в пределах компетенции специально уполномоченными на то государственными органами Российской Федерации, органами в области охраны природы, санитарно-эпидемиологического надзора и другими. Конкретный размер налагаемого штрафа определяется органом, налагающим штраф, в зависимости от характера и вида совершенного правонарушения, степени вины правонарушителя и причиненного вреда.

Другой группой средств, поступающих на деятельность фонда, являются пожертвования граждан и организаций-партнеров экологического фонда. В современных реалиях денежные переводы осуществляются при помощи информационных систем-сайтов, позволяющих пользователю переводить деньги со своего электронного кошелька.

Помимо сбора средств на решения экологических проблем, фонд обычно занимается многими другими видами деятельности, среди которых:

1. мониторинг окружающей среды на вид экологических нарушений;

2. тесное сотрудничество с организациями-партнерами и другими фондами;
3. сохранение базы экологических нарушений и их решений;
4. проведение пропаганды среди молодежи о защите окружающей среды и важности ее сохранения;
5. проведение собраний специалистов-экологов для решения возникающих экологических проблем.

Мониторинг окружающей среды осуществляется экологами-специалистами, которые имеют договоренности о сотрудничестве или являются его работниками. Как правило, многие из них расквартированы в различных уголках страны. Это позволяет незамедлительно реагировать на поступающие жалобы об экологических нарушениях, проводить расследование «на месте» и поставлять точные данные организации-фонду.

Благодаря сотрудничеству с другими организациями география действия фонда значительно расширяется, что делает работу экологической организации еще более эффективной.

На основе поступающей информации об экологических нарушениях формируются и назначаются советы экологов-специалистов, на которых выносятся важные решения по поводу хода решения экологических проблем, выставленных на обсуждение на текущем совещании.

Велика роль экологических фондов и в образовательной сфере. Часто специалисты организации назначаются на ведение экологических кружков в школах по защите и сохранении окружающей среды.

Автором курсовой работы также было рассмотрено множество информационных источников, помогших ему спроектировать и продумать механику работы системы. Например, корреспондент газеты «Ведомости»[1] очень подробно изложил информацию об основных источниках поступления средств в экологический фонд. На сайте фонда имени В.И. Вернадского приводится информация о том, что фонд – не только экономическая организация, но и социальная, которая активно взаимодействует с обществом, проводя множество эколо-

гических занятий и семинаров в различных государственных учреждениях (школах, колледжах, университетах).

Помимо источников, касающихся характеристик предметной области, автор курсовой работы было использовано множество статей и книг (в том числе опубликованных на английском языке) о технологии программирования ASP.NET MVC 4 и технологии создания и развертывания баз данных EntityFramework 5. Наиболее полезным источником оказалась книга [3], в которой описаны основные подходы разработки MVC-приложений. Также автор курсовой работы активно использовал сайт MSDN, где было найдено множество ответов на возникающие вопросы во время разработки итогового web-приложения.

[1]

[<http://www.vedomosti.ru/glossary/%D0%AD%D0%BA%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5%20%D1%84%D0%BE%D0%BD%D0%B4%D1%8B>]

[2] <http://www.rsci.ru/grants/fonds/141.php>

[3] Кинжка по MVC

# 1 Техническое задание

## 1.1 Общие сведения

Наименование разрабатываемой системы – Автоматизированная система управления экологическим фондом “Слон” (далее система).

## 2.1 Назначение и цели создания системы

### 2.2.1 Назначение системы

Разрабатываемая система предназначена для автоматизации процессов решения проблем экологии дикой природы. Система оказывает поддержку широкого круга пользователей, заинтересованных в поддержке автоматизированной системы, среди них: пользователи, люди, интересующиеся состоянием окружающей среды; экологи, пополняющие базу данных экологического фонда сведениями о нерешенных проблемах; секретарь, рассматривающий заявки организаций на сотрудничество, определяющий время проведения экологических советов по решению проблем и контролирующий выплаты штрафов со стороны нарушителей экологических прав; организации-партнеры, сообщающие фонду об известных им экологических проблемах; администратор, ответственный за работу системы в целом.

### 2.2.1 Цели создания системы

В результате применения разрабатываемой системы управления будут достигнуты следующие цели:

1. Упрощение сбора информации о проблемах экологии окружающей среды при внедрении системы;
2. Проведение рекламы работы фонда, что должно увеличить количество организаций-партнеров и пользователей при размещении информации о достижениях экологического фонда;
3. Популяризация среди пользователей темы экологического воспитания и возможность электронной регистрации на экологические кружки, преподавателями которых будут специалисты-экологи.
4. Облегчение хранения протоколов и решений с экологических собраний.

### 1.3 Характеристика объекта автоматизации

Экологический фонд является внебюджетным фондом, средства которого направляются на решение экологических проблем окружающей среды. Учредителями экологических фондов являются краевые, областные и республиканские комитеты по экологии и природоведению. Организации, как правило, являются самостоятельными юридическими лицами, имеют собственный баланс. Основными задачами работы экологических фондов являются проведение мероприятий и разработка программ по следующим направлениям:

1. создание информационной системы сбора, хранения, систематизации и обработки экологической информации;
2. проведение мероприятий с целью решения экологических проблем;
3. ведение базы сбора информации об экологических нарушениях. Нарушение может быть как со стороны организации (сброс химически опасных веществ заводом в реку), так и со стороны физического лица (браконьерство);
4. организация экологического образования и воспитания, пропаганда экологических знаний.

Экологический фонд является неотъемлемой частью механизма регулирования природопользования и образуется за счет поступлений средств от предприятий и физических лиц нарушителей. Таким образом, в основном денежный баланс фонда формируется из следующих платежей:

1. штрафы за загрязнение окружающей среды;
2. сверхнормативное использование природных ресурсов;
3. штрафы за нарушение природоохранного законодательства.

### 1.4 Требования к системе

#### 1.4.1 Требования к системе в целом

##### 1.4.1.1 Требования к структуре и функционированию системы

Определяется общей постановкой задачи задания на курсовую работу



#### 1.4.1.2 Требования к защите информации от несанкционированного доступа

Защита информации от несанкционированного доступа является важной особенностью при разработке подобной системы, так как в этой системе имеются операции с банковскими счетами, переводами денежных средств и личными данными зарегистрированных пользователей. Утечка таких данных может повлечь за собой ухудшение взаимоотношений между фондом и его партнерами, а также тривиальную кражу денежных средств.

Основными мерами по сохранению данных от несанкционированного доступа к данным являются стандартные уровни защиты SQL Server, среди которых выделяются:

1. хранение строки подключения к базе данных в защищенном конфигурационном файле;
2. использование аутентификации Windows при работе с SQL-сервером базы данных.
3. использование Code First подхода технологии Entity Framework, позволяющего скрыть данные методом инкапсуляции объектно-ориентированного программирования.

#### 1.4.2 Требования к функциям, выполняемым системой

Неавторизованный пользователь системы имеет следующие возможности:

1. просмотра общей информации об экологическом фонде, а также сведений об организациях-партнерах;
2. просмотра списка текущих экологических проблем, взятых в разработку фондом, и информация о них;
3. просмотра достижений экологического фонда;
4. просмотра информации об экологических кружках.

Представитель организации-партнера имеет следующую возможность:

1. добавления записи-жалобы на экологическое нарушение.

Рядовой пользователь системы имеет следующие возможности:

1. добавления записи-жалобы на экологическое нарушение;
2. электронной регистрации на посещение экологических кружков.

Эколог имеет следующие возможности при пользовании системой:

1. добавления записи о новых экологических проблемах на основе проводимого экологом мониторинга;
2. регистрации своего участия в организованных фондом экологических собраниях по отслеживанию хода решения определенной экологической проблемы;
3. рассмотрения жалоб пользователей системы на экологические проблемы и принятия решения по включению их в соответствующий список или их отклонение.

Секретарь имеет возможность доступа к следующим операциям системы:

1. рассмотрение заявок организаций на сотрудничество с экологическим фондом;
2. объявления информации по проведению экологических советов;
3. контроль выплаты штрафов со стороны организаций/лиц-штрафников.

Администратор системы имеет следующие возможности:

1. добавление/удаление учетных записей пользователей системы;
2. редактирование и добавление информации о достижениях фонда по решению экологических проблем;
3. создание записей о новых экологических кружках.

Следует отметить, что все авторизованные пользователи также могут просматривать информацию, как и неавторизованные.

### 1.4.3 Требования к видам обеспечения

#### 1.4.3.1 Требования к техническому обеспечению

Материнская плата – Intel HM77 Express

Процессор – Intel Core i5-3120M, 2500 MHz

Оперативная память – DDR3-1333 Memory, 6 Гб

Жесткий диск – HDD Sata, 500 Гб.

Видеокарта – NVIDIA GeForce 710M, 2Гб

Звуковая карта – Intel High Definition Audio

Монитор – Acer Aspire V3-771G-53216G75Maii Display 17.3”.

Оптический привод – Acer Aspire V3-771G-53216G75Maii DVD-RW

#### 1.4.3.2 Требования к программному обеспечению

При разработке системы будет использоваться операционная система семейства Windows версии 8.1.

При создании диаграммы “сущность-связь” будет использоваться программа Erwin Data Modeler.

#### 1.5 Состав и содержание работ по созданию системы

Определяется этапами выполнения работы задания на курсовую работу.

#### 1.6 Порядок контроля и приёмки системы

Определяется порядком защиты и критериями оценки работы задания на курсовую работу.

