

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«ПОЛОЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет информационных технологий
Кафедра технологий программирования

ДОПУЩЕН К ЗАЩИТЕ
Заведующий кафедрой
_____ О.Н. Петрович
подпись инициалы и фамилия
« ____ » _____ 20 ____ г.

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
ДИПЛОМНОГО ПРОЕКТА**

« Веб-приложение «Служба вопрос-ответ QandA-Online» для
социального взаимодействия пользователей »
наименование темы

Специальность 1-40 01 01 «Программное обеспечение информационных технологий»
код, наименование специальности

Специализация 1-40 01 01 01 «Веб-технологии и программное обеспечение мобильных систем»
код, наименование специализации

Студент-дипломник
группы 16-ИТ-1 _____
номер подпись, дата инициалы и фамилия

Руководитель _____
подпись, дата инициалы и фамилия, уч. степень, звание

Консультанты:

по разделу Экономическая часть _____
наименование раздела подпись, дата инициалы и фамилия, уч. степень, звание

по разделу Нормоконтроль _____
наименование раздела подпись, дата инициалы и фамилия, уч. степень, звание

по разделу _____
наименование раздела подпись, дата инициалы и фамилия, уч. степень, звание

по разделу _____
наименование раздела подпись, дата инициалы и фамилия, уч. степень, звание

по разделу _____
наименование раздела подпись, дата инициалы и фамилия, уч. степень, звание

по разделу _____
наименование раздела подпись, дата инициалы и фамилия, уч. степень, звание

по разделу _____
наименование раздела подпись, дата инициалы и фамилия, уч. степень, звание

Объем проекта:

пояснительная записка - 66 страниц;

графическая часть - 4 листов;

магнитные (цифровые) носители - 1 единиц;

Новополоцк 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«ПОЛОЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Кафедра _____ *технологий программирования* _____

УТВЕРЖДАЮ

Заведующий кафедрой

_____ *Петрович О.Н.* _____
подпись (фамилия, инициалы)

« _____ » _____ 20 20 г.

ЗАДАНИЕ НА ДИПЛОМНЫЙ ПРОЕКТ

Студенту _____ *Дубовский Н.В.* _____
(фамилия, инициалы)

1. Тема дипломного проекта _____ *Веб-приложение «Служба вопрос-ответ QandA-Online»* _____
_____ для _____ *социального* _____
пользователей _____
(наименование темы)

Утверждена приказом руководителя учреждения высшего образования
от « 05 » _____ 05 _____ 20 20 г. № 256ув _____

2. Исходные данные к дипломному проекту _____
техническое задание от руководителя дипломного проекта

3. Перечень подлежащих разработке вопросов или краткое содержание расчетно-пояснительной записки _____

Введение

1. *Анализ исходных данных и постановка задачи*

2. *Проектирование программного обеспечения*

3. *Реализация и тестирование*

Заключение

4. Перечень графического материала (с точным указанием обязательных чертежей и графиков)

Схема базы данных

Экранные формы приложения

Диаграмма себестоимости программного обеспечения

Диаграмма распределения времени работы при разработке продукта

5. Консультанты по дипломному проекту с указанием относящихся к ним разделов

Экономическая часть – С.В. Измайлович, к.э.н., доцент

Нормоконтроль – Г.Ф. Конаплева

6. Примерный календарный график выполнения дипломного проекта

22 апреля 2020 50% дипломного проекта и пояснительной записки к нему

13 мая 2020 80% дипломного проекта и пояснительной записки к нему

3 июня 2020 100% дипломного проекта и пояснительной записки к нему

7. Дата выдачи задания «_23_» _____ 03_____ 20_20_ г.

8. Срок сдачи законченного дипломного проекта «_3_» _____ 06_____ 20_20_ г.

Руководитель _____

Д.В. Пяткин
(фамилия, инициалы)

Подпись обучающегося _____

Н.В. Дубовский
(фамилия, инициалы)

«_____» _____ 20_20_ г.

РЕФЕРАТ

ВЕБ-ПРИЛОЖЕНИЕ «СЛУЖБА ВОПРОС-ОТВЕТ QANDA-ONLINE»
ДЛЯ СОЦИАЛЬНОГО ВЗАИМОДЕЙСТВИЯ ПОЛЬЗОВАТЕЛЕЙ: дипломный
проект / Н.В. Дубовский. – Полоцк: ПГУ, 2020, - п.з. - 66 с., 28 рис., 25 табл., 9
источников, графический материал – 4 л. формата А4.

ВЕБ-ПРИЛОЖЕНИЕ, СЛУЖБА ВОПРОС-ОТВЕТ, ЛОГИЧЕСКАЯ
МОДЕЛЬ, ВЗАИМОДЕЙСТВИЕ ПОЛЬЗОВАТЕЛЕЙ, СОЦИАЛЬНАЯ
ПЛОЩАДКА

Объектом разработки является создание «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей.

Цель проекта: разработка веб-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей.

В процессе проектирования выполнены следующие разработки:

- дизайн сайта;
- база данных;
- подсистема регистрации;
- подсистема авторизации;
- подсистема вопрос-ответ;
- подсистема подписки;
- подсистема оповещения;
- подсистема администратора.

Областью возможного практического применения являются использование в коммерческих целях сайта для размещения рекламы.

Студент-дипломник подтверждает, что приведенный в дипломном проекте расчетно-аналитический материал объективно отражает состояние разрабатываемого объекта, все заимствованные из литературных и других источников теоретические и методологические положения и концепции сопровождаются ссылками на их авторов.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	7
1 АНАЛИЗ ИСХОДНЫХ ДАННЫХ И ПОСТАНОВКА ЗАДАЧИ.....	8
1.1 Описание предметной области.....	8
1.2 Сравнительный анализ аналогов и прототипов	9
1.3 Постановка задачи проектирования.....	13
1.4 Анализ требований к проекту	14
1.5 Выбор и обоснование средств и методов решения задач	15
1.6 Разработка технического задания	16
2 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	17
2.1 Функциональная структура приложения.....	17
2.2 Проектирование диаграммы вариантов использования	18
2.3 Информационное обеспечение проекта.....	18
2.4 Разработка структуры конфигурации	19
2.4 Разработка концепции пользовательского интерфейса	24
2.5 Безопасность и защита данных	24
3 РЕАЛИЗАЦИЯ И ТЕСТИРОВАНИЕ	25
3.1 Реализация функциональных частей программного обеспечения	25
3.2 Разработка программной документации	27
3.3 Тестирование.....	28
4 ЭКОНОМИЧЕСКАЯ ЧАСТЬ.....	31
4.1 Обоснование необходимости вывода продукта на рынок	31
4.2 Структура (этапы) работ по созданию программного обеспечения	31
4.3. Составление сметы затрат на разработку программного обеспечения....	33
4.4 Расчет экономического эффекта разработчика и пользователя (заказчика) программного обеспечения	41

					ДНВ.508190.009.ПЗ		
Изм	Лист	№ докум.	Подпись	Дата			
Разраб.		Дубовский Н.В..			Web-приложения «Служба во- прос-ответ QandA-Online» для социального взаимодействия пользователей	Лит.	Лист
Провер.		Пяткин Д.В.					5
Реценз.		Богущ Р.П.					66
Н. Контр.		Конаплева Г.Ф.				Учреждение образования «Полоцкий государственный университет», гр.16-ИТ-1	
Утверд.		Петрович О.Н.					

ЗАКЛЮЧЕНИЕ.....	46
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	47
ПРИЛОЖЕНИЕ А (обязательное) Техническое задание	48
ПРИЛОЖЕНИЕ Б (рекомендуемое) Диаграмма модели данных	52
ПРИЛОЖЕНИЕ В (обязательное) Описание программы	53
ПРИЛОЖЕНИЕ Г (обязательное) Программа и методика испытаний.....	55
ПРИЛОЖЕНИЕ Д (обязательное) Руководство оператора	59
ПРИЛОЖЕНИЕ Е (обязательное) Руководство администратора	64

ВВЕДЕНИЕ

В настоящее время возрастает роль и популярность веб-приложений. Это неудивительно, так как они обладают рядом преимуществ, а именно:

- пользователь не нуждается в установке на свою машину тяжеловесного программного обеспечения. Все, что требуется для полноценной работы — это браузер, обычно поставляемый вместе с операционной системой, и доступ в Интернет;
- устанавливая приложения на свой компьютер, поневоле приходится брать на себя обязанности администратора, что доставляет неопытным пользователям массу хлопот. В случае с браузерным приложением, фактически лежащем на сервере, беспокоиться об этом не придется;
- web-приложения не требовательны к ресурсам и не предъявляют никаких требований к аппаратной платформе;
- нет никаких проблем с поддержкой старых версий программ и обратной совместимостью;
- web-приложения позволяют своим пользователям быть по-настоящему мобильными.

					ДНВ.508190.009.ПЗ	Лист
						7
Изм.	Лист	№ докум.	Подпись	Дата		

1 АНАЛИЗ ИСХОДНЫХ ДАННЫХ И ПОСТАНОВКА ЗАДАЧИ

1.1 Описание предметной области

При появлении интернета как самого понятия люди начали приспосабливать его к различным сферам жизнедеятельности. Для более удобного взаимодействия клиента и сервера планируется разработать Web-приложение. По своей сути Web-приложение – это клиент серверное приложение, в которой клиент взаимодействует с web-приложением с помощью браузера. Конкретно данное приложение будет предлагать реализацию программы типа «Служба вопрос - ответ».

Система вопросов и ответов — вид веб-сайтов, позволяющий пользователям задать вопрос или ответить на уже заданные вопросы. Таким образом формируется социальная сеть, участником которой может стать каждый. Часто пользователи могут также голосовать за ответы, выделяя таким образом, по их мнению, правильный. В некоторых системах вопросов и ответов активные пользователи, чьи ответы получают большое количество голосов, получают статус экспертов, что помогает новичкам оценить компетентность полученных ответов.[7]

Бесспорно то, что в наши дни люди ценят каждую минуту своего времени. Каждый желает быть в курсе последних событий, вопросов и решений, которые в данный момент наиболее актуальны. Поэтому чтобы получить как можно быстрее необходимую информацию, они заходят в Интернет и задают свой вопрос онлайн.

Данное web-приложение создан для миллионов людей, которые ежедневно нуждаются в ответах на множество вопросов. Сервис вопросов позволяет узнать информацию на любую волнующую тему, а также показать свою компетентность в той или иной сфере. Это интересный и простой способ узнавать больше друг о друге. Ответы всегда разнообразны, от крайне увлекательных до порой очень весёлых и разблачающих.

Сервис вопросов и ответов QandA-Online – это удобный ресурс, чтобы задать вопрос онлайн, голосовать за понравившийся ответ и подписываться на интересных пользователей. Данное web-приложение — это место, где каждый может стать народным экспертом и делиться своими практическим и теоретическим жизненным опытом. Пользователи смогут заслужить репутацию и признание в благодарность за знания в тех областях, в которых они сильны.

Преимущество сервиса вопросов заключается в том, что обмениваясь информацией, пользователи делятся личными знаниями или предположениями, что делает каждую пару «вопрос - ответ» уникальной. Чтобы начать, пользователь сможет воспользоваться поиском по ключевым словам или меткам. Так он узнает, отвечал ли уже кто-то на аналогичный вопрос, найдено ли решение для похожей ситуации, предлагались ли какие-либо варианты действий. Также ему доступен просмотр наиболее популярных запросов и самых распространенных проблем.

Для того чтобы каждый смог задать вопрос, на сайте предусмотрена авторизация «в один клик», что с самого начала работы создает положительное впечатление и настраивает на позитивное и продуктивное общение.

1.2 Сравнительный анализ аналогов и прототипов

Перед тем, как приступить к реализации дипломного проекта, следует проанализировать существующие на данный момент программные продукты подобного характера и тематики. Это поможет сделать верные выводы о том, как необходимо создавать собственный программный продукт на основе достоинств и недостатков рассмотренных аналогов.

Любое Web-приложение имеет свои преимущества и недостатки. Таковые имеет и сайт otvet.mail.ru. Рассмотрим положительные аспекты данного сайта:

- в приложении имеется возможность задать вопрос, который будет виден всем пользователям и посетителям данного сайта, что позволяет увеличить охваты пользователей и вероятность ответа на вопрос;
- пользователь может поставить положительную оценку или же пожаловаться на ответ, что представлено на рисунке 1.1. На основании данных показателей формируется рейтинг, который отражается на рейтинге ответа, а также на общем рейтинге пользователя, оставившего ответ;

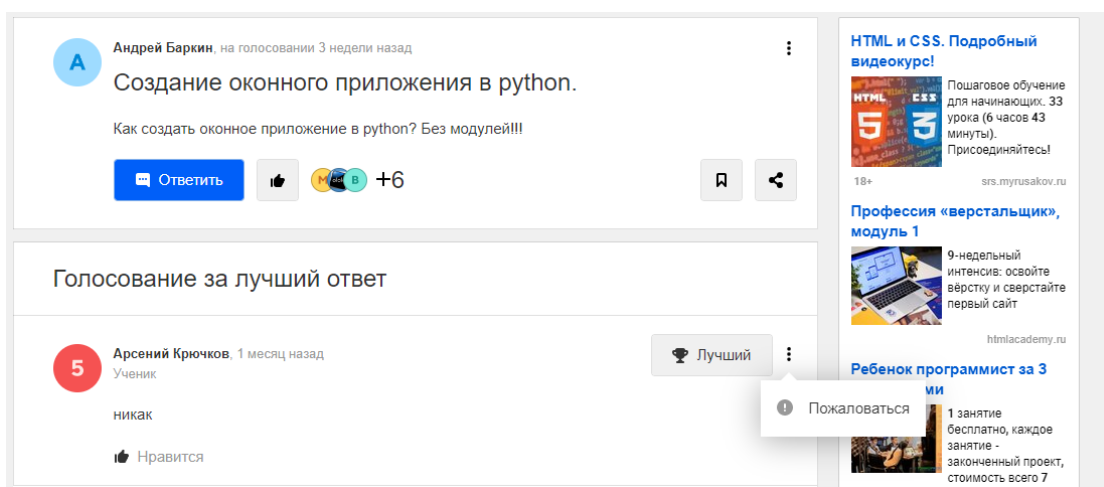


Рисунок 1.1 – Вариант оценки ответа

- реализована подписка на пользователей, представлено на рисунке 1.2;
- так же есть возможность выбора категории заданного вопроса, что существенно облегчает поиск, что можно увидеть на рисунке 1.3;
- есть возможность прикрепить медиафайлы к вопросу, такие как фото и видео.