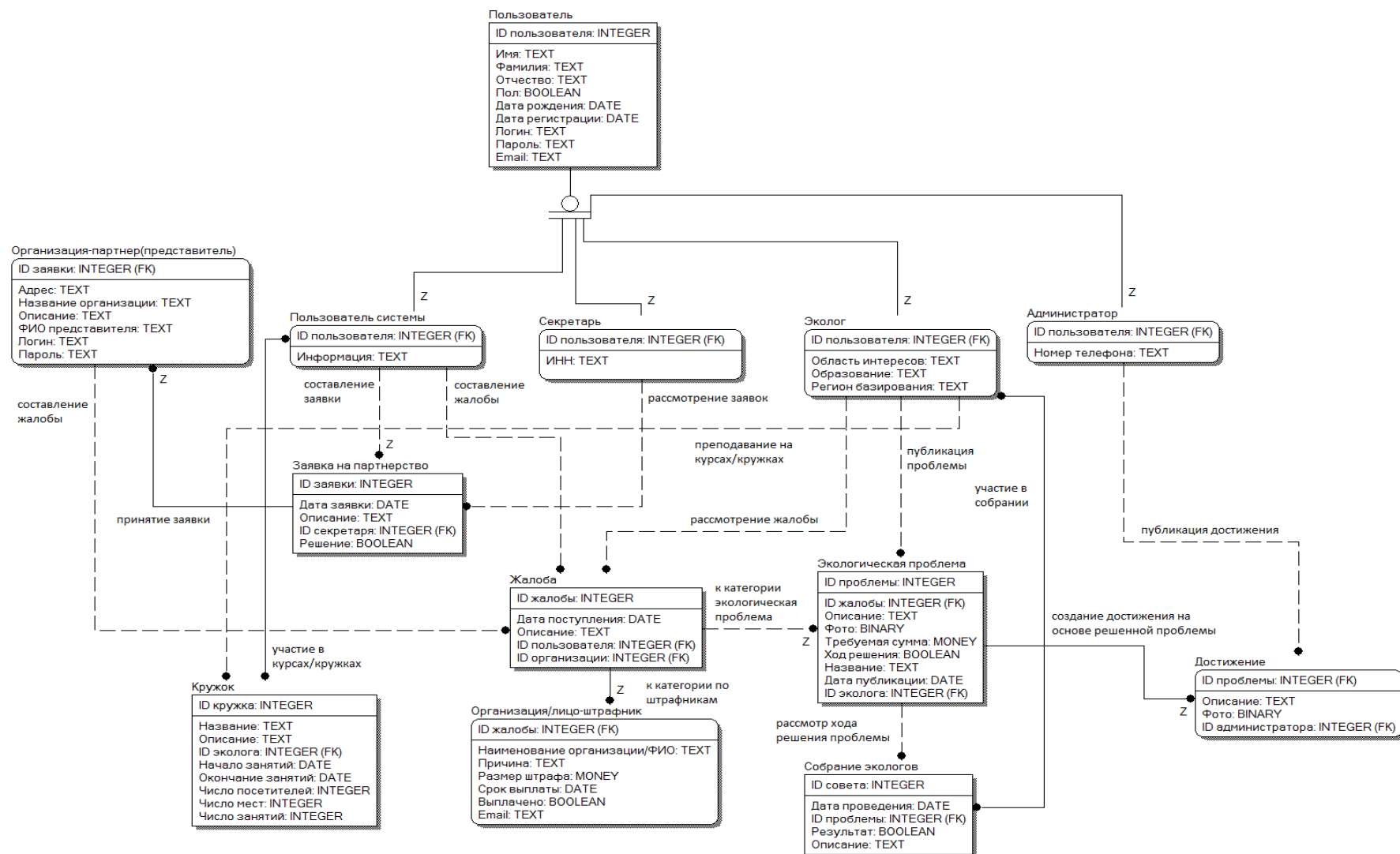
#### 1.7 Требования к документированию

Ни один компонент из пояснительной записки удален не будет.

## 2 Информационное обеспечение системы

### 2.1 Инфологическая модель предметной области

#### 2.1.1 Диаграмма «сущность-связь»



## 2.1.2 Сущности и их атрибуты

Сущность «**Пользователь**» – содержит в себе общую информацию обо всех пользователях системы.

Сущность «**Пользователь системы**» – описывает рядового пользователя системы.

Сущность «**Секретарь**» – описывает работника фонда, занимающегося рассмотрением заявок на сотрудничество и объявлением информации по проведению экологических советов.

Сущность «**Организация-партнер (представитель)**» – приводится описание организации, которая сотрудничает с экологическим фондом «Слон».

Сущность «**Эколог**» – описывает специалистов, занятых поиском и фиксированием в системе экологических проблем на территории.

Сущность «**Администратор**» – описывает сотрудников, занимающихся сопровождением и технической поддержкой системы.

Сущность «**Заявка на партнерство**» – описывает заявки, принятые от пользователей (представителей организаций), желающих заключить партнерские отношения с фондом.

Сущность «**Жалоба**» – описывает жалобы по нарушению экологических прав и законов, принятые от рядовых пользователей системы и представителей организаций-партнеров. Атрибут *ID пользователя* заполняется в случае указании причины пользователем системы, при этом атрибут *ID организации* не заполняется. Возможна и обратная ситуация.

Сущность «**Организация/лицо-штрафник**» – включает в себя записи о тех организациях и людях, которые нарушили экологические права и законы. Атрибут *Ход решения* показывает, была ли произведена выплата штрафа. Атрибут *Причина* содержит в себе пояснение, написанное экологом, о причинах введения штрафных санкций по отношению к организации/лицу-штрафнику.

Сущность «**Экологическая проблема**» – описывает экологическую проблему, обнаруженную экологом в результате мониторинга, или обозначенную

на основе жалобы, выраженной пользователем системы. Атрибут *Ход решения* показывает, была ли решена проблема. Если ответ на этот вопрос положительный – это сигнал администратору системы опубликовать отзыв о ней и добавить его в **Достижения** фонда.

Сущность «**Собрание экологов**» – описывает встречу экологов по решению определенной экологической проблемы. Атрибут *Результат* показывает, какое было принято решение по результатам собрания и были ли проведены соответствующие меры со стороны фонда по решению проблемы.

Сущность «**Достижения**» – включает в себя записи о тех экологических проблемах, которые фонду удалось решить.

Сущность «**Кружок**» – описывает всю необходимую информацию о кружках и секциях, посвященных экологическому воспитанию. Атрибут *ID эколога* определяет эколога-преподавателя, который будет вести занятия. Атрибут *Число мест* показывает максимальное количество людей, которое может быть включено в группу на данный курс.

### 2.1.3 Связи между сущностями

Сущности «**Пользователь**» и «**Пользователь системы**» соединены связью типа «Есть».

Сущности «**Пользователь**» и «**Секретарь**» соединены связью типа «Есть».

Сущности «**Пользователь**» и «**Эколог**» соединены связью типа «Есть».

Сущности «**Пользователь**» и «**Администратор**» соединены связью типа «Есть».

Сущности «**Заявка на партнерство**» и «**Организация-партнер (представитель)**» соединены связью, которая идентифицирует отношение, тип связи – «1,1:0,1». Каждая новая организация-партнер должна была оставить заявку на сотрудничество, каждой заявке соответствует одна организация-партнер или не соответствует в случае отклонения заявки секретарем.

Сущности «**Организация-партнер**» и «**Жалоба**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0,N». У жалобы может быть только один инициатор (в данном случае представитель организации-партнера), в то время как представитель компании может давать неограниченное число жалоб.

Сущности «**Секретарь**» и «**Заявка**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0,N». Конкретную заявку может рассматривать только один секретарь, в то время как секретарь может рассматривать множество заявок.

Сущности «**Пользователь системы**» и «**Заявка на партнерство**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0,1». Заявка составляется одним пользователем системы, в то время как пользователь может составить либо одну заявку для секретаря, либо ни одной.

Сущности «**Пользователь системы**» и «**Кружок**» соединены связью, которая не идентифицирует отношение, тип связи – «0,N:0,M». Это означает, что пользователь может регистрироваться на множество кружков, а на кружок может быть зарегистрировано множество пользователей.

Сущности «**Пользователь системы**» и «**Жалоба**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0:N». Жалоба может быть написана одним пользователем, пользователь может писать множество жалоб.

Сущности «**Эколог**» и «**Жалоба**» соединены связью, не идентифицирующей отношение, тип связи – «1,1:0:N». Конкретная жалоба может рассматриваться только одним экологом, в то время как специалист может рассматривать множество жалоб.

Сущности «**Эколог**» и «**Экологическая проблема**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0,N». Это означает, что проблема может быть опубликована одним экологом. В свою очередь, эколог может записывать информацию о неограниченном количестве экологических проблем.

Сущности «**Эколог**» и «**Собрание экологов**» соединены связью, которая не идентифицирует отношение, тип связи – «0,N:0,M». Эколог может входить во множество рабочих групп по разрешению конкретных проблем, так и собрание экологов может состоять из нескольких специалистов.

Сущности «**Эколог**» и «**Кружок**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0,N». Кружок ведется одним преподавателем, в то время как эколог может преподавать на нескольких кружках одновременно.

Сущности «**Жалоба**» и «**Организация/лицо-штрафник**» соединены связью, которая идентифицирует отношение, тип связи – «1,1:0:1». Организация/лицо появляются в списке только, если на нее/него была подана жалоба. При этом жалоба может быть написана на организацию/лицо или же такой жалобы может и не существовать.

Сущности «**Экологическая проблема**» и «**Собрание экологов**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0,N». Группа экологов в течение заседания пытается разрешить взятую конкретную проблему. Проблема может рассматриваться на нескольких собраниях.

Сущности «**Экологическая проблема**» и «**Достижение**» соединены связью, которая идентифицирует отношение, тип связи – «1,1:0,1». Решенной проблеме соответствует либо одна публикация о достижении фонда или ни одной публикации. Одной публикации соответствует только одна решенная проблема.

Сущности «**Администратор**» и «**Достижение**» соединены связью, которая не идентифицирует отношение, тип связи – «1,1:0,N». Статья о решенной проблеме может быть опубликована только одним администратором, а каждый администратор может опубликовать множество статей.

#### 2.1.4 Инструментальное средство моделирования

При разработке инфологической модели базы данных мною были рассмотрены следующие программные продукты: Erwin Data Modeler, ERConstructor и



Microsoft Office Visio. На основании проведенного анализа ( табл. 1) программного обеспечения была выбрана программа Erwin Data Modeler. Ключевыми критериями выбора программного пакета Erwin Data Modeler стали первый и четвертый критерии оценки.

Таблица 2.1.4.1.

«Сравнение программного обеспечения для построения инфологических моделей»

<b>Критерий сравнения</b>	<b>ERConstructor</b>	<b>Erwin Data Modeler</b>	<b>Visio</b>
Простота использования	Достаточно просто использовать	Достаточно просто использовать	Достаточно просто использовать
Быстрота создания модели	Быстро	Быстро	Достаточно долго, т.к. приходится рисовать модели сущностей самостоятельно
Стоимость ПО	Бесплатное ПО	Платное ПО	Необходимо приобретать лицензию на использование
Опыт использования ПО на момент начала курсового проектирования	-	+	+

## 2.2 Даталогическая модель предметной области

### 2.2.1 Анализ инфологической модели

В представленной инфологической модели существуют следующие типы связи:

1. Связь типа «Есть». Например, между сущностями «Пользователь» и «Секретарь».
2. Связь типа 0,N:0,M (Многие-ко-многим). Например, данный тип используется между сущностями «Пользователь системы» и «Экологическая секция».
3. Связь типа 1,1:0,N. Например, сущности «Эколог» и «Экологическая секция» соединены таким типом связи.
4. Связь типа 1,1:0,1 (Один-к-одному). Например, тип используется между сущностями «Жалоба» и «Организация/лицо-штрафник».

Наиболее сложной для реализации оказалась связь типа Многие-ко-многим. Для ее объявления необходимо создание третьей таблицы, содержащей пары значений соответствующих идентификаторов. На примере связи между сущностями «Пользователь системы» и «Экологическая секция» это будет выглядеть следующим образом: «Идентификатор пользователя, Идентификатор секции».

Еще одна сложность создания связи Многие-ко-многим состояла в том, что при удалении одной из записей, необходимо было определять: применять каскадное удаление записей или разрушать только связь с удалением одной из записей. При реализации был выбран второй вариант. Этот выбор объясняется требованиями к системе.

Связи типа 1,1:0,N или 1,1:0,1 между сущностями не вызвали затруднений, так как создаются две отдельные таблицы, связанные между собой по идентифицирующему столбцу.

Связь типа «Есть» была реализована при помощи связи типа 1,1:0,1, что является логичным, так как запись из первой таблицы соответствует только одной записи из второй.