Приступим к недостаткам:

- низкая квалификация заинтересованной аудитории;
- лишняя информация в виде баннеров и навязчивой рекламы, что

можно наблюдать на рисунке 1.1 и рисунке 1.3.

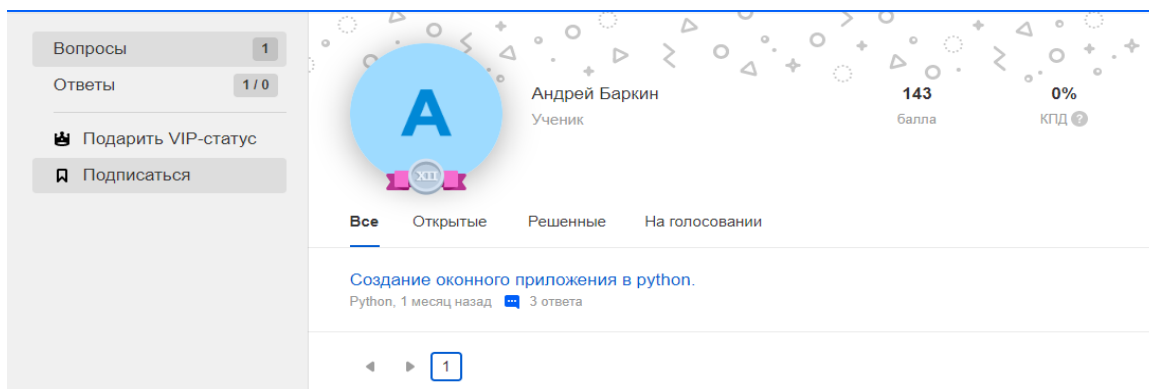


Рисунок 1.2 – Подписка на пользователя

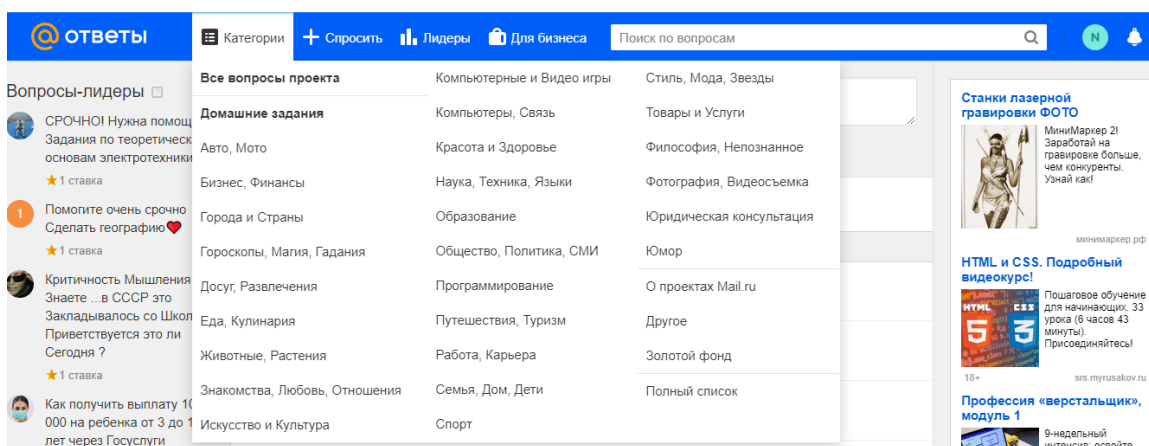


Рисунок 1.3 – Категории вопросов

Теперь рассмотрим web-сайт Яндекс.Кью и выделим следующие достоинства:

— реализована подписка на категории и пользователей, что позволяет пользователю управлять контентом, это отображено на рисунке 1.4 и рисунке 1.5;

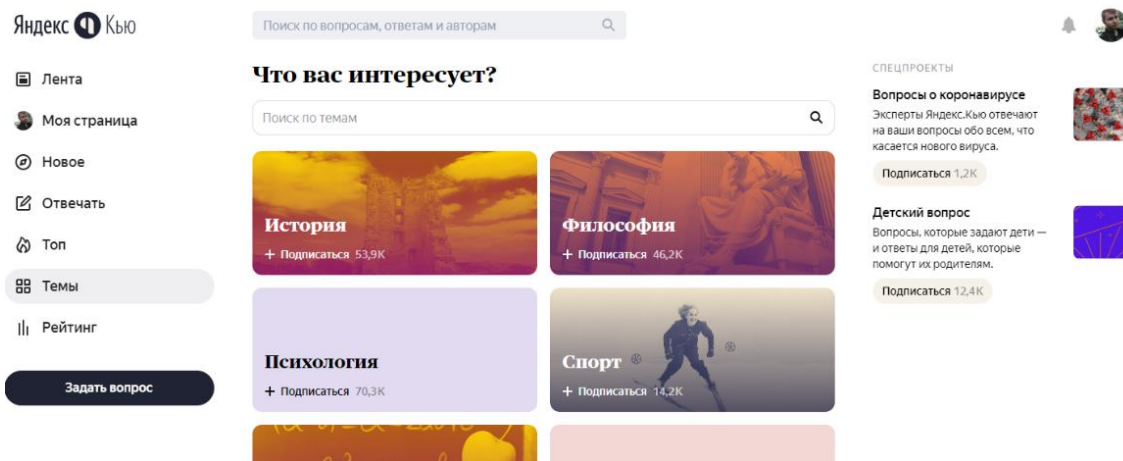


Рисунок 1.4 – Подписка на категории

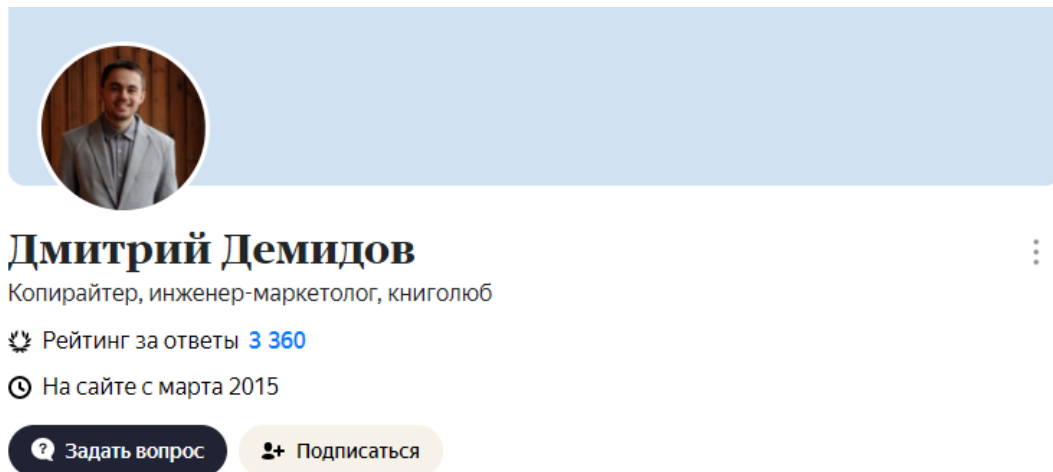


Рисунок 1.5 – Подписка на пользователя

- имеется возможности задать анонимный вопрос, что позволяет пользователю сохранить свою конфиденциальность, что представлено на рисунке 1.6;

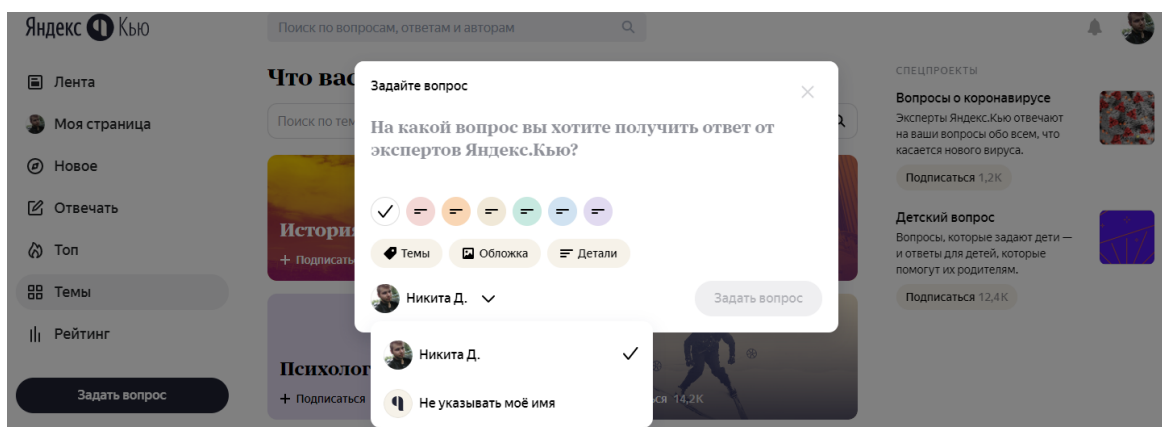


Рисунок 1.6 – Анонимный вопрос

- пользователей может задать вопрос конкретному человеку, что повышает вероятность компетентного ответа, что так же можно наблюдать на рисунке 1.5;
- высококвалифицированное комьюнити, заинтересованное в компетентном ответе, даже на весьма специфичные вопросы.

Приступим к рассмотрению минусов web-сайта Яндекс.Кью:

- интуитивно непонятный интерфейс: для того, чтобы обнаружить основные органы управления необходимо некоторое время на взаимодействие с сайтом;

- лишняя информация в виде баннеров и навязчивой рекламы, что можно наблюдать на рисунке 1.4 и рисунке 1.6;

- к вопросам и ответам можно прикрепить только изображения.

Далее рассмотрим web-сайт ask.ee.ru с его достоинствами:

- на сайте представлены неконкретные категории, можно видеть на рисунке 1.7;

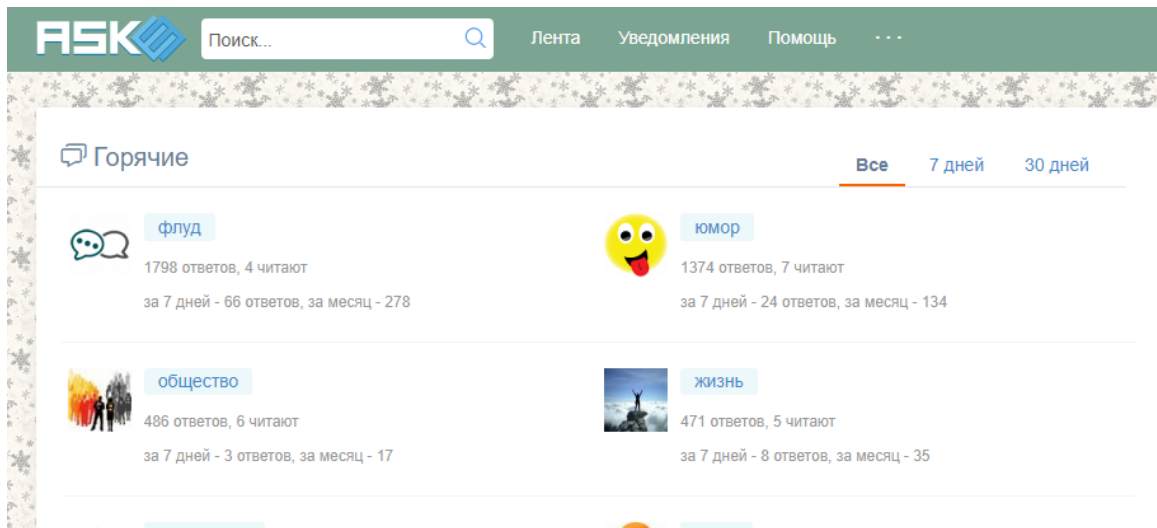


Рисунок 1.7 – Категории

- имеется возможность добавления медиа контента к вопросам и ответам;
- пользователь может персонализировать визуальную часть сайта, рисунок 1.8;

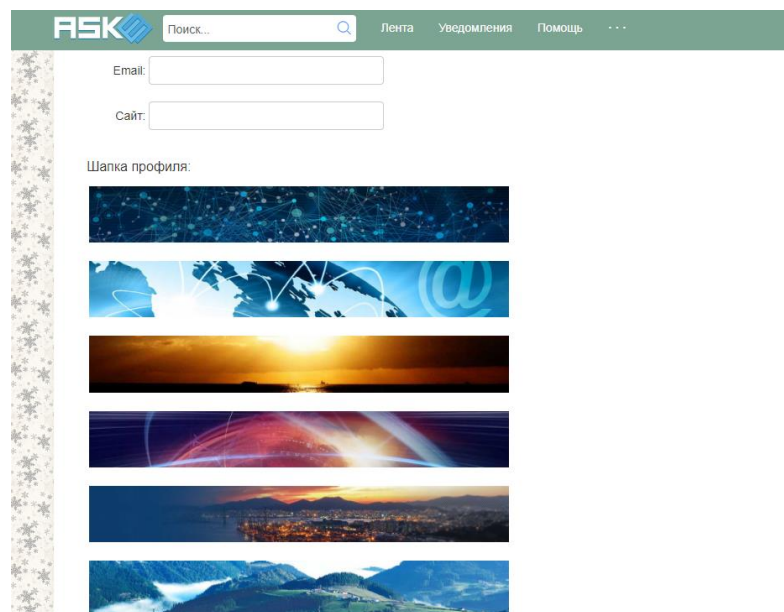


Рисунок 1.8 – Визуализация сайта

- отсутствие рекламных баннеров;
- реализована возможность общения между пользователями посредством личных сообщений.

Перейдем к недостаткам данного сайта:

- существует множество недоработок в интерфейсе сайта;
- много лишней личной информации о пользователе.

Перейдем к рассмотрению Web-приложения QandA-Online. Плюсами программы являются:

- простота использования (сервис не перегружен ненужными функциями);

- удобство регистрации;
- возможность задать неограниченное количество вопросов;
- навигация по сервису интуитивно понятна;
- реализована подписка на пользователя с push-уведомлениями.

Выявлены следующие минусы:

- отсутствие возможности писать личные сообщения пользователям;
- не возможности подписаться на вопрос или же категорию.

По результатам анализа аналогов были выявлены их достоинства и недостатки, которые позволили сформировать функциональные требования к разрабатываемому программному средству, представлены в разделе 1.3.

1.3 Постановка задачи проектирования

В рамках преддипломной практики осуществлялась разработка приложения для получения ответов на заданные вопросы.

У каждого периодически возникают вопросы, ответы на которые не удастся найти в Интернете. Т.к. приложение QandA-Online в перспективе имеет обширную аудиторию различной возрастной, интеллектуальной и социальной категорий, то пользователь имеет реальную возможность получить корректный ответ на интересующий вопрос.

Для того, чтобы грамотно реализовать приложение, необходимо изучить в каком формате пользователям удобнее размещать вопросы и получать ответы.

Функциональное назначение разрабатываемого программного средства состоит в получении ответов на разноплановые вопросы. Основным функциям программы являются:

- регистрация пользователя;
- авторизация пользователя;
- просмотр и редактирование данных профиля;
- добавление вопроса или ответов;
- удаление собственных вопросов и ответов;
- просмотр вопросов и ответов на них;
- оценка ответа;
- возможность подписки на пользователя для получения push-уведомлений о новых записях, подписках и ответах.

Для того, чтобы разрабатываемая система была удобна в использовании и конкурентоспособна, необходимо выделить требования, которые будут дополнительными ориентирами при написании:

- удобство использования – пользователь должен производить как можно меньше действий. Потребность в различных настройках должна быть сведена к минимуму;

- переносимость – легкость в переносе приложения или данных с одного компьютера на другой без потерь функциональности;
- устойчивость к нагрузкам – при выполнении обработки большого объема данных приложение должно рационально использовать ресурсы системы, не вызывая проблем с использованием данных ресурсов в других работающих приложениях;
- стабильность работы – приложение должно обеспечивать возможность корректного завершения в случае критической ошибки с извещением пользователя о причине сбоя;

1.4 Анализ требований к проекту

При разработке приложения необходимо решить следующие задачи:

- разработка и построение базы данных для хранения информации необходимой при формировании заданий;
- определение логико-функциональной структуры приложения;
- проведение анализа и оптимизации основных программных модулей, разработка структуры и интерфейса их взаимодействия;
- реализация приложения и его оптимизация в виде единого прикладного программного комплекса, определение необходимых аппаратных требований и условий функционирования программного продукта;
- проведение практического применения программного продукта с целью проверки его работоспособности.

Интерфейс необходим для удобной работы пользователя, в связи с этим он должен быть интуитивно понятным, с приятным визуальным оформлением и при этом выполнять все функции, описанные в пункте 1.3. Необходима плотная связь между входными и выходными данными. При совершении некоторых действий пользователь должен быть проинформирован об успешном или неуспешном окончании совершения операции.

В приложении, предназначенном для взаимодействия пользователей с целью получения ответов на вопросы, необходимо предоставить пользователю все права и возможности управления программой. Исходя из всего вышеизложенного, можно выделить возможности пользователя:

- регистрация и авторизация;
- создание нового вопроса;
- ответ на вопрос;
- оценки ответа;
- просмотр вопросов и ответов;
- подписка на пользователей;
- изменение профиля.

А также не исключение что пользователи будут не здравомыслящи, поэтому предусмотрена и роль администратора, имеющего следующие возможности:

- редактирование профиля любого пользователя;
- удаление вопросов и ответов пользователей;
- удаление профиля пользователя.

1.5 Выбор и обоснование средств и методов решения задач

Выбор системы управления базами данных (СУБД) представляет собой сложную многопараметрическую задачу и является одним из важных этапов при разработке клиент-серверных приложений. Определяющим здесь являются вид программного продукта и категории пользователей (или профессиональные программисты, или конечные пользователи, или и то, и другое).

Другими показателями, влияющими на выбор СУБД являются:

- удобство и простота использования;
- качество средств разработки, защиты и контроля базы данных;
- уровень коммуникационных средств в случае применения ее в сетях;
- фирма-разработчик;
- стоимость.

SQL Server 2018 обеспечивает встроенные в базы данных возможности обработки данных в оперативной памяти при любых рабочих нагрузках, более быстрое получение результатов анализа данных с использованием знакомых средств аналитики, а также использование решений по обработке больших данных на корпоративном уровне.

Microsoft SQL Server – система управления реляционными базами данных, разработанная корпорацией Microsoft. Основной используемый язык запросов – Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для небольших и средних по размеру баз данных, и в последние 5 лет – для крупных баз данных масштаба предприятия, конкурирует с другими СУБД в этом сегменте рынка [1].

Для реализации приложения была выбрана среда разработки Microsoft Visual Studio 2019 – язык программирования C#. Был выбран именно этот язык по причине следующих его достоинств:

- претендует на подлинную объектную ориентированность (а всякая языковая сущность претендует на то, чтобы быть объектом);
- призван практически реализовать компонентно-ориентированный подход к программированию, который способствует меньшей машинно-архитектурной зависимости результирующего программного кода, большей гибкости, переносимости и легкости повторного использования (фрагментов) программ;

- изначальная ориентация на безопасность кода (что особенно заметно в сравнении с языками С и С++);
- расширенная поддержка событийно-ориентированного программирования;
- является «родным» для создания приложений в среде Microsoft .NET [2].

Для реализации приложения была выбрана клиент-серверная архитектура. Основными преимуществами работы в клиент-серверном варианте являются:

- высокая производительность при обработке больших объемов данных информационной базы;
- удобство администрирования и упорядочивание доступа пользователей к информационной базе [3].

Учитывая выше сказанное, MS SQL Server 2018 и С# (с использованием среды разработки Microsoft Visual Studio 2019) подходят для реализации Web-приложения.

1.6 Разработка технического задания

На основании анализа сущности проблемы и исходных данных были сформулированы следующие требования к функциональности разрабатываемого приложения:

- возможность задать вопрос;
- оставить ответ на чей-либо вопрос;
- просмотр профиля;
- подписка на пользователей;
- оценка ответа;
- удаление своих вопросов и ответов;
- просмотр вопросов.

Расширенное техническое задание на разрабатываемый программный продукт приведено в приложении А.

2 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

2.1 Функциональная структура приложения

Приложение предназначено для получения ответа на вопрос, и должно предоставлять возможность списка вопросов и списка ответов отдельных пользователей, подписке на пользователей, возможность авторизации.

Помимо предоставления информации приложение должно обладать удобным и интуитивно понятным интерфейсом, оптимальными механизмами поиска информации.

Исходя из приведенных характеристик и требований технического задания (приложение А), была разработана функциональная структура данного обучающего приложения, которая представлена на рисунке 2.1.

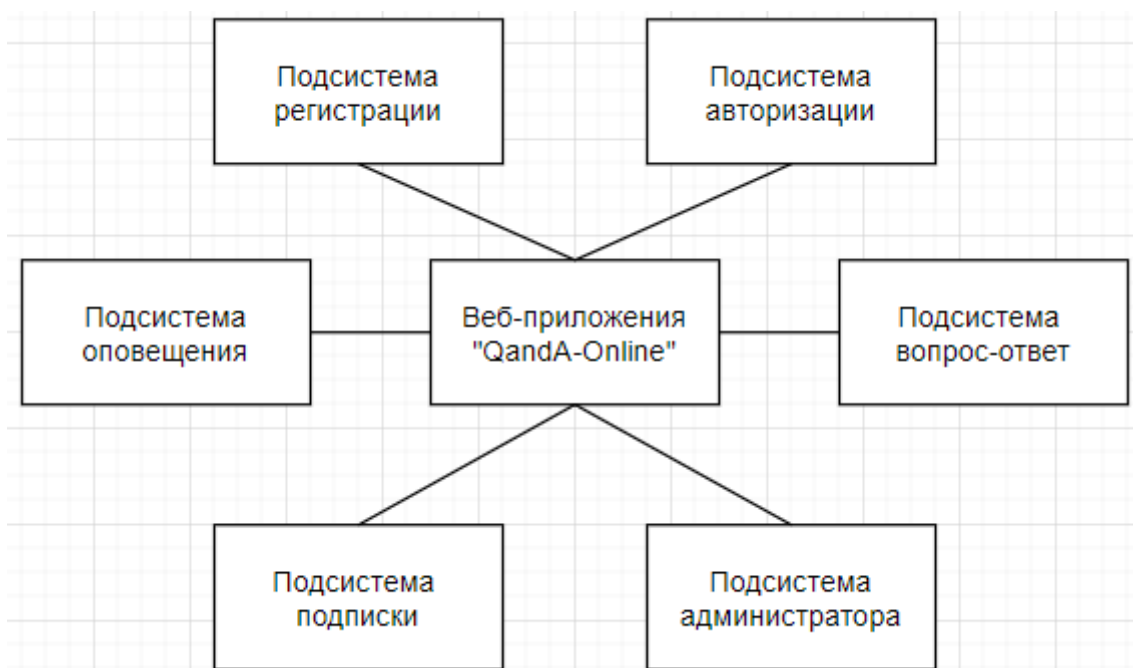


Рисунок 2.1 – Подсистемы программы

Приложение состоит из следующих подсистем:

- подсистема регистрации, позволяющая регистрироваться новым пользователям, вводить данные и отправлять их на сервер для дальнейшей обработки;
- подсистема авторизации, которая, при корректном вводе требующихся для авторизации данных, предоставляет права и доступ к личному кабинету;
- подсистема вопрос-ответ позволяющая задать вопрос и ответить на него;
- подсистема подписки на пользователей с дальнейшим уведомлением о новых ответах данного пользователя;

- подсистема оповещения, предоставляющая уведомления о произошедших событиях (подписка, ответ на вопрос, оценка ответа и новые вопросы у пользователей);
- подсистема администратора, для модерирования пользовательского контента.

2.2 Проектирование диаграммы вариантов использования

Визуальное моделирование в UML можно представить, как некоторый процесс поуровневого спуска от наиболее общей и абстрактной концептуальной модели исходной системы к логической, а затем и к физической модели соответствующей программной системы. Для достижения этих целей вначале строится модель диаграммы вариантов использования (use case diagram), которая описывает функциональное назначение системы или, другими словами, то, что система будет делать в процессе своего функционирования [4].

Диаграммы вариантов использования предназначены для упрощения взаимодействия с будущими пользователями системы, с клиентами, и особенно пригодятся для определения необходимых характеристик системы. Она представляет собой диаграмму, на которой изображаются варианты использования проектируемой системы, как правило, заключенные в границу субъекта и внешние актеры, а также определенные взаимоотношения между актерами и вариантами использования. Основными элементами (предметами языка UML), отображаемыми на диаграмме вариантов использования, являются: вариант использования, актер, системная граница, примечание [5].

Для данного Web-приложения можно выделить одного актера – это пользователь. Он наделен всеми правами и возможностями, то есть может просматривать списки отвеченных и заданных вопросов, отвечать на вопросы, задавать вопросы, а также просматривать профили других пользователей. Диаграмма вариантов использования представлена в приложении Б.

2.3 Информационное обеспечение проекта

В состав информационного обеспечения проекта на различных этапах разработки разработано «Web-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей» входят семейство стандартов UML, Microsoft SQL Server и C#. Методология стандартов IDEF на этапе проектирования применялась для моделирования информационной системы при помощи CASE-средства ERwin Data Modeler r7.2. СУБД Microsoft SQL 2018 Server

и язык программирования C# в среде разработки Microsoft Visual Studio 2019 применяются для создания приложения.

2.4 Разработка структуры конфигурации

Главной целью проектирования базы данных является выбор оптимального количества таблиц для хранения данных, полей, которые должны войти в ту или иную таблицу, а также планирование отношений между таблицами. Эту и многие другие проблемы при проектировании баз данных можно решить с помощью нормализации. При проектировании баз данных могут применяться различные подходы. Основная задача проектирования баз данных – это сокращение избыточности хранимых данных, а, следовательно, экономия используемых ресурсов (оперативной и дисковой памяти), уменьшение затрат на обновление избыточности копий данных, а также снижение вероятности нарушения целостности данных.