## 2.2.1 База данных системы

### 2.2.1.1 Таблица [dbo].[Users]

```
CREATE TABLE [dbo].[Users](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Name] [nvarchar](max) NOT NULL,
    [Surname] [nvarchar](max) NOT NULL,
    [FatherName] [nvarchar](max) NOT NULL,
    [Sex] [bit] NOT NULL,
    [BirthDate] [datetime] NOT NULL,
    [Login] [nvarchar](max) NOT NULL,
    [Password] [nvarchar](max) NOT NULL,
    [Email] [nvarchar](max) NOT NULL,
    [RegistrationDate] [datetime] NOT NULL,
    [Address] [nvarchar](max) NULL,
    [CompanyName] [nvarchar](max) NULL,
    [Description] [nvarchar](max) NULL,
    [Reason] [nvarchar](max) NULL,
    [IsSolved] [bit] NULL,
    [Discriminator] [nvarchar](128) NULL,
    [Secretary_ID] [int] NULL,
    CONSTRAINT [PK_dbo.Users] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет первичный ключ ID, который является единственным детерминантом.

### 2.2.1.2 Таблица [dbo].[Secretaries]

```
CREATE TABLE [dbo].[Secretaries](
    [ID] [int] NOT NULL,
    [IndividualTaxNumber] [nvarchar](max) NULL,
    CONSTRAINT [PK_dbo.Secretaries] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет первичный ключ ID, который является единственным детерминантом.

### 2.2.1.3 Таблица [dbo].[RankUsers]

```
CREATE TABLE [dbo].[RankUsers](
    [ID] [int] NOT NULL,
    [Information] [nvarchar](max) NULL,
    CONSTRAINT [PK_dbo.RankUsers] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет первичный ключ ID, который является единственным детерминантом.

#### 2.2.1.4 Таблица [dbo].[Administrators]

```
CREATE TABLE [dbo].[Administrators](
    [ID] [int] NOT NULL,
    [PhoneNumber] [nvarchar](max) NULL,
    CONSTRAINT [PK_dbo.Administrators] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет первичный ключ ID, который является единственным детерминантом.

#### 2.2.1.5 Таблица [dbo].[Ecologists]

```
CREATE TABLE [dbo].[Ecologists](
    [ID] [int] NOT NULL,
    [Education] [nvarchar](max) NOT NULL,
    [InterestsSphere] [nvarchar](max) NULL,
    [DistrictLocation] [nvarchar](max) NULL,
    CONSTRAINT [PK_dbo.Ecologists] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет первичный ключ ID, который является единственным детерминантом.

#### 2.2.1.6 Таблица [dbo].[Partners]

```
CREATE TABLE [dbo].[Partners](
    [ID] [int] NOT NULL,
    [Secretary_ID] [int] NOT NULL,
    [Address] [nvarchar](max) NOT NULL,
    [CompanyName] [nvarchar](max) NOT NULL,
    [Description] [nvarchar](max) NOT NULL,
    [Reason] [nvarchar](max) NOT NULL,
    [IsSolved] [bit] NOT NULL,
    CONSTRAINT [PK_dbo.Partners] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет первичный ключ ID, который является единственным детерминантом.

#### 2.2.1.7 Таблица [dbo].[EcologicalProblems]

```
CREATE TABLE [dbo].[EcologicalProblems](
    [ProblemID] [int] NOT NULL,
    [Description] [nvarchar](max) NOT NULL,
    [PhotoType] [nvarchar](max) NULL,
    [PhotoFile] [varbinary](max) NULL,
    [RequiredSum] [decimal](18, 2) NOT NULL,
```

```

[IsSolved] [bit] NOT NULL,
[Title] [nvarchar](max) NOT NULL,
[PublicationDate] [datetime] NOT NULL,
[Creator_ID] [int] NOT NULL,
CONSTRAINT [PK_dbo.EcologicalProblems] PRIMARY KEY CLUSTERED
(
    [ProblemID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ProblemID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант (Title, Creator\_ID) являющийся ключом-кандидатом.

#### 2.2.1.8 Таблица [dbo].[Councils]

```

CREATE TABLE [dbo].[Councils](
    [CouncilID] [int] IDENTITY(1,1) NOT NULL,
    [Title] [nvarchar](max) NOT NULL,
    [AssignmentDate] [datetime] NOT NULL,
    [CouncilResult] [bit] NOT NULL,
    [Description] [nvarchar](max) NOT NULL,
    [Problem_ProblemID] [int] NOT NULL,
    CONSTRAINT [PK_dbo.Councils] PRIMARY KEY CLUSTERED
(
    [CouncilID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ детерминант (Title, Problem\_ProblemID) является ключом-кандидатом.

#### 2.2.1.9 Таблица [dbo].[Complaints]

```
CREATE TABLE [dbo].[Complaints](
    [ComplaintID] [int] IDENTITY(1,1) NOT NULL,
    [AppearingDate] [datetime] NOT NULL,
    [Description] [nvarchar](max) NOT NULL,
    [Title] [nvarchar](max) NOT NULL,
    [IsHidden] [bit] NOT NULL,
    [Creator_ID] [int] NOT NULL,
    [Partner_ID] [int] NULL,
    [Ecologist_ID] [int] NULL,
    CONSTRAINT [PK_dbo.Complaints] PRIMARY KEY CLUSTERED
(
    [ComplaintID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута ComplaintID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант (Title, Creator\_ID), который является ключом-кандидатом.

#### 2.2.1.10 Таблица [dbo].[Achievements]

```
CREATE TABLE [dbo].[Achievements](
    [AchievementID] [int] NOT NULL,
    [Description] [nvarchar](max) NULL,
    [PhotoType] [nvarchar](max) NULL,
    [PhotoFile] [varbinary](max) NULL,
    [Title] [nvarchar](max) NOT NULL,
    [Administrator_ID] [int] NOT NULL,
```



```

CONSTRAINT [PK_dbo.Achievements] PRIMARY KEY CLUSTERED
(
    [AchievementID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута AchievementID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант (Title, Administrator\_ID), который является ключом-кандидатом.

#### 2.2.1.11 Таблица [dbo].[OrganisationDebtors]

```

CREATE TABLE [dbo].[OrganisationDebtors](
    [OrganisationDebtorID] [int] NOT NULL,
    [Name] [nvarchar](max) NOT NULL,
    [Reason] [nvarchar](max) NOT NULL,
    [FineAmount] [decimal](18, 2) NOT NULL,
    [PayTime] [datetime] NOT NULL,
    [IsPaid] [bit] NOT NULL,
    [Email] [nvarchar](max) NULL,
    [ResponsiblePerson_ID] [int] NULL,
    CONSTRAINT [PK_dbo.OrganisationDebtors] PRIMARY KEY CLUSTERED
(
    [OrganisationDebtorID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута OrganisationDebtorID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант OrganisationDebtorID, который является первичным ключом таблицы.

#### 2.2.1.12 Таблица [dbo].[Sections]

```
CREATE TABLE [dbo].[Sections](
    [SectionID] [int] IDENTITY(1,1) NOT NULL,
    [Title] [nvarchar](max) NOT NULL,
    [Description] [nvarchar](max) NOT NULL,
    [StartLessonsTime] [datetime] NOT NULL,
    [ParticipantsCount] [int] NOT NULL,
    [SpotsCount] [int] NOT NULL,
    [FreeSpotsCount] [int] NOT NULL,
    [LessonsCount] [int] NOT NULL,
    [Ecologist_ID] [int] NOT NULL,
    CONSTRAINT [PK_dbo.Sections] PRIMARY KEY CLUSTERED
(
    [SectionID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута Section\_ID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа Section\_ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант (Title, Ecologist\_ID), который является ключом-кандидатом.

#### 2.2.1.13 Таблица [dbo].[PartnershipRequests]

```
CREATE TABLE [dbo].[PartnershipRequests](
    [RequestID] [int] IDENTITY(1,1) NOT NULL,
    [Reason] [nvarchar](max) NULL,
    [IsAccepted] [bit] NOT NULL,
    CONSTRAINT [PK_dbo.PartnershipRequests] PRIMARY KEY CLUSTERED
```

```
(
    [RequestID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута RequestID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей, так как все атрибуты полностью и функционально зависят от ключа Request\_ID.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант RequestID, который является первичным ключом таблицы.

#### 2.2.1.14 Таблица [dbo].[SectionRankUsers]

```
CREATE TABLE [dbo].[SectionRankUsers](
    [Section_SectionID] [int] NOT NULL,
    [RankUser_ID] [int] NOT NULL,
    CONSTRAINT [PK_dbo.SectionRankUsers] PRIMARY KEY CLUSTERED
(
    [Section_SectionID] ASC,
    [RankUser_ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута Section\_SectionID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант Section\_SectionID, который является первичным ключом таблицы.

### 2.2.1.15 Таблица [dbo].[CouncilEcologists]

```
CREATE TABLE [dbo].[CouncilEcologists](
    [Council_CouncilID] [int] NOT NULL,
    [Ecologist_ID] [int] NOT NULL,
    CONSTRAINT [PK_dbo.CouncilEcologists] PRIMARY KEY CLUSTERED
(
    [Council_CouncilID] ASC,
    [Ecologist_ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
```

Таблица находится в 1НФ, так как содержит только простые и однозначные атрибуты.

Таблица находится во 2НФ, так как она находится в 1НФ и каждый неключевой атрибут функционально зависит только от атрибута Council\_CouncilID, который является ключом таблицы.

Таблица находится в 3НФ, так как находится во 2НФ и не содержит транзитивных зависимостей.

Таблица находится в НФБК, так как находится в 3НФ и имеет детерминант Council\_CouncilID, который является первичным ключом таблицы.

### 2.2.1.16 Хранимая процедура [dbo].[GetCrucialDebtor] с использованием транзакции

```
CREATE PROCEDURE [dbo].[GetCrucialDebtor]
    @day_count int
AS
BEGIN
    BEGIN TRANSACTION;
    SET NOCOUNT ON;
    SELECT * FROM [dbo].[OrganisationDebtors] WHERE (PayTime
<= DATEADD(DAY, @day_count, GETDATE())) AND IsPaid = 0)
    COMMIT TRANSACTION;
END
```

Примечание: Данная процедура используется секретарем системы для получения списка организаций/лиц-должников, срок выплаты штрафных санкций которых, наступает в течение @day\_count дней.

### 2.2.1.17 Хранимая процедура [dbo].[GetEcologicalProblems] с использованием транзакции

```
CREATE PROCEDURE [dbo].[GetEcologicalProblems]
    @days_range int
AS
BEGIN
    BEGIN TRANSACTION;
    SET NOCOUNT ON;
    SELECT * FROM [dbo].[EcologicalProblems] WHERE
(PublicationDate >= DATEADD(DAY, -@days_range, GETDATE()))
    COMMIT TRANSACTION;
END
```

Примечание: Данная процедура используется секретарем системы для получения списка экологических проблем, которые были опубликованы в интервале @days\_range числа дней.

### 2.2.1.18 Функция [dbo].[GetAdultsQuantity]

```
CREATE FUNCTION [dbo].[GetAdultsQuantity]
(
)
RETURNS int
AS
BEGIN
    DECLARE @adults int;
    SELECT @adults = COUNT(*)
    FROM [dbo].[Users] WHERE BirthDate <= DATEADD(YEAR, -
18, GETDATE())
    RETURN @adults
END
```

Примечание: Данная функция возвращает скалярное значение числа пользователей, которые являются совершеннолетними.

### 2.2.1.19 Функция [dbo].[GetChildrenQuantity]

```
CREATE FUNCTION [dbo].[GetChildrenQuantity]
(
)
RETURNS int
AS
BEGIN
    DECLARE @children int;
    SELECT @children = COUNT(*)
    FROM [dbo].[Users] WHERE BirthDate >= DATEADD(YEAR, -
18, GETDATE())
    RETURN @children
END
```

Примечание: Данная функция возвращает скалярное значение числа пользователей, которые являются несовершеннолетними.

#### 2.2.1.20 Триггер [dbo].[CalculateSpotsForInsert]

```
CREATE TRIGGER [dbo].[CalculateSpotsForInsert]
    ON [dbo].[SectionRankUsers]
    AFTER INSERT
    AS
    BEGIN
        SET NOCOUNT ON;
        UPDATE dbo.Sections SET ParticipantsCount = (SELECT
COUNT(*) FROM [dbo].[SectionRankUsers] WHERE [dbo].[SectionRankUsers].Section_SectionID =
SectionId) WHERE SectionId IN (SELECT i.Section_SectionId FROM inserted AS i)
        UPDATE dbo.Sections SET FreeSpotsCount = SpotsCount -
(SELECT COUNT(*) FROM [dbo].[SectionRankUsers] WHERE
[dbo].[SectionRankUsers].Section_SectionID = SectionId) WHERE SectionId IN (SELECT
i.Section_SectionId FROM inserted AS i)
    END
```

Примечание: Данный триггер предназначен для пересчета числа свободных мест и числа посетителей экологической секции после регистрации нового участника.

#### 2.2.1.21 Триггер [dbo].[CalculateSpotsForDelete]

```
CREATE TRIGGER [dbo].[CalculateSpotsForDelete]
    ON [dbo].[SectionRankUsers]
    AFTER DELETE
    AS
    BEGIN
        SET NOCOUNT ON;
        UPDATE dbo.Sections SET ParticipantsCount = (SELECT
COUNT(*) FROM [dbo].[SectionRankUsers] WHERE [dbo].[SectionRankUsers].Section_SectionID =
SectionId) WHERE SectionId IN (SELECT i.Section_SectionId FROM deleted AS i)
        UPDATE dbo.Sections SET FreeSpotsCount = SpotsCount -
(SELECT COUNT(*) FROM [dbo].[SectionRankUsers] WHERE
[dbo].[SectionRankUsers].Section_SectionID = SectionId) WHERE SectionId IN (SELECT
i.Section_SectionId FROM deleted AS i)
    END
```

Примечание: Данный триггер предназначен для пересчета числа свободных мест и числа посетителей экологической секции после deregистрации одного из участников.

### 3 Математическое обеспечение системы

Приводится полное описание основных алгоритмов.

#### 3.1 Регистрация пользователя системы

**1. Общая характеристика:** Алгоритм регистрации и добавления сведений о новом зарегистрированном пользователе в базу данных.

**2. Используемые данные:** таблицы [dbo].[RankUsers], [dbo].[Partners] и [dbo].[Ecologists].

**3. Результаты выполнения:** добавление записи о новом пользователе в одну из таблиц (в зависимости, от того какой тип регистрации выбрал пользователь): [dbo].[RankUsers], [dbo].[Partners] или [dbo].[Ecologists]. В случае успешного прохождения регистрации осуществляется переход на страницу с соответствующим сообщением, иначе указание регистрирующемуся всех совершенных им ошибок в ходе регистрации.

**4. Математическое описание:** -.

**5. Логическое описание:**

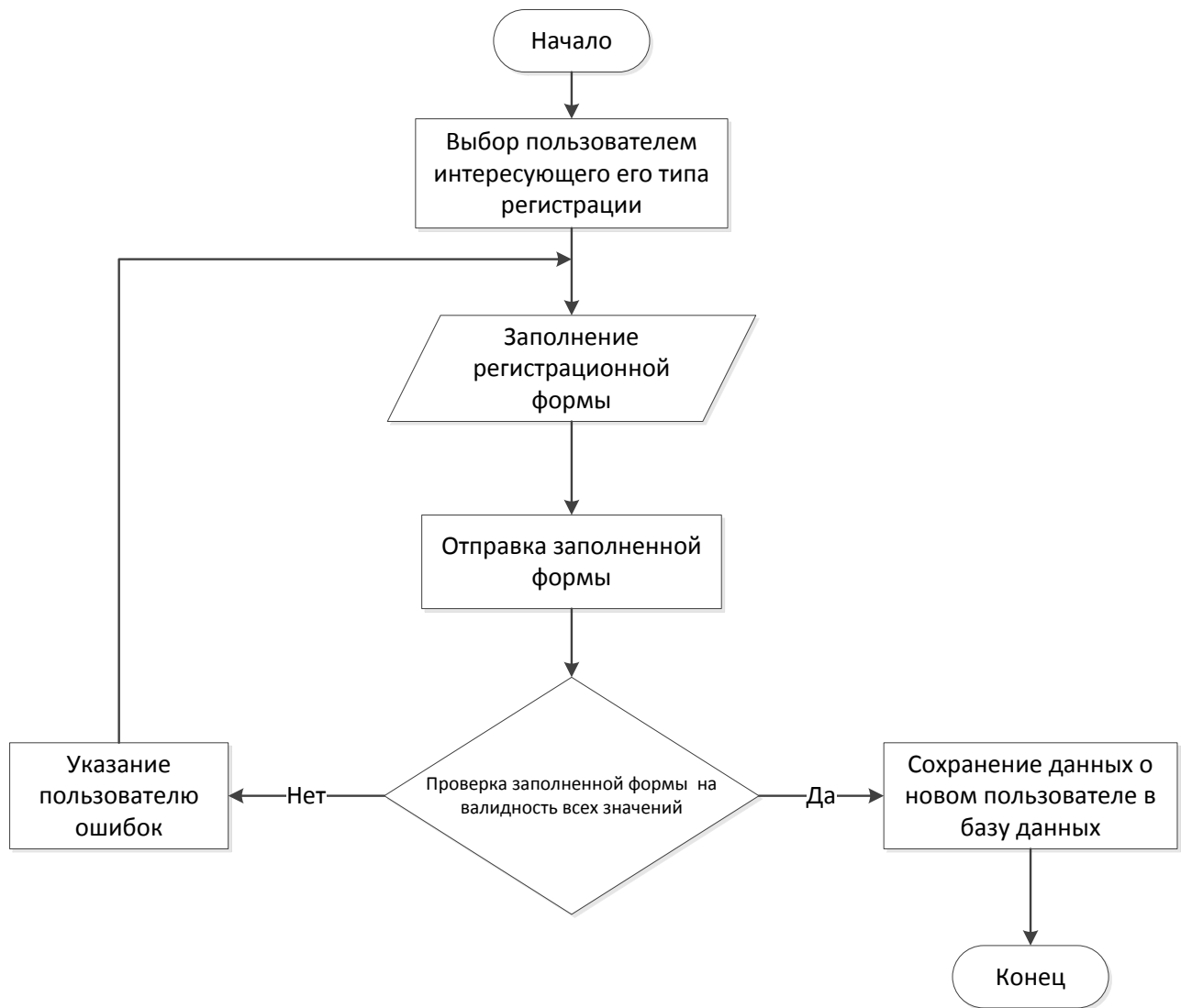


Рис. 3.1.1. Блок-схема алгоритма регистрации пользователя

### 3.2 Авторизация пользователя системы

**1. Общая характеристика:** Алгоритм авторизации пользователя системы предназначен для определения его роли в системе с целью открытия определенного дополнительного функционала.

**2. Используемые данные:** таблицы [dbo].[RankUsers], [dbo].[Partners] и [dbo].[Ecologists], [dbo].[Administrators], [dbo].[Secretaries].

**3. Результаты выполнения:** Открытие дополнительных возможностей при пользовании системой.

**4. Математическое описание:** -.

**5. Логическое описание:**



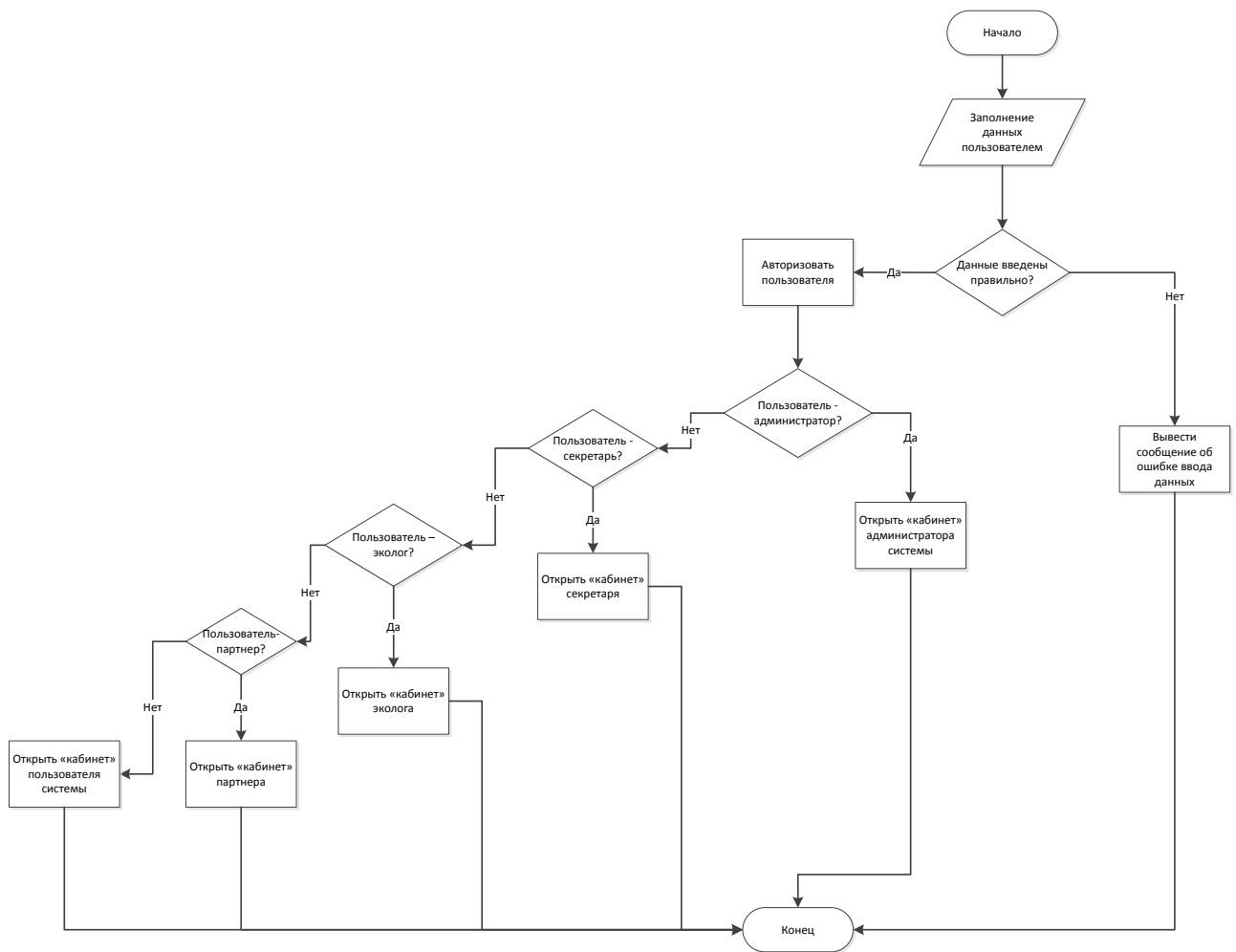


Рис. 3.2.1. Блок-схема алгоритма авторизации пользователя

### 3.3 Поиск информации об экологе

1. **Общая характеристика:** Алгоритм поиска осуществляет поиск информации о конкретном экологе или группе специалистов.