Реляционная база данных – это совокупность отношений, в которых хранится вся информация баз данных. Для пользователя такая база данных представляется набором двумерных таблиц, что облегчает понимание структуры данных и управления ими. Таблицы реляционной базы данных связаны между собой отношениями. Нормализация – это процесс приведения структур данных в состояние, обеспечивающее лучшие условия выборки, включения, изменения и удаления данных.

Схема базы данных разработано «Web-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей» представлена на рисунке 2.2.

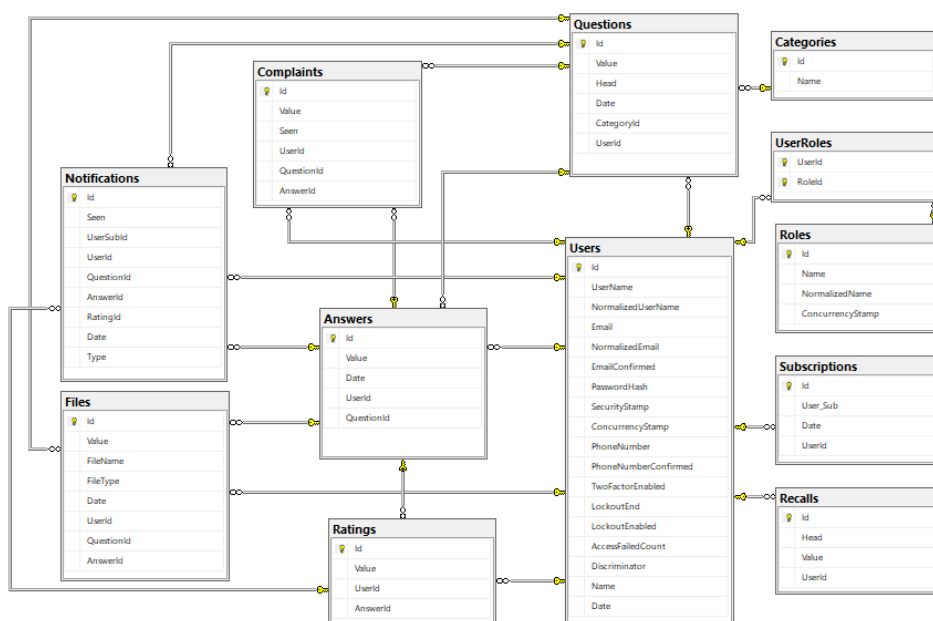


Рисунок 2.2 – Схема базы данных

Для хранения данных в приложении необходимы таблицы базы данных, перечисление которых представлен в таблице 2.1.

Таблица 2.1 – Перечень таблиц базы данных

Название таблицы	Описание
Users	Хранит основную информацию о пользователе
UserRoles	Промежуточная таблица, реализующая связь «многие ко многим»
Roles	Хранит роли пользователей
subscriptions	Хранит информацию о подписке пользователей
files	Хранит файлы аватарки пользователей, а также прикрепленных к вопросам и ответам
questions	Хранит информацию о вопросе
answers	Хранит информацию о ответе
ratings	Хранит информацию о рейтинге ответа
categories	Хранит категории вопросов
Recalls	Хранит отзывы пользователей о сайте
Complaints	Хранит жалобы пользователей
Notifications	Хранил оповещения

Рассмотрим подробнее структуру таблиц, используемых в приложении.

Основная информация о пользователях хранится в таблице Users. Наименование идентификаторов и их описание представлены в таблице 2.2.

Таблица 2.2 – Структура таблицы Users

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор пользователя	Nvarchar(450)
UserName	Логин пользователя	Nvarchar(256)
NormalizedUserName	Нормализованный логин	Nvarchar(256)
Email	Электронная почта	Nvarchar(256)
NormalizedEmail	Нормализованная электронная почта	Nvarchar(256)
EmailConfirmed	Подтверждение почты	bit
PasswordHash	Хэш пароля	Nvarchar(max)
PhoneNumber	Номер телефона	Nvarchar(max)
PhoneNumberConfirmed	Подтверждение номера телефона	bit
TwoFactorEnabled	Двухфакторная защита	bit

Идентификатор	Комментарий	Тип значения
AccessFailedCount	Количество неверных попыток входа	int
Name	Имя пользователя	Nvarchar(150)

Информация о ролях пользователей хранится в таблице Roles. Наименование идентификаторов и их описание представлены в таблице 2.3.

Таблица 2.3 – Структура таблицы Roles

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор роли	Nvarchar(450)
Name	Имя роли	Nvarchar(256)
NormalizedName	Нормализованное имя роли	Nvarchar(256)

Информация о ролях пользователей хранится в таблице UserRoles. Наименование идентификаторов и их описание представлены в таблице 2.4.

Таблица 2.4 – Структура таблицы UserRoles

Идентификатор	Комментарий	Тип значения
UserId	Уникальный идентификатор токена	Nvarchar(450)
RoleId	Уникальный идентификатор роли	Nvarchar(450)

Информация о подписках пользователей хранится в таблице subscriptions. Наименование идентификаторов и их описание представлены в таблице 2.5.

Таблица 2.5 – Структура таблицы subscriptions

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор подписки	Nvarchar(450)
User_Sub	Идентификатор пользователя, на которого подписан	Nvarchar(450)
UserId	Идентификатор пользователя, который подписан	Nvarchar(450)

Информация о файлах пользователей, а также прикрепленных к вопросам и ответам хранится в таблице files. Наименование идентификаторов и их описание представлены в таблице 2.6.

Таблица 2.6 – Структура таблицы files

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор файла	Nvarchar(450)
Value	Биты хранимого файла	Varbinary(max)
FileName	Имя файла	Nvarchar(max)
FileType	Тип файла	Nvarchar(max)

Идентификатор	Комментарий	Тип значения
Date	Дата создания записи	Datetime2(7)
UserId	Идентификатор пользователя	Nvarchar(450)
QuestionId	Идентификатор вопроса	Nvarchar(450)
AnswerId	Идентификатор ответа	Nvarchar(450)

Информация о вопросах хранится в таблице questions. Наименование идентификаторов и их описание представлены в таблице 2.7.

Таблица 2.7 – Структура таблицы questions

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор вопроса	Nvarchar(450)
Value	Вторичный текст вопроса	Nvarchar(max)
Head	Основной текст вопроса	Nvarchar(max)
Date	Дата созданий записи	Datetime2(7)
CategoryId	Идентификатор категории	Nvarchar(450)
UserId	Идентификатор пользователя	Nvarchar(450)

Информация о ответах хранится в таблице answers. Наименование идентификаторов и их описание представлены в таблице 2.8.

Таблица 2.8 – Структура таблицы answers

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор ответа	Nvarchar(450)
Value	Текст ответа	Varbinary(max)
Date	Дата создания записи	Datetime2(7)
UserId	Идентификатор пользователя	Nvarchar(450)
QuestionId	Идентификатор вопроса	Nvarchar(450)

Информация о оценках хранится в таблице ratings. Наименование идентификаторов и их описание представлены в таблице 2.9.

Таблица 2.9 – Структура таблицы ratings

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор оценки	Nvarchar(450)
Value	Оценка ответа	int
UserId	Идентификатор пользователя	Nvarchar(450)
AnswerId	Идентификатор ответа	Nvarchar(450)

Информация о категориях хранится в таблице categories. Наименование идентификаторов и их описание представлены в таблице 2.10.

Таблица 2.10 – Структура таблицы categories

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор ответа	Nvarchar(450)
Name	Название категории	Nvarchar(max)

Информация о отзывах, комментариях и предложениях для сайта хранятся в таблице Recalls. Наименование идентификаторов и их описание представлены в таблице 2.11.

Таблица 2.11 – Структура таблицы Recalls

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор отзыва	Nvarchar(450)
Head	Основной текст отзыва	Nvarchar(max)
Value	Вторичный текст отзыва	Nvarchar(max)
UserId	Уникальный идентификатор пользователя	Nvarchar(450)

Информация о жалобах хранится в таблице Complaints. Наименование идентификаторов и их описание представлены в таблице 2.12.

Таблица 2.12 – Структура таблицы Complaints

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор жалобы	Nvarchar(450)
Value	Текст жалобы	Nvarchar(max)
Seen	Отметка о просмотре жалобы	bit
UserId	Уникальный идентификатор пользователя	Nvarchar(450)
QuestionId	Уникальный идентификатор вопроса	Nvarchar(450)
AnswerId	Уникальный идентификатор ответа	Nvarchar(450)

Информация о оповещениях хранится в таблице Notifications. Наименование идентификаторов и их описание представлены в таблице 2.13.

Таблица 2.13 – Структура таблицы Notifications

Идентификатор	Комментарий	Тип значения
Id	Уникальный идентификатор оповещения	Nvarchar(450)
Seen	Отметка о просмотре оповещения	bit
UserSubId	Уникальный идентификатор пользователя, на которого подписались	Nvarchar(450)
UserId	Уникальный идентификатор пользователя	Nvarchar(450)
QuestionId	Уникальный идентификатор вопроса	Nvarchar(450)
AnswerId	Уникальный идентификатор ответа	Nvarchar(450)

Идентификатор	Комментарий	Тип значения
RatingId	Уникальный идентификатор оценки	Nvarchar(450)
Date	Дата создания записи	Datetime2(7)
Type	Тип оповещения	Nvarchar(max)

2.4 Разработка концепции пользовательского интерфейса

В создании Web-приложение большую роль играет правильный пользовательский интерфейс, который должен быть удобен, понятен и лаконичен.

Так как на данный Web-приложение пользователи могут заходить не только с компьютеров, но и с мобильных телефонов, должна быть учтена и предотвращена возможность «съезжания» и поломки элементов сайта.

С анализа аналогов данной тематики были вынесены следующие пункты: главная страница, страница регистрации, страница авторизации, страница просмотра профиля пользователя, страница просмотра вопроса, страница категорий.

Главная страница будет содержать информацию о последних отвеченных вопросах, последних заданных вопросов и рейтингом пользователей.

На странице с просмотром профиля пользователю будет находиться кнопка для обновления фото в профиле, для обычных пользователей данная кнопка не будет доступна, также будут отображены подписки и вопросы пользователя.

2.5 Безопасность и защита данных

Обеспечение безопасности данных строится в основном на принципе, что целостность данных подразумевает недопустимость модификации информации пользователями или ее порчи вследствие сбоев или вирусных атак. Поэтому у пользователя не будет доступа к базе данных и к исходному коду программы.

3 РЕАЛИЗАЦИЯ И ТЕСТИРОВАНИЕ

3.1 Реализация функциональных частей программного обеспечения

Web API представляет иной способ построения приложения ASP.NET несколько отличный от ASP.NET MVC. Web API представляет собой веб-службу, которая может взаимодействовать с различными приложениями. При этом приложение может быть веб-приложением ASP.NET, либо может быть мобильным или обычным десктопным приложением.

Также надо отметить, что платформа Web API 2 не является частью фреймворка ASP.NET MVC и может быть задействована как в связке с MVC, так и в соединении с Web Forms. Поэтому в Web API имеется своя система версий. Так, первая версия появилась с .net 4.5. А вместе с .NET 4.5.1 и MVC 5 вышла Web API 2.0 [8].

Angular представляет фреймворк от компании Google для создания клиентских приложений. Прежде всего он нацелен на разработку SPA-решений (Single Page Application), то есть одностраничных приложений. В этом плане Angular является наследником другого фреймворка AngularJS. В то же время Angular это не новая версия AngularJS, а принципиально новый фреймворк.

Angular предоставляет такую функциональность, как двустороннее связывание, позволяющее динамически изменять данные в одном месте интерфейса при изменении данных модели в другом, шаблоны, маршрутизация и так далее.

Одной из ключевых особенностей Angular является то, что он использует в качестве языка программирования TypeScript. Но Angular не ограничен языком TypeScript. При желании можем писать приложения на Angular с помощью таких языков как Dart или JavaScript. Однако TypeScript все таки является основным языком для Angular [9].

Пример класса контроллера авторизации и регистрации пользователя на стороне backend представлена в листинге 3.1.

Листинг 3.1 – Регистрация и авторизация пользователя

```
[Route("[controller]")]
[ApiController]
public class AppUserController : ControllerBase
{
    private UserManager<User> _userManager;
    private readonly ApplicationSettings _appSettings;
    public AppUserController(UserManager<User> userManager,
        IOptions<ApplicationSettings> appSettings)
    {
        _userManager = userManager;
        _appSettings = appSettings.Value;
    }
    [HttpPost]
    [Route("Register")]
}
```

```

public async Task<object> RegisterUser(AppUserModel model)
{
    User appuser = new User()
    {
        UserName = model.UserName,
        Email = model.Email,
        Name = model.Name,
    };
    try
    {
        var result = await _userManager.CreateAsync(appuser,
model.Password);
        return Ok(result);
    }
    catch (Exception ex)
    {
        throw ex;
    }
}
[HttpPost]
[Route("Login")]
public async Task<IActionResult> LoginUser(LoginModel model)
{
    var user = await _userManager.FindByNameAsync (model.UserName);
    if (user != null && await _userManager.CheckPasswordAsync (user,
model.Password))
    {
        var tokenDescriptor = new SecurityTokenDescriptor
        {
            Subject = new ClaimsIdentity(new Claim[]
            {
                new Claim("UserID",user.Id.ToString())
            }),
            Expires = DateTime.UtcNow.AddDays(1),
            SigningCredentials = new SigningCredentials(new
SymmetricSecurityKey(Encoding.UTF8.GetBytes(_appSettings.
JWT_Secret)), SecurityAlgorithms.HmacSha256Signature)
        };
        var tokenHandler = new JwtSecurityTokenHandler();
        var securityToken = tokenHandler.CreateToken(tokenDescriptor);
        var token = tokenHandler.WriteToken(securityToken);
        return Ok(new { token });
    }
    else
        return BadRequest(new { message = "Username or password is
incorrect." });
}
}

```

Пример класса компонента авторизации пользователя на стороне frontend представлена в листинге 3.2.