2. **Используемые данные:** таблицы [dbo].[Ecologists].

3. **Результаты выполнения:** Отображение информации о конкретном экологе или группе специалистов в виде таблицы.

4. **Математическое описание:** -.

5. **Логическое описание:**

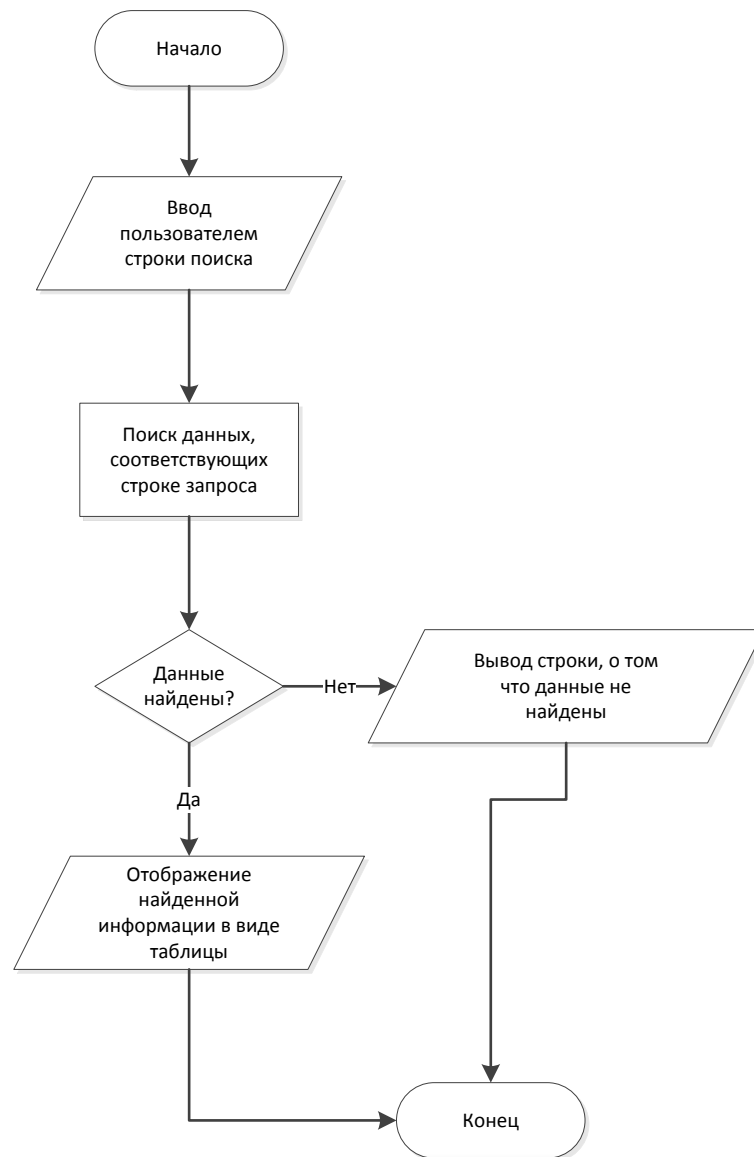


Рис. 3.3.1. Блок-схема алгоритма поиска информации об экологах

### 3.4 Фильтрация информационных данных об организациях/лицах-штрафниках

**1. Общая характеристика:** Алгоритм фильтрации данных по атрибуту «IsPayed».

**2. Используемые данные:** таблицы [dbo].[OrganisationDebtors].

**3. Результаты выполнения:** Отображение информации о выплативших/не выплативших штрафы организаций.

**4. Математическое описание:** -.

**5. Логическое описание:**

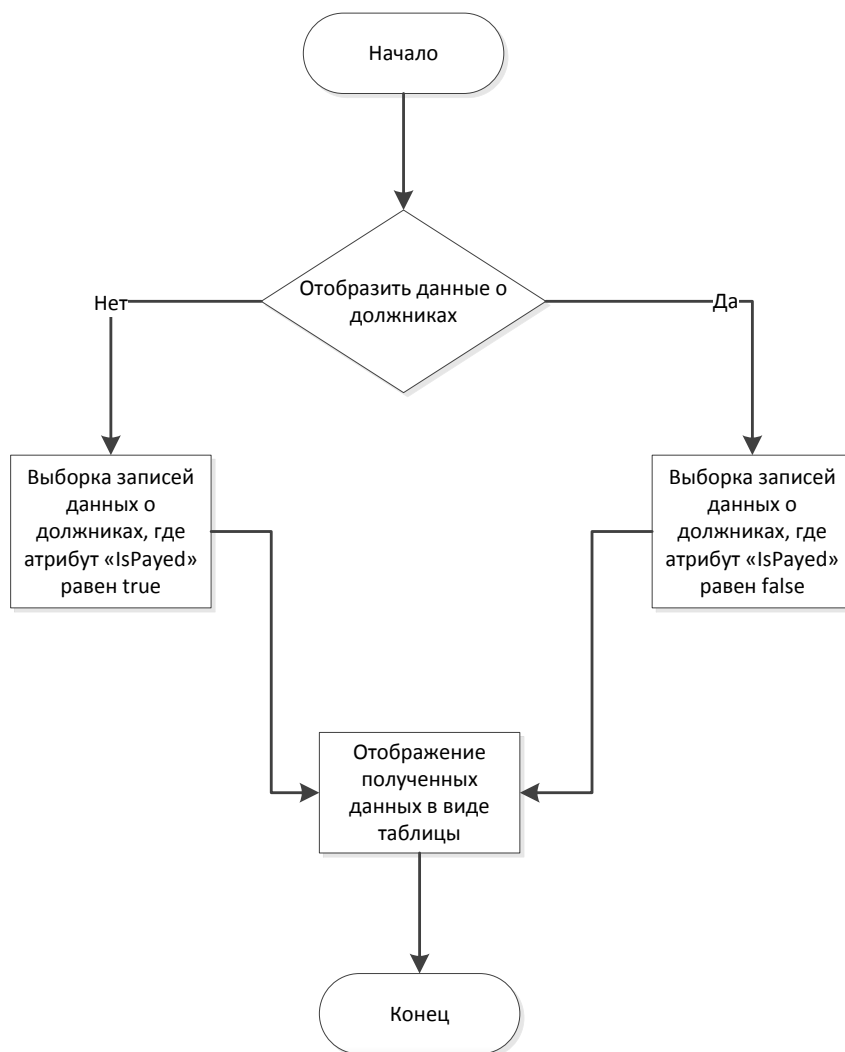


Рис. 3.1.1. Блок-схема алгоритма фильтрации организаций/лиц-штрафников

## 4 Прикладное программное обеспечение системы

### 4.1 Общая характеристика прикладного программного обеспечения

#### 4.1.1. Оценка соответствия разработанного web-приложения общим требованиям

1. Инструментальная среда разработки – Visual Studio 2012 Ultimate.
2. СУБД – SQL Server 2012.
3. Просмотр всех данных в табличном виде – на страницах web-приложения все данные представлены в табличном виде. Например, отображаемые данные о достижениях экологического фонда заключают в себе заголовок, описание и файл-фотографию.
4. Ввод новых записей/редактирование уже существующих записей - осуществляется при помощи специальных форм-бланков, отображаемых отдельными страницами в приложении.
5. Удаление записей – все таблицы поддерживают интерфейс удаления записей. В таблице [dbo].[Complaints] удаления не происходит, а вместо этого происходит сокрытие записей (реализовано в учебных целях).
6. Все операции реализуются корректно с сохранением логической и ссылочной целостности.
7. Фильтрация и поиск – реализованы без применения языка SQL, но при помощи механизма «точечной нотации» языка программирования высокого уровня C#. При этом при фильтрации пользователей системы имеется возможность установки нескольких критериев фильтрации.
8. Моделирование данных выполнено при помощи программного пакета Erwin Data Modeler. Основания, на которых производился выбор, подробно представлены в пункте 2.1.4 пояснительной записки.

#### 4.1.2. Оценка соответствия разработанного web-приложения дополнительным требованиям

В данном подразделе проводится оценка соответствия разработанного web-приложения уровню «5» по следующим критериям:

1. технология разработки web-системы – ASP .NET MVC;
2. количество таблиц в базе данных 15 (10) с суммарным числом атрибутов (76);
3. используемая технология доступа к данным – ADO.NET Entity Framework 5;
4. система безопасности – с учетом того, что в Entity Framework 5 при подходе Code First невозможно создать пользователей базы данных, было принято решение создать этих пользователей внутри web-системы, при этом каждая вид пользователя обладает своими возможностями доступа к данным;
5. технология ADO.NET Entity Framework поддерживает возможность создания индексов и значений по умолчанию;
6. в ходе разработки web-приложения для доступа к данным применялись такие возможности БД, как триггеры, транзакции, хранимые процедуры и функции;
7. имена файлов, классов, объектов являются значимыми и позволяют легко ориентироваться в структуре проекта и коде приложения;
8. пользовательский интерфейс программы разрабатывался средствами HTML 5, CSS 3, jQuery и библиотеки стилей Bootstrap. Наличие такого числа технологий говорит о его достаточной сложности. Интерфейс выполняет и главную функцию системы – минимизация усилий пользователя в процессе его работы с системой;
9. web-приложение поддерживает расширенную обработку исключительных ситуаций. Например, при переходе пользователя на несуществующую страницу отобразится страница-ошибка с кодом 404;

10. в разработанном приложении используется шаблон главной страницы. Это означает, что при переходе между страницами приложения основная часть (меню, форма авторизации, галерея, футер) не перерисовывается, а изменяется только информационная часть;
11. в ходе проектирования системы использовались многие элементы управления формы, применялись оригинальные решения при верстке макета сайта, вместо стандартных стилей элементов управления применялись стили из библиотеки Bootstrap;
12. выполненная работа является полностью оригинальной и влияние работ других студентов отсутствует.

## 4.2 Структура и состав прикладного программного обеспечения

### 4.2.1 Основные директории проекта приложения в VS 2012 Ultimate

В таблице 4.2.1.1 представлены основные директории структуры проекта с необходимыми описаниями.