Листинг 3.2 – Авторизация пользователя

```
export class LoginComponent implements OnInit {
  formModel = {
    UserName: '',
    Password: ''
  }
  constructor(private service: UserService, private router: Router, private toastr: ToastrService) { }
  ngOnInit() {
    if (localStorage.getItem('token') != null)
      this.router.navigateByUrl('/home');
  }
  onSubmit(form: NgForm) {
    this.service.login(form.value).subscribe(
      (res: any) => {
        localStorage.setItem('token', res.token);
        this.router.navigateByUrl('/home');
      },
      err => {
        if (err.status == 400)
          this.toastr.error('Incorrect username or password.', 'Authentication failed.');
```

В результате реализации программного средства было разработано «Web-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей», отвечающий всем требованиям технического задания. Представление результатов реализации приведено в документе «Описание программы», представленном в приложении Г.

3.2 Разработка программной документации

Вследствие создания программного продукта была разработана следующая документация:

– руководство оператора (приложение Е) – для помощи пользователям в ознакомлении и освоении программного средства;

					ДНВ.508190.009.ПЗ	Лист
						27
Изм.	Лист	№ докум.	Подпись	Дата		

– руководство администратора (приложение Ж) – для помощи модераторам в ознакомлении и освоении программного средства.

3.3 Тестирование

Тестирование – один из важнейших этапов контроля качества разрабатываемого ПО. Целью проведения испытаний является установление работоспособности разработанного программного обеспечения, соответствия техническому заданию и другим программным документам. При выявлении несоответствий в работе программы, ошибок в программе или программной документации требуется доработка программного обеспечения и (или) документации.

С целью обнаружения ошибок, проверки реализации функциональных и нефункциональных требований к программе, было проведено тестирование разработанного программного средства.

В качестве метода тестирования было выбрано функциональное тестирование.

Функциональное тестирование – это тестирование ПО в целях проверки реализуемости функциональных требований, то есть способности ПО в определенных условиях решать задачи, нужные пользователям. Функциональные требования определяют, что именно делает ПО, какие задачи оно решает.

Результаты тестирования представлены в таблице 3.1.

Таблица 3.1 – Результаты тестирования программного продукта

Тестовый вариант	Входные данные	Ожидаемый результат	Результат тестирования
Тестирование клиентского модуля			
Запуск клиентского модуля	Вход на сайт	Открытие главной страницы	Тест пройден успешно
Переход на страницу «UserList»	Нажатие на пункт меню «UserList»	Открытие страницы «UserList»	Тест пройден успешно
Переход к определенной категории	Нажатие на раздел меню «Category» и выбрать категорию	Открытие страницы выбранной категории	Тест пройден успешно
Переход в раздел «Sing In»	Нажатие на раздел меню «Sing In/Sing Up»	Открытие раздела с «Sing In»	Тест пройден успешно
Переход в раздел «Sing Up»	Нажатие на раздел меню «Sing In/Sing Up» и перейти в вкладку «Sing Up»	Открытие раздела «Регистрация»	Тест пройден успешно

Тестовый вариант	Входные данные	Ожидаемый результат	Результат тестирования
Переход на страницу с отзывами после авторизации	Нажатие на кнопку «Recall»	Открытие страницы с отзывами	Тест пройден успешно
Добавление нового комментария после авторизации	Нажатие на кнопку «Add Answer»	Новый комментарий успешно добавлен	Тест пройден успешно
Редактирование фотографии после авторизации	Нажатие на кнопку «Upload»	Редактирование прошло успешно	Тест пройден успешно
Поиск вопросов и пользователей после авторизации	Ввод данных в поле ввода	Поиск прошел успешно	Тест пройден успешно
Подписка и отписка от других пользователей после авторизации	Нажатие на кнопку «subscribe/unsubscribe»	Подписка/Отписка прошла успешно	Тест пройден успешно
Добавление лайков понравившемуся комментарию после авторизации	Нажатие на кнопку «Лайк»	Лайк успешно добавлен	Тест пройден успешно
Переход в личный кабинет пользователя после авторизации	Нажатие на кнопку «Profile»	Открытие личного кабинета пользователя	Тест пройден успешно
Переход на страницу добавления вопроса после авторизации	Нажатие на раздел меню «Add question»	Открытие раздела «Add question»	Тест пройден успешно
Тестирование модуля администратора			
Редактирование информации о пользователях	Изменение названия и фото пользователя; Нажатие кнопки «Сохранить»	Измененная информация о пользователе сохранена	Тест пройден успешно
Удаление пользователя	Нажатие на раздел меню «User List»; нажатие на кнопку «Delete» выбранного пользователя	Удаление выбранного пользователя	Тест пройден успешно

Продолжение таблицы 3.1

Тестовый вариант	Входные данные	Ожидаемый результат	Результат тестирования
Удаление вопроса	Нажатие на раздел «All questions»; Нажатие на кнопку «Delete» выбранного вопроса	Удаление бранного проса вы- во-	Тест пройден успешно
Удаление комментария	Нажатие на вопрос; нажатие на кнопку «Delete» выбранного комментария	Удаление бранного ментария вы- ком-	Тест пройден успешно

В результате тестирования система показала себя, как стабильный и готовый к работе программный продукт.

4 ЭКОНОМИЧЕСКАЯ ЧАСТЬ

4.1 Обоснование необходимости вывода продукта на рынок

Цель экономического раздела дипломного проекта – рассчитать затраты на разработку программного обеспечения и определить экономическую эффективность от его внедрения.

Проведённый в главе 1.2 сравнительный анализ web-приложений-аналогов показал, что они имеют ряд недостатков. Это позволит разрабатываемому программному комплексу найти своих потребителей.

Данный продукт имеет широкое применение среди различных социальных групп, так как не имеет специфических ограничений в вопросах, которые можно задать.

Web-приложение применимо для всевозможных слоев населения, которым нужно быстро и эффективно найти ответ на вопрос, в случае если ответ не может быть найден из уже имеющихся, то задать его. Еще одним способом применения является привлечение пользователей для ответа на заданные вопросы, заинтересовав их наличием разнообразных категорий, из которых каждый может выбрать свою область специализации.

Исходя из этого можно сделать вывод об актуальности разрабатываемого продукта на рынке ввиду того, что большое количество людей имеют не менее большое количество вопросов, на которые срочно требуются ответы.

4.2 Структура (этапы) работ по созданию программного обеспечения

Для разработки программного обеспечения необходимо время и структура этапов разработки, которое выражается в количестве недель, затраченных на его разработку. В таблице 4.1. представлена структура этапов и время для каждого этапа необходимого для разработки «Web-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей».

Таблица 4.1 – Таблица суммарного времени разработки программного обеспечения

Этапы разработки	Затраты времени, недели	Процент от общего времени, процент
Проектирование изделия	2 недели	17
Реализация	недель	58
Тестирование	1 неделя	8
Доработка	2 недели	17
Итог	12 недель	100

Процент от общего времени находим по следующей формуле:

$$T_p = \frac{W_i}{W_s} \times 100\%, \quad (4.1)$$

где T_p – процент от общего времени;
 W_i – число недель одного этапа;
 W_s – количество недель разработки.

$$\frac{2}{12} \times 100\% = 17\%, \quad (4.2)$$

$$\frac{7}{12} \times 100\% = 58\%, \quad (4.3)$$

$$\frac{1}{12} \times 100\% = 8\%, \quad (4.4)$$

$$\frac{2}{12} \times 100\% = 17\%. \quad (4.5)$$

Диаграмма выполнения работ по разработке программного обеспечения (рисунок 4.1).



Рисунок 4.1 - Диаграмма распределения времени работы при разработке программного обеспечения

Кроме деления жизненного цикла программного обеспечения на стадии и фазы, можно выделить восемь видов работ, которые выполняются в процессе создания программного продукта:

- анализ требований и планирование – предусматривает разработку спецификаций, анализ и модификацию функциональных, технических, интерфейсных требований;
- проектирование изделий – включает в себя определение, спецификацию, анализ и модификации аппаратно-программной архитектуры проекта программы и базы данных;
- программирование – детальное проектирование, кодирование, автономная отладка и комплексирование отдельных компонентов программы, а также планирование работ программистов, разработка базы данных, документирование отдельных компонентов и организация программирования на уровне компонентов;
- планирование отладки;
- верификация – процесс проверки корректности требований, отладки изделий и приемные испытания;
- управление проектом – планирование и контроль проекта, контроль и регулирование договоров, связь с пользователями;
- контроль качества – разработка и контроль, стандартные и технические проверки программных средств и процессов разработки;
- документирование – разработка и корректировка руководства для пользователя и операторов.

Деление жизненного цикла программных средств на стадии, фазы и этапы, а также более подробное описание работ, выполняемых в процессе их разработки, производства и эксплуатации, необходимо для детального определения затрат, требуемых на эти цели

4.3. Составление сметы затрат на разработку программного обеспечения

Стоимостная оценка ПО представлена составлением сметы затрат.

1. Расходы по статье «Материалы и комплектующие» (М) отражают расходы на магнитные носители, бумагу, красящие ленты и другие материалы, необходимые для разработки ПО.

Таблица 4.2 – Стоимость материалов, необходимых для разработки программного обеспечения

Наименование материалов	Ед. измерения	Количество	Цена за единицу, бел. р.	Сумма, бел. р.
USB Flash MIREX	Гб	1	10,90	10,90

Наименование материалов	Ед. измерения	Количество	Цена за единицу, бел. р.	Сумма, бел. р.
Бумага офисная Хероx Performer формата А4, 500л.	л	1	9,60	9,60
Всего:	20,50			

Общая стоимость основных и вспомогательных материалов М=20,50 бел. р.

2. Статья «Электроэнергия». Стоимость электроэнергии необходимой для разработки программного обеспечения представлена в таблице 4.4. Для расчета данного показателя необходим иметь в наличии следующие данные:

- Продолжительность недель разработки.
- Количество часов потребления электроэнергии за весь период разработки.
- Стоимость одного кВт/ч электроэнергии (тариф).
- Значение курса белорусского рубля по отношению к доллару США, установленного Национальным банком РБ, на дату установления тарифа на электрическую энергию.

Таблица 4.4 – Стоимость электроэнергии необходимой для разработки программного обеспечения

Статья расходов	Длительность периода, недели	Количество кВт/ч	тариф за 1 кВт/ч.	Расход (бел.руб.)
Электроэнергия	12	480	0,376	180,48

Затраты электроэнергии формируются из затрат электроэнергии за один рабочий день, количество дней в неделю, количество недель разработки. При 8-часовом рабочем дне и 5-дневной неделе получаем в неделю 40 часов, за 12 недель разработки 480 часов.

$$Hw = h \times wd \times ws, \quad (4.6)$$

где Hw – количество рабочего времени, затраченного на разработку;
 h – продолжительность рабочего дня (часы);
 Wd – количество рабочих дней;
 Ws – сумма недель разработки.

$$8 \times 5 \times 12 = 480 \quad (4.7)$$

Базовый тариф для прочих потребителей с 01.01.2020г. при соотношении курса белорусского рубля к доллару США (1\$ = 2.1085 руб. РБ) составляет 0,33048 руб. за 1 кВт/ч (согласно Приказа Министерства антимонопольного регулирования и торговли Республики Беларусь от 18.01.2018 №11).

При изменении курса доллара США тариф индексируется по формуле:

$$T_n = T_6 * (0,24 + 0,76 * \frac{K_n}{K_6}) \quad (4.8)$$

где T_n – тариф на электрическую энергию, подлежащий применению на день оформления платежных документов и день оплаты потребителем за потребленную электрическую энергию;

T_6 – тариф на электрическую энергию, установленный декларацией;

K_n – значение курса белорусского рубля по отношению к доллару США, установленного Национальным банком РБ, на день оформления платежных документов и день оплаты за потребленную электрическую энергию (на 25.05.2020 курс белорусского рубля к доллару США 2,418:1);

K_6 – значение курса белорусского рубля по отношению к доллару США, установленного Национальным банком РБ, на дату установления тарифа на электрическую энергию.

На 05.05.2020 курс белорусского рубля к доллару США 2,418:1 установлен Национальным банком РБ. На основе данного курса рассчитаем тариф на электрическую энергию.

$$T_n = 0,33048 * (0,24 + 0,76 * \frac{2,418}{2,1085}) = 0,376 \quad (4.9)$$

3. Статья «Основная заработная плата». Основной статьей расходов на создание ПО является заработная плата разработчиков (исполнителей) проекта, в число которых принято включать.

					ДНВ.508190.009.ПЗ	Лист
						35
Изм.	Лист	№ докум.	Подпись	Дата		

Таблица 4.5 – Стоимость электроэнергии необходимой для разработки программного обеспечения

Специалисты	Количество (чел.)	Количество месяце в разработки	Основная зарплата (с учетом премии) за мес. (бел. руб.)	Основная с учетом дополнительной за месяц (бел. руб.)	Общая сумма заработной платы (бел. руб.)	Общая + дополнительная (бел. руб.)
Инженеры-программисты (пишут код)	1	4	551,85	634,62	2207,42	2538,53
Итого:	2538,53					

С 1 января 2020 года в Беларуси появится новая система начисления заработной платы работникам бюджетной сферы. Правовое основание для этого – Указ Президента от 18 января 2019 г. № 27 «Об оплате труда работников бюджетных организаций». Для оплаты труда бюджетников применяется базовая ставка – она пришла на «смену» тарифной ставке первого разряда. Она составляет 185 бел. руб.