Таблица 4.2.1.1

Основные директории проекта web-приложения

Название директории проекта	Описание
Properties	Директория, содержащая файл расширения .cs, в котором задаются основные параметры сборки приложения (copyright, trademark, culture и др.)
References	Директория содержит ссылки на сборки и библиотеки, включенные в проект.
App_Data	Директория, предназначенная для хранения данных приложения (в нашем случае подключенной базы данных).

App_Start	Директория, в которой хранятся файлы .cs, задающие параметры подключения js-скриптов, выбора главной страницы приложения и т.д.
Content	Директория, в которой хранятся стили, отвечающие за внешний вид приложения. Папка не претерпевала никаких изменений в ходе разработки проекта, так как автор создал свои директории, где хранил изображения, стили и js-скрипты.
Controllers	Директория, содержащая MVC контроллеры.
Images	Директория, в которой хранятся изображения и картинки, которые были использованы при разработке сайта.
Migrations	Директория, содержащая файлы-миграции, которые автор автоматически генерировал в консоли Package Manager Console для создания базы данных. Такой подход к созданию БД объясняется тем, что была выбрана методика Code First при работе с Entity Framework 5. Файл configuration.cs содержит переопределенный метод void Seed(FundContext), который каждый раз выполняется при обновлении БД. В этом методе можно определить значения по умолчанию для БД, а также

	триггеры, функции и хранимые процедуры.
Models	Директория, содержащая классы, на основе которых происходило построение базы данных.
Scripts	Директория, содержащая js-скрипты, которые были использованы при разработке web-приложения. Например, скрипт галереи, скрипты библиотеки стилей Bootstrap и jQuery.
Styles	Директория, в которой находятся стили, определяющие внешний вид пользовательского интерфейса.
Views	Директория, в которой хранятся представления страниц приложения, в том числе и шаблон мастер-страницы.

С точки зрения технологии паттерна MVC (Model-View-Controller) наибольший интерес представляет содержимое директорий Controllers, Views, Models, речь о которых пойдет в следующих трех частях пояснительной записки.

#### 4.2.2 Контроллеры web-приложения

**Контроллер (Controller)** с точки зрения паттерна программирования MVC – это набор функций, которые выполняются по запросу со стороны пользователя. В таблице 4.2.1.2 представлены контроллеры web-приложения и соответствующие описания к ним.

Таблица 4.2.1.2

Контроллеры разработанного web-приложения



Контроллер	Назначение
AccountController	В контроллере реализован функционал авторизации пользователя системы. Количество значимых строк кода: 61.
AchievementsController	Отображение и поиск данных страницы достижений экологического фонда. Количество значимых строк кода: 70.
EcologistsController	Отображение и поиск данных на странице экологов приложения. Количество значимых строк кода: 55.
ErrorController	Отображение страниц-ошибок в случае возникновения исключительной ситуации. Количество значимых строк кода: 9.
HomeController	Отображение главной страницы сайта экологического фонда. Количество значимых строк кода: 3.
PartnersController	Отображение и поиск данных на странице организаций-партнеров. Количество значимых строк кода: 44.
ProblemsController	Отображение и поиск данных на странице экологических проблем web-приложения. Количество значимых строк кода: 23.
RegistrationController	В контроллере реализован функционал регистрации нового пользователя системы. Количество значимых строк кода: 54.
RoomAdministratorController	В контроллере реализован функционал

	администраторской секции приложения. Наиболее интересная часть - отображение пользователей системы по ролям и их фильтрация. Количество значимых строк кода: 163.
RoomEcologistController	В контроллере реализован функционал секции пользователя-эколога. Наиболее интересная часть – реализация механизма создания экологической проблемы. Количество значимых строк кода: 61.
RoomPartnerController	В контроллере реализован функционал секции представителя организации-партнера. Количество значимых строк кода: 11.
RoomSecretaryController	В контроллере реализован функционал секции секретаря системы, включая фильтрацию организаций/должников два из трех запросов системы. Количество значимых строк кода: 154.
RoomUserController	В контроллере реализован функционал секции авторизованного пользователя системы. Количество значимых строк кода: 32.
SectionsController	Отображение страницы экологических секций, поиск данных, а также запрос по экологическим секциям. Количество значимых строк кода: 70.

### 4.2.3 Модели web-приложения

**Модель (Model)** – это класс, заключающий в себя поля-свойства и отношения, которые используются для генерации базы данных.

В проекте присутствует 13 моделей, которые подробно рассмотрены в таблице 4.2.3.1.

Таблица 4.2.3.1

Модели web-приложения

Модель	Назначение
Achievement	Описание сущности «Достижение» базы данных. В классе модели определены связи между данной сущностью и сущностями «Администратор» и «Экологическая проблема».
Complaint	Описание сущности «Жалоба» базы данных. В классе определены связи между данной сущностью и сущностями «Организация/Лицо-штрафник», «Пользователь» и «Экологическая проблема»
Council	Описание сущности «Экологический совет» базы данных. Определены связи между данной сущностью и сущностями «Экологическая проблема» и «Эколог».
EcologicalProblem	Описание сущности «Экологическая проблема» базы данных. Определены связи между данной сущностью и сущностями «Жалоба», «Эколог» и «Достижение».

OrganisationDebtor	Описание сущности «Организация/Лицо-штрафник». Определены связи между данной сущностью и сущностями «Жалоба» и «Секретарь»
PartnerShipRequest <sup>1</sup>	Описание сущности «Запрос на сотрудничество».
Section	Описание сущности «Экологическая секция». Определены связи между данной сущностью и сущностями «Эколог» и «Пользователь системы».
User	Описание сущности «Пользователь». Следует заметить, что от класса User.cs наследуются остальные типы пользователей системы, тем самым образуя связь типа «Есть».
RankUser	Описание сущности «Пользователь системы». Связь типа «Есть» с сущностью «Пользователь»
Secretary	Описание сущности «Секретарь». Связь типа «Есть» с сущностью «Пользователь».
Ecologist	Описание сущности «Эколог». Связь типа «Есть» с сущностью «Пользователь».
Administrator	Описание сущности «Администратор». Связь типа «Есть» с сущностью «Пользователь».
Partner	Описание сущности «Организация-

<sup>1</sup> Об использовании этой сущности более подробно рассказывается в пункте 4.3

	партнер (представитель)». Связь типа «Есть» с сущностью «Пользователь».
--	---

#### 4.2.4 Контекст данных web-приложения

Для связи созданной модели данных в приложении определен специальный класс **FundContext**, наследуемый от стандартного класса Entity Framework **DbContext**. Этот класс используется повсеместно в разработанном приложении для доступа к данным, хранящимся в базе данных. Объект класса **FundContext** обладает рядом специальных методов для извлечения, сохранения новых и измененных данных.

#### 4.2.5 Представления web-приложения

**Представление (View)** – файлы, состоящие из HTML-разметки с вставками программного кода на языке C#, другими словами это то, что пользователь видит в окне своего браузера. В ASP.NET MVC 4 для того чтобы осуществить вставку кода в разметку представления необходимо использовать один из двух движков ASPX и Razor. На самом деле разница невелика и зачастую выбор зависит от предпочтения самого разработчика. Мною был выбран движок Razor, потому что он подробнее описан в литературе по MVC и на форумах разработчиков.

В разработанном web-приложении получилось достаточно большое число представлений, поэтому при описании они будут разбиты по принципу «контроллер и его представления».

1. Контроллер **Account** имеет единственное представление в виде мастер-страницы (\_masterPage.cshtml). Это обуславливается тем, что форма авторизации находится на всех страницах web-приложения.
2. Контроллер **Achievement** определяет функциональность представления AchievementsPage.cshtml, где отображаются достижения экологического фонда.

3. Контроллер **Ecologists** определяет функциональность представления EcologistsPage.cshtml, где отображаются сведения об экологах-специалистах фонда.
4. Контроллер **Home** отображает содержание главной страницы сайта Index.cshtml.
5. Контроллер **Partners** отображает содержание и определяет функциональность представления PartnersPage.cshtml.
6. Контроллер **Problems** отображает содержание страницы экологических проблем (ProblemsPage.cshtml), а также определяет ее функционал.
7. Контроллер **Sections** отображает содержание страницы экологических секций (SectionsPage.cshtml) и определяет ее функционал.
8. Страницы, функциональность которых поддерживает контроллер **Registration**, показаны в таблице 4.2.5.1.

Таблица 4.2.5.1

Представления, поддерживаемые контроллером Registration

Название представления	Тип	Назначение
SignUpChoice.cshtml	Промежуточная страница	На данной странице производится выбор регистрации для разных категорий пользователей
SignUpFormEcologist.cshtml	Create <sup>2</sup>	Регистрационная форма для эколога
SignUpFormPartner.cshtml	Create	Регистрационная форма для организации-партнера
SignUpFormRankUser.cshtml	Create	Регистрационная

<sup>2</sup> При создании нового представления ему можно задать один из шаблонов Create, Details, Empty, Delete, Edit и List. Применение такой техники построения сайта значительно сокращает время разработки.

		форма для пользователя системы
RegistrationResult.cshtml	Страница-сообщение	Страница, на которую осуществляется переход в случае успешного прохождения регистрации

9. Страницы, функциональность которых поддерживает контроллер RoomAdministrator, показаны в таблице 4.2.5.2.

Таблица 4.2.5.2

Представления, поддерживаемые контроллером RoomAdministrator

Название представления	Тип	Назначение
AdministratorRoom.cshtml	Страница-меню	Страница, где администратору системы предоставляется выбор между его дальнейшими действиями. Другими словами – меню выбора.
CreateNews.cshtml	Create	Создание новости о достижении.
CreateSection.cshtml	Create	Создание экологической секции.
EditNews.cshtml	Edit	Редактирование новости о достижении.
EditSection.cshtml	Edit	Редактирование данных экологической секции.

SystemNewsCreation.cshtml	Details	Отображение списка экологических проблем, по которым можно создать новость.
SystemSections.cshtml	Details	Отображение списка созданных секций.
SystemUsers.cshtml	Details	Отображение списка пользователей системы.

10. Страницы, функциональность которых поддерживает контроллер **RoomEcologist**, показаны в таблице 4.2.5.3.

Таблица 4.2.5.3

Представления, поддерживаемые контроллером RoomEcologist

Название представления	Тип	Назначение
EcologistRoom.cshtml	Страница-меню	Страница, где экологу-специалисту предоставляется выбор между его дальнейшими действиями. Другими словами – меню выбора.
Councils.cshtml	Details	Список экологических советов, на которые эколог может зарегистрироваться или наоборот deregистироваться.



CreateComplaint.cshtml	Create	Создание жалобы на экологическое нарушение.
CreateProblem.cshtml	Edit	Создание экологической проблемы на основе жалобы
Complaints.cshtml	Edit	Отображение списка жалоб.

11. Страницы, функциональность которых поддерживает контроллер **RoomPartner**, показаны в таблице 4.2.5.4.

Таблица 4.2.5.4

Представления, поддерживаемые контроллером RoomPartner

Название представления	Тип	Назначение
PartnerRoom.cshtml	Страница-меню	Страница, где представителю организации-партнера предоставляется выбор между его дальнейшими действиями. Другими словами – меню выбора.
CreateComplaint.cshtml	Create	Создание жалобы на экологическое нарушение.