Рассчитаем зарплату месячную тарифную ставку (оклад). Для программистов, работающих в бюджетной организации представлены коэффициенты в таблице 4.6.

Таблица 4.6 – Тарифные ставки.

Категории работников	Разряд	Коэффициент
1.4. специалисты, занимающие должности, квалификационными требованиями которых предусматривается наличие высшего, среднего специального образования	4 - 8	1,21 – 1,57

Месячная тарифная ставка каждого исполнителя (T_m) определяется путем умножения действующей базовой ставки (T_{m1}) на тарифный коэффициент (T_k), соответствующий установленному тарифному разряду:

$$T_m = T_{m1} * T_k, \quad (4.10)$$

$$185 * 1,57 = 290,45 \quad (4.11)$$

Основная заработная плата исполнителей на конкретное ПО рассчитывается по формуле:

$$З_0 = T_m * K * n * t \quad (4.12)$$

где n – количество исполнителей, занятых разработкой конкретного ПО;
 T_m – месячная тарифная ставка (бел. руб.);
 K – коэффициент премирования;
 t – период времени, затраченный на разработку ПО.

Рассчитываем основную заработную плату исполнителя на конкретное ПО:

$$290,45 * 1,9 * 1 * 4 = 2207,42 \text{ бел. руб.} \quad (4.13)$$

4. Статья «Дополнительная заработная плата» на конкретное ПО ($З_d$) включает выплаты, предусмотренные законодательством о труде (оплата отпусков, льготных часов, времени выполнения государственных обязанностей и других выплат, не связанных с основной деятельностью исполнителей), и определяется по нормативу в процентах к основной заработной плате:

$$З_{di} = З_{io} * H_d / 100\% \quad (4.14)$$

где $З_{di}$ – дополнительная заработная плата исполнителей на конкретное ПО (руб.);

H_d – норматив дополнительной заработной платы (принимается в размере 7 - 15 %).

$$2207,42 * 15 / 100 = 331,11, \quad (4.15)$$

$$331,11 / 4 = 82,77 \text{ (доп. зарплата за 1 месяц).} \quad (4.16)$$

5. Статья «Отчисления на социальные нужды». К данным затратам относят отчисления в фонд социальной защиты населения ($H_{сз}$ – 34 %) и отчисления на обязательное страхование от несчастных случаев ($H_{стр}$ – 0,1 – 0,6 %).

Отчисления на социальные нужды определяются в соответствии с действующими законодательными актами по нормативу в процентном отношении к фонду основной и дополнительной зарплаты исполнителей, определенной по нормативу, установленному в целом по организации:

$$З_{си} = (З_{di} + З_{io}) * (H_{сз} + H_{стр}) / 100 \quad (4.17)$$

где $H_{сз}$ – норматив отчислений в фонд социальной защиты населения (%);

$H_{стр}$ – норматив отчислений на обязательное страхование от несчастных случаев (%).

$$(331,11+2207,42) * (34+0,6) / 100 = 878,331 \quad (4.18)$$

6. Статья «Амортизация основных средств и нематериальных активов». Определим стоимость основных средств (ноутбук) и нематериальных активов (программное обеспечение), на которые распространяются амортизационные отчисления (таблица 4.7).

Таблица 4.7 – Данные по амортизационной стоимости.

Активы	Стоимость актива, бел. руб.	Период Амортизации (лет)	%	Коэфф. ускорения	за 1 год, бел. р.	Сумма за 1 мес. бел. р.	за период разработки (мес.) бел. р.
Ноутбук	1800	4	25	-	450	37,50	150,00
Программное обеспечение	1138,09	8	12,5	2	284,52	23,71	94,84
Итого:	244,84						

Определим установленный размер амортизационных отчислений на полное восстановление, выраженное в % по формуле:

$$H_a = \frac{1}{T_n} * 100\%, \quad (4.19)$$

где H_A – годовая норма амортизации;
 T_n – нормативный срок службы.

$$1 / 8 * 100 = 12,5\% \quad (4.20)$$

К нормам амортизации могут применяться поправочные коэффициенты, которые учитывают степень загрузки оборудования.

При нелинейном способе годовая сумма амортизационных отчислений рассчитывается методом суммы чисел лет либо методом уменьшаемого остатка с коэффициентом ускорения от 1 до 2,5 раз. Нормы начисления амортизации в первом и каждом из последующих лет срока применения нелинейного способа могут быть различными.

Проиллюстрируем расчет амортизации на примере. Первоначальная стоимость объекта 1138,09 бел. рублей. Срок полезного использования – 8 лет.

Годовая норма амортизации – 12,5 %. Коэффициент ускорения составит 2, а при маленьком периоде амортизации учитывать не будем. Тогда норма амортизации составит:

$$H_A = \frac{1}{T_n} * k_{\text{уск}} * 100\% = \frac{1}{8} * 2 * 100\% = 25\% \quad (4.21)$$

Сумма начисленной амортизации за 1-й год составит:

$$A_i = \frac{S \cdot H_A}{100}, \quad (4.22)$$

где A_i – годовая норма амортизации;
 H_A – годовая норма амортизации;
 S – остаточная или первоначальная стоимость актива.

$$1138,09 \cdot 25 / 100 = 284,52 \quad (4.23)$$

За период разработки:

$$284,52 / 12 \cdot 4 = 94,84 \quad (4.24)$$

7. Статья «Расходы на спецоборудование» (P_{ci}) включает затраты средств на приобретение вспомогательных технических и программных средств специального назначения, необходимых для разработки конкретного ПО, включая расходы на их проектирование, изготовление, отладку, установку и эксплуатацию. Затраты по этой статье определяются в соответствии со сметой расходов, которая составляется перед разработкой ПО.

Данная статья включается в смету расходов на разработку ПО в том случае, когда приобретаются специальное оборудование или специальные программы, предназначенные для разработки и создания только данного ПО:

$$P_{ci} = \sum_{i=1}^n C_{ci}, \quad (4.25)$$

где C_{ci} – стоимость конкретного специального оборудования (бел. руб.);
 n – количество применяемого специального оборудования.

Расходы на спецоборудование не понадобились.

8. Статья «Прочие прямые расходы» (P_3) на конкретное программное обеспечение включает затраты: на оплату услуг связи, Интернета, транспортные расходы, канцтовары, приобретение и подготовку специальной научно-технической информации и специальной литературы. Расходы определяются студентом самостоятельно по действующим на момент написания работы ценам и тарифам.

Тарифный план высокоскоростного не лимитируемого доступа в сеть Интернет («Рекорд 200» (скорость прием/передача до 200/100 Мбит/с)), в месяц 43 руб.

$$43 \text{ руб.} \cdot 4 \text{ месяца} = 172 \text{ (бел. руб.)} \quad (4.26)$$

Общая сумма расходов по смете (плановая себестоимость) (С) на ПО рассчитывается по формуле:

$$C = M + \mathcal{E} + \mathcal{Z}_o + \mathcal{Z}_d + O_{cn} + A + P_c \cdot \Pi_3 \quad (4.27)$$

Результаты расчетов оформляются в виде таблицы.

Таблица 4.8 – Расчет плановой себестоимости программного обеспечения «Web-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей»

Статья затрат	Затраты, бел. руб.
1. Материалы и комплектующие (М);	20,50
2. Электроэнергия;	180,48
3. Основная заработная плата исполнителей (З _о)	2207,42
4. Дополнительная заработная плата исполнителей (З _д);	331,11
5. Отчисления на социальные нужды (О _{сн});	878,33
6. Амортизация (А);	244,84
7. Расходы на спецоборудование (Р _с);	----
8. Прочие прямые расходы (П ₃);	172
Общая сумма расходов по смете (плановая себестоимость С)	4034,68

Таблицу 4.8. удобно представить в виде диаграммы, отображающей влияние статей затрат программного обеспечения на окончательную себестоимость программы (рис. 4.2).

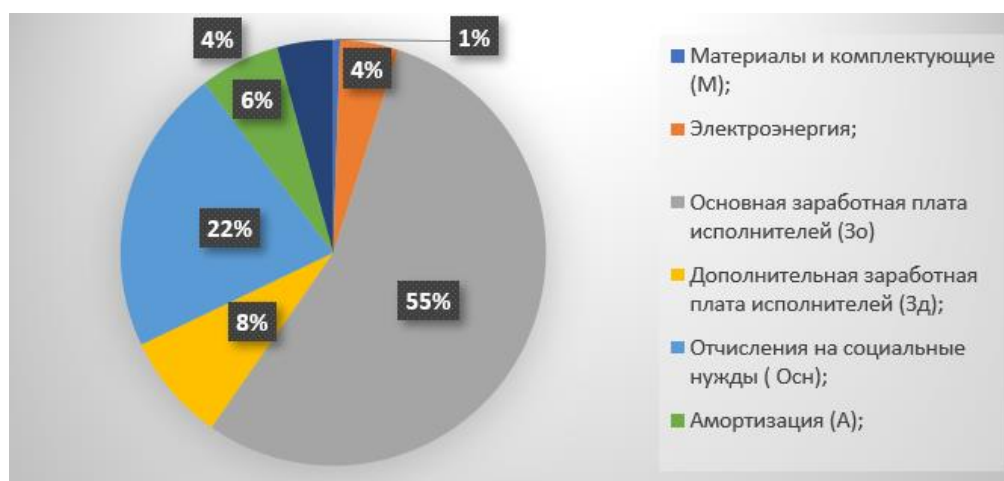


Рисунок 4.2 - Динамика затрат на разработку программного обеспечения

4.4 Расчет экономического эффекта разработчика и пользователя (заказчика) программного обеспечения

4.4.1 Экономический эффект у разработчика программного обеспечения.

Рассчитаем отпускную цену программного обеспечения.

С учетом действующих в республике нормативных документов отпускная цена на продукцию предприятия рассчитывается по формуле:

$$\text{ОЦ} = \text{С} + \text{П}, \quad (4.28)$$

где ОЦ – отпускная цена;
С – плановая себестоимость;
П – прибыль.

$$4034,68 + 794,33 = 4829,01 \text{ (бел. руб.)} \quad (4.29)$$

Прибыль закладывается в цену по нормативу рентабельности (устанавливается самостоятельно), расчет производится по следующей формуле:

$$\text{П} = \text{R} * \text{С}, \quad (4.30)$$

где R – норматив рентабельности (например, если рентабельность 20 %, то при расчете переводим в коэффициент и получаем 0,2).

С – плановая себестоимость, бел. руб.;

$$4034,68 * 0,2 = 806,94 \text{ (бел. руб.)} \quad (4.31)$$

Отпускная цена изготовителя с налогом на добавленную стоимость:
Ставка налога на добавленную стоимость установлена в размере 20 %.

$$\text{НДС} = (\text{С} + \text{П}) \cdot \text{ставка НДС (\%)} / 100 \%, \quad (4.32)$$

где С – плановая себестоимость, руб.;
П – прибыль, руб.;

Ставка НДС = 20 %;

$$(4829,01 + 806,94) * 20 / 100 = 1127,19 \text{ (бел. руб.)} \quad (4.33)$$

Прогнозируемая отпускная цена на программное обеспечение с НДС:

$$\text{ОЦ с НДС} = C + П + \text{НДС} \quad (4.34)$$

$$4829,01 + 806,94 + 1127,19 = 6763,14 \text{ (бел. руб.)} \quad (4.35)$$

Таким образом, разработчик программного обеспечения может продать заказчику программное обеспечение по рассчитанной цене, что покроет затраты и обеспечит прибыль за разработку проекта.

4.4.2 Экономический эффект от использования программного обеспечения у пользователя (заказчика).

При сравнении базового и нового варианта программного обеспечения в качестве экономического эффекта будет выступать общая сумма экономии всех видов ресурсов относительно базового варианта.

Приведем расчет экономии некоторых видов ресурсов в связи с использованием нового программного обеспечения.

Экономия затрат на заработную плату (\mathcal{E}_3) при использовании нового программного обеспечения в расчете на объем выполненных работ определяется по формуле:

$$\mathcal{E}_3 = \mathcal{E}_{3e} \times Q, \quad (4.36)$$

где \mathcal{E}_{3e} – экономия затрат на заработную плату при решении задач с использованием нового программного обеспечения в расчете на 1 задачу, руб.

Q – количество задач, решаемых за год с использованием нового программного обеспечения.

Экономия затрат на заработную плату в расчете на 1 задачу (\mathcal{E}_{3e}) рассчитывается по формуле:

$$\mathcal{E}_{3e} = \frac{Z_{cm} \cdot (T_{c1} - T_{c2}) / T_q}{D_p}, \quad (4.37)$$

где Z_{cm} – среднемесячная заработная плата одного программиста, руб.

T_{c1}, T_{c2} – снижение трудоемкости на одну задачу в базовом и новом варианте соответственно, чел.-ч.;

T_q – количество часов работы в день, ч.;

D_p – среднемесячное количество рабочих дней.

Экономия с учетом начислений на заработную плату (\mathcal{E}_n):

$$\mathcal{E}_n = \mathcal{E}_3 \times K_{нз}, \quad (4.38)$$

					ДНВ.508190.009.ПЗ	Лист
						42
Изм.	Лист	№ докум.	Подпись	Дата		

где $K_{\text{нз}}$ – коэффициент начислений на заработную плату. Рассчитывается по данным заказчика.

Экономия за счет сокращения простоя сервиса (Θ_c):

$$\Theta_c = \frac{D_{\text{рг}} * (\Pi_1 - \Pi_2) * C_{\text{п}}}{60}, \quad (4.39)$$

где Π_1, Π_2 – время простоя сервиса, обусловленное программным обеспечением в день;

$D_{\text{рг}}$ – плановый фонд работы сервиса (дней);

$C_{\text{п}}$ – стоимость одного часа, руб.;

Общая экономия от применения нового программного обеспечения рассчитывается по формуле:

$$\Theta_o = \Theta_n + \Theta_c \quad (4.40)$$

Плановый объем выполняемых работ в год составляет 400 задач. Среднемесячная заработная плата программиста в бюджетной организации 634,62 руб. Средняя трудоемкость работ в расчете на 1 задачу 5,0 и 0,6 человеко-часов в базовом и новом варианте программного обеспечения соответственно. Количество часов работы в день 8 часов. Среднемесячное количество рабочих дней 22 дня.

Экономия затрат на заработную плату в расчете на 1 задачу составит:

$$\Theta_{\text{зе}} = \frac{634,62 * (5 - 0,6) / 8}{22} = 15,86 \text{ (бел. руб.)} \quad (4.41)$$

Экономия затрат на заработную плату на весь объем запланированных задач составит:

$$\Theta_z = 15,86 \times 400 = 6346 \text{ (бел. руб.)} \quad (4.42)$$

Экономия с учетом начислений на заработную плату:

$$\Theta_n = 6346 \times 1,2 = 7615,4 \text{ (бел. руб.)} \quad (4.43)$$

Экономия от размещения рекламы на своем сайте, а не на сторонних составляет:

$$P_r = C_1 * \frac{Q_p}{1000} * 12, \quad (4.44)$$

$$P_r = 60 * \frac{10000}{1000} * 12 = 7200 \text{ (бел. руб.)}, \quad (4.45)$$

где P_r – годовая стоимость рекламы;

C_1 – стоимость 1 тысячи показов рекламы;

Q_p - количество показов рекламы.

Но так как появился сайт, для него нужен хостинг и домен. Стоимость хостинга и домена в год составляет 100 бел. руб.

$$\Theta_c = \frac{225 \cdot (20-10) \cdot 0,38}{60} = 14,25 \text{ (бел. руб.)} \quad (4.46)$$

Рассчитаем сумму общей экономии денежных средств:

$$\Theta_o = 7612,8 + 14,25 + 7200 - 100 = 14729,65 \text{ (бел. руб.)} \quad (4.47)$$

Таким образом, общая экономия от применения разработанного программного обеспечения составит 7629,65 бел. руб./год.

В заключение написания экономической части дипломного проекта формулируются соответствующие выводы об эффективности нового продукта (предлагаемого в дипломном проекте) по сравнению с аналогами по техническим параметрам, экономическим показателям и эксплуатационным характеристикам.

4.4.3 Расчет экономической эффективности разработки ПО для заказчика.

В качестве показателей, характеризующих экономическую эффективность ПО, принимается размер экономии денежных средств в результате внедрения нового ПО, рентабельность капитальных вложений, срок окупаемости капитальных вложений на замену ПО.

Рентабельность капитальных вложений заказчика в приобретение ПО:

$$P_{\text{кв}} = \frac{\Delta C_0}{K_{\text{рк}}} \cdot 100\%, \quad (4.48)$$

$$P_{\text{кв}} = \frac{14729,65}{6763,14} \cdot 100\% = 218\%, \quad (4.49)$$

где ΔC_0 – размер экономии денежных средств, полученный в результате внедрения ПО заказчиком;

$K_{\text{рк}}$ – капитальные вложения на приобретение ПО заказчиком, бел. руб.

$P_{\text{г}}$ – годовая стоимость рекламы

Период окупаемости капитальных вложений:

$$T = K_{\text{рк}} / (\Delta C_0 + P_{\text{г}}), \quad (4.50)$$

$$T = \frac{6763,14}{14729,65} = 0,46 \text{ лет.} \quad (4.51)$$

Результаты расчета экономической эффективности ПО для заказчика представлен в виде таблицы 7.8.

Таблица 7.8 - Показатели экономической эффективности ПО.

Показатели	Условные обозначения	Единицы измерения	Значения показателей
Капитальные вложения заказчика (стоимость ПО у разработчика)	ОЦ	бел. руб.	6763,14
Экономия денежных средств заказчиком	ΔСо	бел. руб.	14729,65
Рентабельность	Ркв	%	218
Срок окупаемости капитальных вложений	Ток	лет	0,46

В экономической части проекта представлено обоснование необходимости вывода продукта на рынок и произведены следующие расчеты по разработке ПО: затраты времени на этапы разработки, затраты на материалы и комплектующие, затраты на электроэнергию, заработную плату персоналу, амортизацию материальных и нематериальных активов, приобретение оборудования для разработки, расчет прибыли, отчисления на социальные нужды и налоговые отчисления. Представлен расчет экономической эффективности разработки ПО для (пользователя) заказчика.

Капитальные вложения заказчика на приобретение и внедрение ПО составляют 6763,14 бел. руб. Экономия денежных средств составила для заказчика 14729,65 бел. руб. Рентабельность капитальных вложений составила 218%. Срок окупаемости капитальных вложений равен 0,46 года. Исходя из показателей можно сделать вывод что приобретение и внедрение ПО заказчиком экономически целесообразно.

ЗАКЛЮЧЕНИЕ

Главным итогом дипломного проектирования является разработанный программный продукт «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей.

В ходе реализации дипломного проекта были решены следующие задачи:

- изучение предметной области;
- проведение анализа аналогов и прототипов с описанием преимуществ и недостатков рассматриваемых систем;
- выявление требований к разрабатываемому программному обеспечению;
- разработка технического задания;
- проектирование программного обеспечения;
- реализация программного продукта;
- тестирование готового программного продукта;
- разработка сопроводительной программной документации («Описание программы», «Программа и методика испытаний»);
- проведение расчетов цены и себестоимости программного продукта.

При разработке программного продукта использовались следующие технологии:

1 Серверная часть:

- веб-сервер Docker;
- веб-сервер .Net core;
- СУБД MSSQL;
- серверный язык программирования – ASP.NET core.

2 Клиентская часть:

- Сервер AngularJS;
- HTML – язык гипертекстовой разметки;
- SCSS – то метаязык на основе CSS, предназначенный для увеличения уровня абстракции CSS кода и упрощения файлов каскадных таблиц стилей;
- TypeScript - язык программирования, позиционируемый как средство разработки веб-приложений, расширяющее возможности JavaScript.

В качестве сред разработки использовались:

- интегрированная среда разработки Visual Studio 2019;
- редактор исходного кода Visual Studio Code.

Разработанное программное средство полностью соответствует требованиям технического задания.

					ДНВ.508190.009.ПЗ	Лист
						46
Изм.	Лист	№ докум.	Подпись	Дата		

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Раздел «Microsoft SQL Server 2008» на сайте Microsoft [Электронный ресурс]. – Режим доступа: <http://www.microsoft.com/sqlserver/2008/ru/ru/product-information.aspx>. – Дата доступа: 11.04.2020.
- 2 Арлоу Д., Саливон И.И. С# для профессионалов: Учеб. пособие. – Мн: Технология, 2007.
- 3 Трудовой кодекс Республики Беларусь: текст Кодекса по состоянию на 3 ноября 2006 г. – Мн.: Амалфея, 2006. – 272с.
- 4 Диаграмма вариантов использования [Электронный ресурс] / Портал знаний. – Режим доступа: http://www.znannya.org/?view=uml_use_case_diagram_4. – Дата доступа: 16.04.2020.
- 5 Арлоу Д., Нейшталд А. UML и Унифицированный процесс: практический объектно-ориентированный анализ и проектирование, 2-е издание. – Пер. с англ. – СПб: Символ-Плюс, 2007. – 624 с.
- 6 Дженифер Тидвелл. Разработка пользовательских интерфейсов, 2011. – 480 с.
- 7 Система вопросов и ответов (веб-сервис) [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Система_вопросов_и_ответов_\(веб-сервис\)](https://ru.wikipedia.org/wiki/Система_вопросов_и_ответов_(веб-сервис)) – Дата доступа: 19.04.2020.
- 8 Web API 2 в ASP.NET [Электронный ресурс]. – Режим доступа: https://metanit.com/sharp/aspnet_webapi/1.1.php – Дата доступа: 02.06.2020.
- 9 Angular| Введение и начало работы [Электронный ресурс]. – Режим доступа: <https://metanit.com/web/angular2/1.1.php> – Дата доступа: 29.05.2020.

ПРИЛОЖЕНИЕ А

(обязательное)

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Введение

Наименование продукта: «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей.

Продукт будет использоваться для взаимодействия пользователей с целью получения ответов на вопросы, а также для поиска ответов по сайту.

Также программный продукт позволяет просматривать список отвеченных вопросов того или иного пользователя.

А.1 Основание для разработки

Основанием для разработки является приказ № 256 от «05» 05 2020 г. «О закреплении тем дипломных проектов студентов факультета информационных технологий специальности 1-40 01 01 «Программное обеспечение информационных технологий», специализации 1-40 01 01 01 «Веб-технологии и программное обеспечение мобильных систем».

Тема разработки – разработано «Web-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей», выданная студенту группы 16ИТ-1 Дубовскому Никите Владимировичу, руководителем назначен старший преподаватель кафедры технологий программирования Пяткин Дмитрий Владимирович.

А.2 Назначение разработки

Основной целью разработки приложения является получение ответов на интересующие вопросы. Приложение планирует предоставлять некоторые возможности.

Каждый авторизованный пользователь сможет задать вопрос, а также ответить на вопросы, которые оставили другие пользователи сервиса. Для более удобного, практичного и гибкого использования возможностей web-приложения будет реализовано прикрепление различного рода файлов, размер которых не должен превышать 50 мегабайт.

					ДНВ.508190.009.ПЗ	Лист
						48
Изм.	Лист	№ докум.	Подпись	Дата		

Пользователь будет имеет возможность просмотреть учетные записи других пользователей и подписаться на понравившихся людей. Кроме того, планируется воплотить систему оценивания ответов на вопросы.

Пользователь может воспользоваться поиском, по ключевым словам, меткам. Так он узнает, отвечал ли уже кто-то на аналогичный вопрос.

А.3 Требования к программе или программному изделию

А.3.1 Требования к функциональным характеристикам

К функциональным характеристикам выдвигаются следующие требования:

- пользователь может просматривать все вопросы, не имеющие ответов;
- пользователь может просматривать учетные записи других пользователей (с их ответами);
- пользователь может задать вопрос;
- пользователь может ответить на вопрос;
- пользователь имеет возможность подписаться на другого пользователя;
- пользователь может найти свой вопрос уже имеющий ответ на него.

А.3.2 Требования к надежности

Программное средство должно обеспечивать контроль корректности входных данных. В случае обнаружения ошибок во входных данных пользователю должны выводиться соответствующие сообщения.

А.3.3 Условия эксплуатации

Данное программное средство не требует предварительной подготовки для использования.

А.3.4 Требования к составу и параметрам технических средств

					ДНВ.508190.009.ПЗ	Лист
						49
Изм.	Лист	№ докум.	Подпись	Дата		

Минимальными и достаточными требованиями по конфигурации оборудования клиента для функционирования системы являются:

- процессор с тактовой частотой 1350 МГц и выше;
- оперативная память 2 Гб и выше;
- монитор;
- клавиатура, мышь;
- подключение к сети Интернет (со скоростью соединения не менее 1 мбит/с);
- установленный один из популярных веб-браузеров последней версии;
- наличие свободного места на жестком диске не менее 50 МБ для кэширования браузера.

Сервер обработки запросов должен соответствовать следующей конфигурации:

- процессор с тактовой частотой не менее 1350 МГц;
- не менее 2 Гб оперативной памяти;
- наличие свободного места на жестком диске не менее 1 Гб.

А.3.5 Требования к информационной и программной совместимости

Для программного обеспечения клиента требуется:

- Операционная система: Windows 7, 8, 10.

Для программного обеспечения сервера требуется:

- Microsoft SQL Server 2018;
- Docker.

А.3.6 Требования к маркировке и упаковке

Требования к маркировке и упаковке отсутствуют.

А.3.7 Требования к транспортированию и хранению

Требования к транспортировке и хранению отсутствуют.

А.4 Требования к программной документации

					ДНВ.508190.009.ПЗ	Лист
						50
Изм.	Лист	№ докум.	Подпись	Дата		

Программная документация должна состоять из следующих документов:

- техническое задание в соответствии с ГОСТ 19.201-78 ЕСПД;
- описание программы в соответствии с ГОСТ 19.402-2000 ЕСПД;
- руководство оператора в соответствии с ГОСТ 19.505-79 ЕСПД;
- программа и методика испытаний в соответствии с ГОСТ 19.301-2000 ЕСПД.

Содержание и структура программной документации соответствует требованиям ЕСПД.

А.5 Технико-экономические показатели

Эффективность данного программного продукта в основном обуславливается отсутствием лишнего функционала и стабильной работой данного приложения.