12. Страницы, функциональность которых поддерживает контроллер **RoomSecretary**, показаны в таблице 4.2.5.5.

Таблица 4.2.5.5

Представления, поддерживаемые контроллером RoomSecretary

Название представления	Тип	Назначение
SecretaryRoom.cshtml	Страница-меню	Страница, где секретарю предоставляется выбор между его дальнейшими действиями. Другими словами – меню выбора.
Councils.cshtml	Details	Список созданных экологических советов, на которые открыта регистрация.
Requests.cshtml	Details	Список заявок на регистрацию от представителей организаций-партнеров.
Debtors.cshtml	Details	Отображение списка организаций-штрафников.
CreateCouncil.cshtml	Create	Создание нового экологического совета.
CreateDebtor.cshtml	Create	Создание записи об организации-штрафнике на основе полученной жалобы.
EditCouncil.cshtml	Edit	Редактирование записи экологического совета.
EditDebtor.cshtml	Edit	Редактирование записи об организа-

		ции/лице-штрафнике.
--	--	---------------------

13. Страницы, функциональность которых поддерживает контроллер **RoomUser**, показаны в таблице 4.2.5.6.

Таблица 4.2.5.6

Представления, поддерживаемые контроллером RoomUser

Название представления	Тип	Назначение
UserRoom.cshtml	Страница-меню	Страница, где пользователю систему предоставляется выбор между его дальнейшими действиями. Другими словами – меню выбора.
Complaint.cshtml	Create	Создание жалобы на экологическое нарушение.
Sections.cshtml	Details	Список экологических секций, на которые пользователь может зарегистрироваться.

14. Страницы, функциональность которых поддерживает контроллер **Error**, показаны в таблице 4.2.5.7.

Таблица 4.2.5.4

Представления, поддерживаемые контроллером RoomPartner

Название представления	Тип	Назначение
General.cshtml	Страница-ошибки	Страница, появляю-

		щаяся в случае внутренней ошибки web-приложения.
Http404.cshtml	Страница-ошибки	Страница, появляющаяся в случае несуществования страницы, запрашиваемой пользователем.
Http403.cshtml	Страница-ошибки	Страница, появляющаяся в случае запрета доступа к определенному ресурсу.

#### 4.3 Особенности реализации и сопровождения

Разработанная система имеет ряд важных особенностей, которые следует учитывать при ее дальнейшем сопровождении:

1. Сущность PartnershipRequests (Запрос на сотрудничество) в системе не используется. На этапе написания технического задания предполагалось хранить заявки организаций-штрафников в соответствующей этой сущности таблице. В процессе разработки было найдено другое решение, основанное на использование поля-флага для объекта сущности «Организация-партнер». При регистрации нового партнера, значение этого поля, по умолчанию устанавливалось в значение false. Секретарь системы из своей «комнаты» может принять заявку, тем самым установив это значение в true и разрешив вход в систему. В будущем планируется улучшить функционал запроса на сотрудничество, поэтому удалять сущность пока не имеет смысла.

2. Особое внимание следует обратить на то, что при малейшем изменении моделей сущностей базы данных необходимо применять механизм миграций, так как используемый подход к построению базы данных – Code First, предполагающий полностью ручное создание сущностей данных.
3. При проектировании связей типа «Многие-ко-многим» необходимо быть предельно внимательным в плане разрешения каскадного удаления записей таблиц. Например, в ходе разработки системы было потрачено много времени на установление такого типа связи между сущностями «Экологический совет» и «Эколог». При удалении «Экологического совета», система выдавала ошибку о том, что удаление объектов сущности «Эколог» может повлечь за собой непоправимые последствия. Это достаточно легко объясняется тем, что сущность «Эколог» также связана с сущностями «Экологическая секция» и «Экологическая проблема», поэтому удаление совета могло повредить и объекты этих сущностей. Решение – запрет каскадного удаления, но в некоторых особых случаях может применяться так называемое сокрытие данных при помощи специальных полей.
4. Рассмотренная в пункте 4 техника сокрытия данных была применена для случая удаления жалобы на экологическое нарушение. Это было сделано в учебных целях.

## 5 Руководство пользователя

### 5.1 Общие сведения

Разработанное web-приложение предназначено для автоматизации многих процессов работы экологического фонда. С системой могут работать 6 типов пользователей:

1. администратор;
2. секретарь;
3. специалист-эколог;
4. представитель организации-партнера;
5. авторизованный пользователь;
6. неавторизованный пользователь.

Уровень подготовки пользователей предусматривает навыки работы с ПК и с программами типа интернет-браузер.

### 5.2 Порядок и особенности работы

В текущей части будут описаны режимы работы с системой от лица каждого типа пользователя и подробно рассмотрен ее интерфейс.

#### 5.2.1 Описание основных частей интерфейса системы

Первое, что видит на своем экране пользователь разработанной системы – это главная страница приложения. Она представлена на рисунке 5.2.1.1.

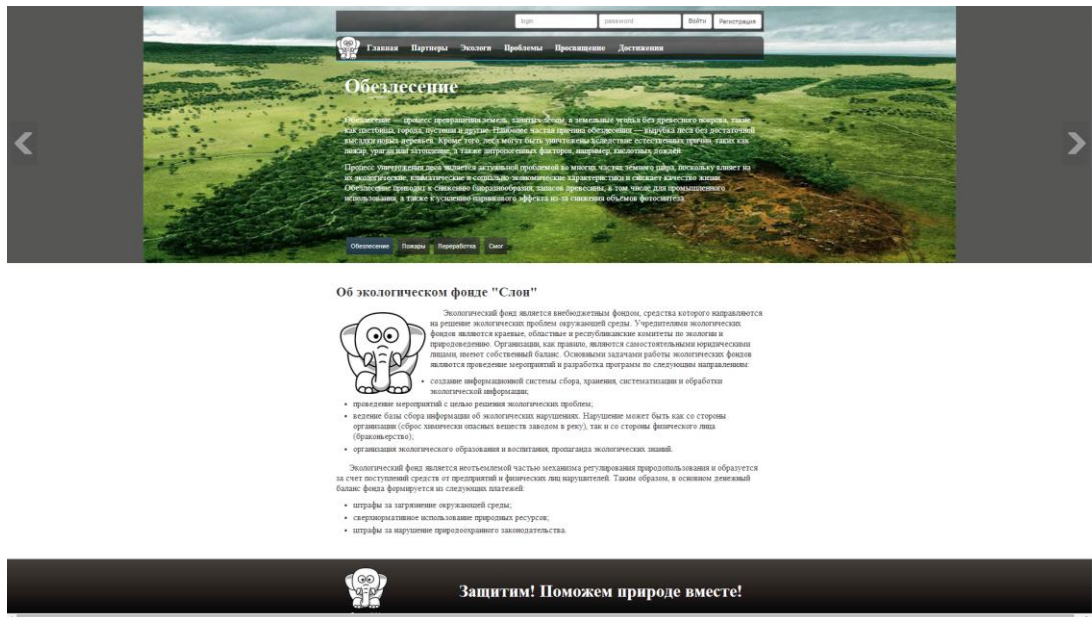


Рис 5.2.1.1 Главная страница web-приложения

Наиболее важная часть интерфейса располагается вверху окна – это блок, состоящий из формы авторизации и ленточного меню (представлен на рисунке 5.2.1.2).

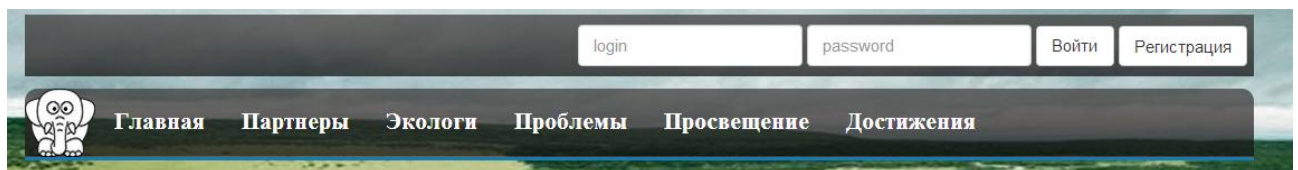


Рис 5.2.1.2 Форма авторизации и лента меню.

Если пользователь зарегистрирован, то он может осуществить вход на сайт и получить доступ к дополнительным функциям, разрешенным для его типа пользователя.

Меню сайта представляет собой группу ссылок, при нажатии на которые можно перемещаться между страницами web-приложения.

Центральная часть приложения (см. рис. 5.2.1.3) является динамической. Это означает, что при перемещении между страницами она будет меняться.

### Об экологическом фонде "Слон"



Экологический фонд является внебюджетным фондом, средства которого направляются на решение экологических проблем окружающей среды. Учредителями экологических фондов являются краевые, областные и республиканские комитеты по экологии и природоохране. Организации, как правило, являются самостоятельными юридическими лицами, имеют собственный баланс. Основными задачами работы экологических фондов являются проведение мероприятий и разработка программ по следующим направлениям:

- создание информационной системы сбора, хранения, систематизации и обработки экологической информации;
- проведение мероприятий с целью решения экологических проблем;
- ведение базы сбора информации об экологических нарушениях. Нарушение может быть как со стороны организации (сброс химически опасных веществ заводом в реку), так и со стороны физического лица (браконьерство);
- организация экологического образования и воспитания, пропаганда экологических знаний.

Экологический фонд является неотъемлемой частью механизма регулирования природопользования и образуется за счет поступлений средств от предприятий и физических лиц нарушителей. Таким образом, в основном денежный баланс фонда формируется из следующих платежей:

- штрафы за загрязнение окружающей среды;
- сверхнормативное использование природных ресурсов;
- штрафы за нарушение природоохранного законодательства.

Рис 5.2.1.3 Центральная часть страницы

## 5.2.2 Неавторизованный пользователь системы

Неавторизованным пользователем информационной системы экологического фонда может быть как случайный посетитель, так и человек, который планирует зарегистрироваться и посещать экологические кружки, работать в экологическом фонде как специалист-эколог, при наличии соответствующего образования, или быть представителем организации, которая хочет наладить сотрудничество с фондом.

Первое, что, по мнению автора, должен сделать такой пользователь, ознакомиться с представленной на сайте информацией. Перемещаясь между ссылками ленточного меню, пользователь может посетить следующие страницы:

1. «Главная страница» (см. рис. 5.2.1.1).
2. Партнеры – на странице (см. рис. 5.2.2.1) отображена основная информация об организациях-партнерах, сотрудничающих с экологическим фондом.



## Организации-партнеры

<div> <input type="text" value="Поиск..."/> <input type="button" value="Искать"/> </div>				
Представитель ФИО	Компания	Адрес	Email	О компании
Голобокова Анна Андреевна	AnnCompanyGroup	Ульяновск	golobokova_ann@gmail.ru	Мы компания, занимающаяся помощью природе.

Рис 5.2.2.1 Информация об организациях-партнерах

3. «Экологи» – на странице (рис. 5.2.2.2) отражена основная информация о специалистах, работающих на экологический фонд. Также есть поисковая строка, при использовании которой можно составить «запрос» и получить интересующую информацию (рис. 5.2.2.3).

## Экологи нашего фонда

<div> <input type="text" value="Поиск..."/> <input type="button" value="Искать"/> </div>						
ФИО	Дата регистрации	Email	Область интересов	Образование	Регион базирования	Публикации
Харитонов Алексей Иванович	20.05.2014	demarvel@yandex.ru	Животные и растения. Lorem Ipsum - это текст-"рыба", часто используемый в печати и веб-дизайне. Lorem Ipsum является стандартной рыбой для текстов на латинице с начала XVI века. В то время некий безымянный печатник создал большую коллекцию размеров и форм шрифтов, используя Lorem Ipsum для распечатки образцов.	УлГТУ. Факультет информационных систем и технологий. Высшее	Калининградская область	Загрязнение реки Амур
Муравьев Игорь Игоревич	20.05.2014	muravey@yandex.ru	Океан, море и озера. Lorem Ipsum - это текст-"рыба", часто используемый в печати и веб-дизайне. Lorem Ipsum является стандартной рыбой для текстов на латинице с начала XVI века. В то время некий безымянный печатник создал большую коллекцию размеров и форм шрифтов, используя Lorem Ipsum для распечатки образцов.	УлГТУ. Факультет информационных систем и технологий. Высшее	Краснодарский край	Нет публикаций
Смеречинский Сергей Орестович	20.05.2014	smerechinskiy@yandex.ru	Интересуюсь многими сферами биологии и экологии. Lorem Ipsum - это текст-"рыба", часто используемый в печати и веб-дизайне.	УлГТУ. Факультет информационных систем и технологий. Высшее	Смоленская область	Лесные пожары в Сибири

Рис 5.2.2.2 Информация об экологах-специалистах

<div> <input type="text" value="Харитонов"/> <input type="button" value="Искать"/> </div>						
ФИО	Дата регистрации	Email	Область интересов	Образование	Регион базирования	Публикации
Харитонов Алексей Иванович	20.05.2014	demarvel@yandex.ru	Животные и растения. Lorem Ipsum - это текст-"рыба", часто используемый в печати и веб-дизайне. Lorem Ipsum является стандартной рыбой для текстов на латинице с начала XVI века. В то время некий безымянный печатник создал большую коллекцию размеров и форм шрифтов, используя Lorem Ipsum для распечатки образцов.	УлГТУ. Факультет информационных систем и технологий. Высшее	Калининградская область	Загрязнение реки Амур

Рис 5.2.2.3 Пример использования поисковой строки

4. Чтобы ознакомиться с экологическими проблемами, которые в данный момент разрабатываются фондом необходимо перейти на страницу «Проблемы» (см. рис. 5.2.2.4).

### Экологические проблемы

<div> <input type="text" value="Поиск..."/> <input type="button" value="Искать"/> </div>			
Название	Описание	Сумма	Ход решения
Загрязнение реки Амур	Проблема загрязнения реки Амур. Lorem Ipsum - это текст-"рыба", часто используемый в печати и веб-дизайне. Lorem Ipsum является стандартной рыбой для текстов на латинице с начала XVI века. В то время некий безымянный печатник создал большую коллекцию размеров и форм шрифтов, используя Lorem Ipsum для распечатки образцов.	2000000,00	Не решена
Загрязнение озера Байкал	Проблема загрязнения озера Байкал. Lorem Ipsum - это текст-"рыба", часто используемый в печати и веб-дизайне. Lorem Ipsum является стандартной рыбой для текстов на латинице с начала XVI века. В то время некий безымянный печатник создал большую коллекцию размеров и форм шрифтов, используя Lorem Ipsum для распечатки образцов.	2000000,00	Не решена
Лесные пожары на Алтае	Проблема лесных пожаров. Lorem Ipsum - это текст-"рыба", часто используемый в печати и веб-дизайне. Lorem Ipsum является стандартной рыбой для текстов на латинице с начала XVI века. В то время некий безымянный печатник создал большую коллекцию размеров и форм шрифтов, используя Lorem Ipsum для распечатки образцов.	2000000,00	Решена
Лесные пожары в Сибири	Лесной пожар в Сибири. Срочно требуется помощи фонда "Слон". Идем на помощь	100000000,00	Решена

Рис 5.2.2.4 Страница «Проблемы» web-приложения

5. На странице «Просвещение» (см. рис. 5.2.2.5) можно узнать, какие курсы преподаются, а также получить полную информацию о них.

### Секции

<div> <input type="text" value="Эколог..."/> <input type="text" value="Другое..."/> <input type="button" value="Искать"/> </div>					
Название	Описание	Начало занятий	Число занятий	Свободные места	Преподаватель
Общий курс экологии. Часть 1.	Очень интересен курс. Подойдет для тех кто только начинается знакомиться с экологией. Ведет просто замечательный преподаватель	25.05.2014	15	39	Загайчук Иван Анатольевич
Общий курс экологии. Часть 2.	Очень интересен курс. Продолжение предыдущего курса. Подойдет для продвинутых пользователей. Ведет также просто замечательный преподаватель	30.05.2014	10	20	Харитонов Алексей Иванович
Экология и человек	Интересный курс как для детей, так и для взрослых	18.07.2014	10	20	Загайчук Иван Анатольевич
Экология и природа	Интереснейший курс	20.09.2014	10	15	Загайчук Иван Анатольевич

Рис 5.2.2.5 Страница «Просвещение» web-приложения

6. «Достижения» – это своеобразная новостная лента (см. рис. 5.2.2.6), рассказывающая о достижениях экологического фонда.

## Достижения


Название	Описание	О проблеме	Фото	Опубликовано
Река Амур!	При непосредственной поддержке фонда был потушен большой пожар в Алтайских лесах. Фонд удостоился правительственной награды.	Загрязнение реки Амур	Нет фото	Моисеев Владислав Валерьевич
Пожар на Алтае потушен!	При непосредственной поддержке фонда был потушен большой пожар в Алтайских лесах. Фонд удостоился правительственной награды.	Лесные пожары на Алтае		Моисеев Владислав Валерьевич
Пожар в Сибири потушен!	При непосредственной поддержке фонда был потушен большой пожар в Сибирских лесах. Фонд удостоился правительственной награды.	Лесные пожары в Сибири	Нет фото	Желепов Алексей Сергеевич

Рис 5.2.2.6 Страница «Достижения» web-приложения

Неавторизованный пользователь также имеет возможность зарегистрироваться, для этого ему необходимо осуществить переход на страницу регистрации, нажав на кнопку «Регистрация» рядом с формой авторизации (см. рис. 5.2.1.2).

После перехода открывается страница, представленная на рисунке (5.2.2.7)

### Регистрация

#### Для пользователя

Хотите узнать побольше об экологии? Записаться на участие в экологических секциях? Вам сюда.

#### Для эколога

Если вы эколог-специалист и хотите сотрудничать с нашим фондом - Вам сюда.

#### Для организации партнера

Вы организация, заинтересованная в сотрудничестве? Представитель организации может зарегистрироваться здесь. (P.S. после регистрации придется немного подождать. Наш секретарь проверит вашу заявку)

Рис 5.2.2.7 Страница выбора регистрации web-приложения

Рассмотрим вариант регистрации «Для пользователя». Для этого нажмем на кнопку «Зарегистрироваться» – откроется форма (см. рис. 5.2.2.8) и заполним ее необходимыми данными.

## Заполните предложенную регистрационную форму (для пользователя)

Введите ваше имя:	<input type="text" value="Иван"/>
Введите вашу фамилию:	<input type="text" value="Иванович"/>
Введите ваше отчество:	<input type="text" value="Иванов"/>
Укажите ваш пол:	<input type="text" value="Мужской"/>
Введите вашу дату рождения:	<input type="text" value="19.04.1994"/>
Придумайте логин:	<input type="text" value="ivan"/>
Придумайте пароль:	<input type="password" value="...."/>
Введите адрес электронной почты:	<input type="text" value="ivan@mail.ru"/>
Введите информацию себе:	<input type="text" value="Интересуюсь экологией...."/>

Рис 5.2.2.8 Страница регистрации пользователя

При нажатии на кнопку «Зарегистрироваться», в случае успешной регистрации будет произведен трансфер на страницу, представленную на рисунке 5.2.2.9, и создана новая учетная запись пользователя.

## Наши поздравления - вы успешно зарегистрировались!

Рис 5.2.2.9 Сообщение о завершении регистрации

Для входа в систему необходимо ввести логин и пароль в форму авторизации и нажать на кнопку «Войти». При успешном входе отобразится приветственное сообщение (см. рис. 5.2.2.10).

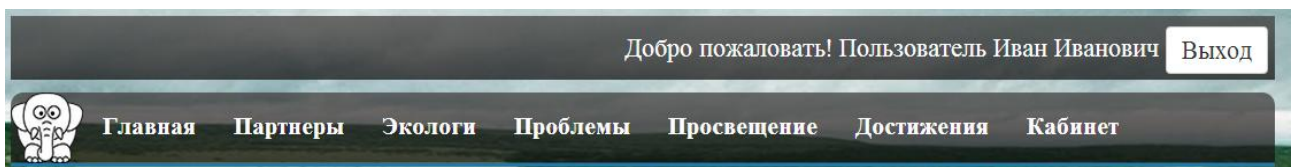


Рис 5.2.2.10 Сообщение об успешном входе в систему

Остальные виды регистрации работают аналогичным образом. Они также предусматривают заполнение регистрационной формы и ее отправку. Но в

случае регистрации организации-партнера вход в систему будет запрещен до тех пор, пока секретарь не рассмотрит и не примет заявку (подробно рассматривается в пункте 5.2.5).

#### 5.2.3 Представитель организации-партнера

#### 5.2.4 Администратор

#### 5.2.5 Секретарь

#### 5.2.6 Эколог-специалист

#### 5.2.7 Авторизованный пользователь системы

Описываются режимы работы программного обеспечения, порядок действий пользователя при выполнении основных операций. Изложение может следовать общей логике и порядку работы с системой либо строиться по иному принципу. Должно иллюстрироваться screenshot'ами соответствующих Web-форм. Объём этого подраздела – не менее 5 стр.

### 5.3 Исключительные ситуации

Отмечаются исключительные (ошибочные) ситуации, которые могут возникнуть при использовании прикладного программного обеспечения из-за некорректных действий пользователя, отсутствия или нехватки необходимых ресурсов, появляющиеся при этом сообщения. Даются необходимые пояснения и конкретные рекомендации по разрешению или предотвращению таких ситуаций. Должно быть рассмотрено не менее 4 исключительных ситуаций.

## **Заключение**

В заключении следует в краткой форме отразить полноту и качество реализации и оформления работы, её соответствие утверждённому техническому заданию, предъявляемым требованиям. Указать имеющиеся недочёты, охарактеризовать трудности, с которыми пришлось столкнуться и степень их преодоления.

## **Список использованных источников**

Список должен содержать библиографическое описание не менее 4-х источников, использованных при выполнении курсовой работы и оформленных согласно **ГОСТ 7.1-2003**. Не следует включать в него источники, которые реально не использовались (об этом можно будет судить из анализа, данного во введении). Не менее 2-х источников должно быть посвящено предметной области.

## **Приложение А. Исходные тексты программных модулей**

Приложение оформляется как продолжение пояснительной записки и должно включать комментированные исходные тексты программных модулей (на языке C#).