А.6 Этапы разработки

Этапы разработки:

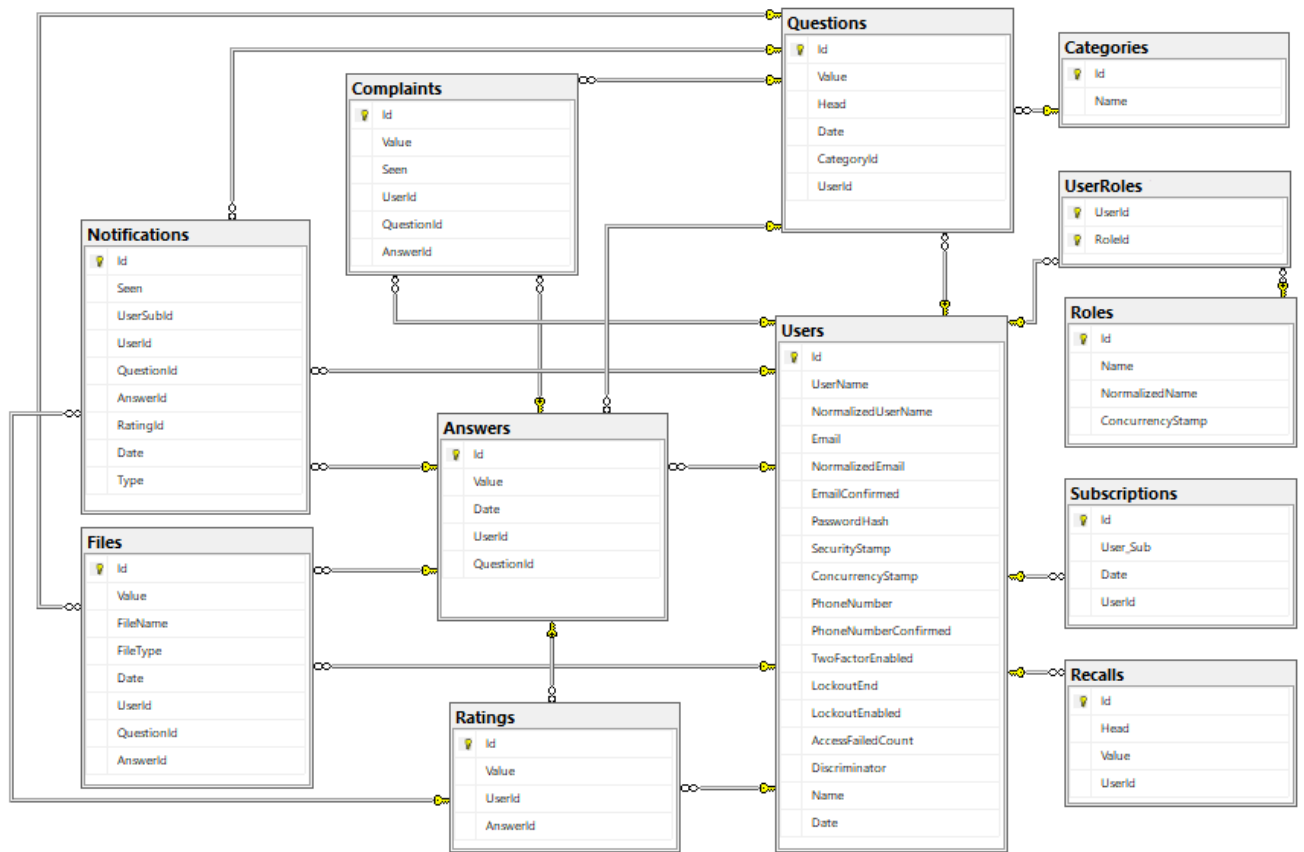
- постановка задачи;
- анализ исходных данных;
- разработка программного средства;
- тестирование программного средства;
- разработка технической документации к данному программному средству.

А.7 Порядок контроля и приемки

Для контроля выполнения требования технического задания необходимо провести испытания. Порядок и состав испытания определяются программой и методикой испытаний.

Приемка готового программного средства осуществляется комиссией по защите дипломного проекта в соответствии с программой и методикой испытаний.

ПРИЛОЖЕНИЕ Б (рекомендуемое) **ДИАГРАММА МОДЕЛИ ДАННЫХ**



ПРИЛОЖЕНИЕ В (обязательное) **ОПИСАНИЕ ПРОГРАММЫ**

В.1 Общие сведения

Наименование программного средства – «Web-приложение «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей».

Данное программное средство предоставляет пользователям удобный и легкий веб-сервис по поиску ответов на вопросы.

Для успешного функционирования программного продукта необходимы:

- операционная система из семейства Windows;
- подключение к сети Интернет;
- установленный один из популярных веб-браузеров последней версии.

Интерфейс программного продукта разработан при помощи языка гипертекстовой разметки HTML5, метаязыка SCSS и JavaScript-фреймворка AngularJS.

В.2 Функциональное назначение

Данное программное средство предназначено для решения следующих задач:

- предоставление пользователям возможность просматривать уже заданные вопросы и ответы к ним;
- предоставление возможности задавать вопросы;
- предоставление возможности оставить ответ, поставить лайки другим ответам и подписаться на других пользователей.

В.3 Описание логической структуры

Данный программный продукт работает по следующему алгоритму:

- для запуска приложения требуется открыть веб-браузер;
- в адресной строке веб-браузера нужно ввести адрес веб-сайта, после чего откроется главная страница веб-сайта;
- для входа в панель администрирования нужно нажать на кнопку «Sing In/Sing Up», после чего откроется страница авторизации, ввести данные администратора и войти в систему.

					ДНВ.508190.009.ПЗ	Лист
						53
Изм.	Лист	№ докум.	Подпись	Дата		

В.4 Используемые технические средства

Минимальная аппаратная конфигурация системы пользователя, при которой обеспечивается стабильная работа программного средства, должна быть не ниже следующей:

- процессор с тактовой частотой 1350 МГц и выше;
- оперативная память 2 Гб и выше;
- монитор;
- клавиатура, мышь;
- подключение к сети Интернет (со скоростью соединения не менее 1 мбит/с);
- установленный один из популярных веб-браузеров последней версии;
- наличие свободного места на жестком диске не менее 50 МБ для кэширования браузера.

Сервер обработки запросов должен соответствовать следующей конфигурации:

- процессор с тактовой частотой не менее 1350 МГц;
- не менее 2 Гб оперативной памяти;
- наличие свободного места на жестком диске не менее 1 Гб.

В.5 Вызов и загрузка

Загрузка программного средства осуществляется при запуске веб-сервера. Вызов приложения осуществляется при отправке любого запроса серверу.

В.6 Входные данные

Для разрабатываемого программного продукта входной будет служить следующая информация:

- информация о пользователе;
- информация о задаваемом вопросе;
- информация об ответе.

В.7 Выходные данные

Выходными данными являются сгенерированные сервером веб-страницы, которые отправляются пользователю.

					ДНВ.508190.009.ПЗ	Лист
						54
Изм.	Лист	№ докум.	Подпись	Дата		

ПРИЛОЖЕНИЕ Г
(обязательное)
ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ

Г.1 Объект испытаний

Объект испытаний «Web-приложение «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей».

Г.2 Цель испытания

Проверка программного продукта «Web-приложение «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей», установление его работоспособности, соответствия техническому заданию, выявление несоответствий в работе программы и в программной документации.

Г.3 Требования к программе

Разработанный программный продукт должен быть разделен на два функциональных модуля:

- модуль администратора;
- модуль пользователя.

Модуль администратора управляет данными, удаляет пользователей/вопросы/ответы, не соответствующие политике приложения, описанной заказчиком. Модуль пользователя позволяет осуществлять просмотр страниц веб-сайта, добавление вопросов и возможность добавлять ответы.

Г.4 Требования к программной документации

Проект должен содержать следующие программные документы:

- описание программы;
- руководство оператора;
- руководство администратора.

					ДНВ.508190.009.ПЗ	Лист
						55
Изм.	Лист	№ докум.	Подпись	Дата		

Г.5 Средства и порядок испытаний

Г.5.1 Средства для испытаний

Для проведения испытаний над веб-сайтом необходимо наличие персонального компьютера, с возможностью выхода в сеть Интернет. Для проверки корректности функционирования разработанного программного обеспечения должны применяться следующие программные средства:

- процессор с тактовой частотой 1350 МГц и выше;
- оперативная память 2 Гб и выше;
- монитор;
- клавиатура, мышь;
- подключение к сети Интернет (со скоростью соединения не менее 1 мбит/с);
- установленный один из популярных веб-браузеров последней версии;
- наличие свободного места на жестком диске не менее 50 МБ для кэширования браузера.

Сервер обработки запросов должен соответствовать следующей конфигурации:

- процессор с тактовой частотой не менее 1350 МГц;
- не менее 2 Гб оперативной памяти;
- наличие свободного места на жестком диске не менее 1 Гб.

Г.5.2 Порядок испытаний

Проверка выполняется в соответствии с тест-планом, представленном в таблице Г.1.

Таблица Г.1 – Порядок выполнения тестирования

Тестовый вариант	Входные данные	Ожидаемый результат	Результат тестирования
Тестирование клиентского модуля			
Запуск клиентского модуля	Вход на сайт	Открытие главной страницы	Тест пройден успешно
Переход на страницу «UserList»	Нажатие на пункт меню «UserList»	Открытие страницы «UserList»	Тест пройден успешно

Тестовый вариант	Входные данные	Ожидаемый результат	Результат тестирования
Переход к определенной категории	Нажатие на раздел меню «Category» и выбрать категорию	Открытие страницы выбранной категории	Тест пройден успешно
Переход в раздел «Sing In»	Нажатие на раздел меню «Sing In/Sing Up»	Открытие раздела с «Sing In»	Тест пройден успешно
Переход в раздел «Sing Up»	Нажатие на раздел меню «Sing In/Sing Up» и перейти в вкладку «Sing Up»	Открытие раздела «Регистрация»	Тест пройден успешно
Переход на страницу с отзывами после авторизации	Нажатие на кнопку «Recall»	Открытие страницы с отзывами	Тест пройден успешно
Добавление нового комментария после авторизации	Нажатие на кнопку «Add Answer»	Новый комментарий успешно добавлен	Тест пройден успешно
Редактирование фотографии после авторизации	Нажатие на кнопку «Upload»	Редактирование прошло успешно	Тест пройден успешно
Поиск вопросов и пользователей после авторизации	Ввод данных в поле ввода	Поиск прошел успешно	Тест пройден успешно
Подписка и отписка от других пользователей после авторизации	Нажатие на кнопку «subscribe/unsubscribe»	Подписка/Отписка прошла успешно	Тест пройден успешно
Добавление лайков понравившемуся комментарию после авторизации	Нажатие на кнопку «Лайк»	Лайк успешно добавлен	Тест пройден успешно
Переход в личный кабинет пользователя после авторизации	Нажатие на кнопку «Profile»	Открытие личного кабинета пользователя	Тест пройден успешно
Переход на страницу добавления вопроса после авторизации	Нажатие на раздел меню «Add question»	Открытие раздела «Add question»	Тест пройден успешно

Тестовый вариант	Входные данные	Ожидаемый результат	Результат тестирования
Тестирование модуля администратора			
Редактирование информации о пользователях	Изменение названия и фото пользователя; Нажатие кнопки «Сохранить»	Измененная информация о пользователях сохранена	Тест пройден успешно
Удаление пользователя	Нажатие на раздел меню «User List»; нажатие на кнопку «Delete» выбранного пользователя	Удаление выбранного пользователя	Тест пройден успешно
Удаление вопроса	Нажатие на раздел «All questions»; Нажатие на кнопку «Delete» выбранного вопроса	Удаление выбранного вопроса	Тест пройден успешно
Удаление комментария	Нажатие на вопрос; нажатие на кнопку «Delete» выбранного комментария	Удаление выбранного комментария	Тест пройден успешно
Просмотр жалоб	Нажать на раздел меню «Complaints»	Открытие страницы с жалобами	Тест пройден успешно

Г.6 Методы испытаний

Проверка данного программного средства должна осуществляться методом «Черного ящика», который предназначен для проверки функционального назначения программы.

Корректность выполнения определяется сравнением фактического и ожидаемого результатов. Если результаты совпали, то тестирование пройдено успешно. Если же результаты не равны или произошел сбой системы, и пользователю вывелось системное сообщение об ошибке, то это свидетельствует о том, что тестирование пройдено неуспешно.

ПРИЛОЖЕНИЕ Д
(обязательное)
РУКОВОДСТВО ОПЕРАТОРА

Д.1 Назначение программы

«Web-приложение «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей» предоставляет пользователям возможность найти ответы на интересующие вопросы или же задать свои вопросы.

Д.2 Условия выполнения программы

Для того чтобы работа с программным средством на компьютере пользователя, компьютер должен иметь минимальную аппаратную комплектацию: монитор, системный блок, клавиатура, мышь, доступ к сети Интернет.

Минимальная аппаратная конфигурация системы пользователя, при которой обеспечивается стабильная работа программного средства, должна быть не ниже следующей:

- процессор с тактовой частотой 1350 МГц и выше;
- оперативная память 2 Гб и выше;
- соединение с сетью интернет (со скоростью соединения не менее 1 мбит/с).

Сервер обработки запросов должен соответствовать следующей конфигурации:

- процессор с тактовой частотой не менее 1350 МГц;
- не менее 2 Гб оперативной памяти;
- наличие свободного места на жестком диске не менее 1 Гб.

Если данные требования будут соблюдены, то программное средство будет корректно работать на компьютере пользователя.

Для работы с программным средством пользователь должен обладать минимальными навыками работы в сети Интернет.

					ДНВ.508190.009.ПЗ	Лист
						59
Изм.	Лист	№ докум.	Подпись	Дата		

Д.3 Выполнение программы

Программный продукт выполнен в виде веб-приложения, в зависимости от запросов сервер генерирует необходимые данные и отправляет их пользователю.

Для начала использования приложения следует открыть веб-браузер и ввести в адресную строку адрес веб-сайта, после чего откроется главная страница веб-сайта (рисунок Д.1). На главной странице можно увидеть навигационное меню, а также на главной странице выводятся информация о последний ответах, последних вопросов и рейтинг пользователей.

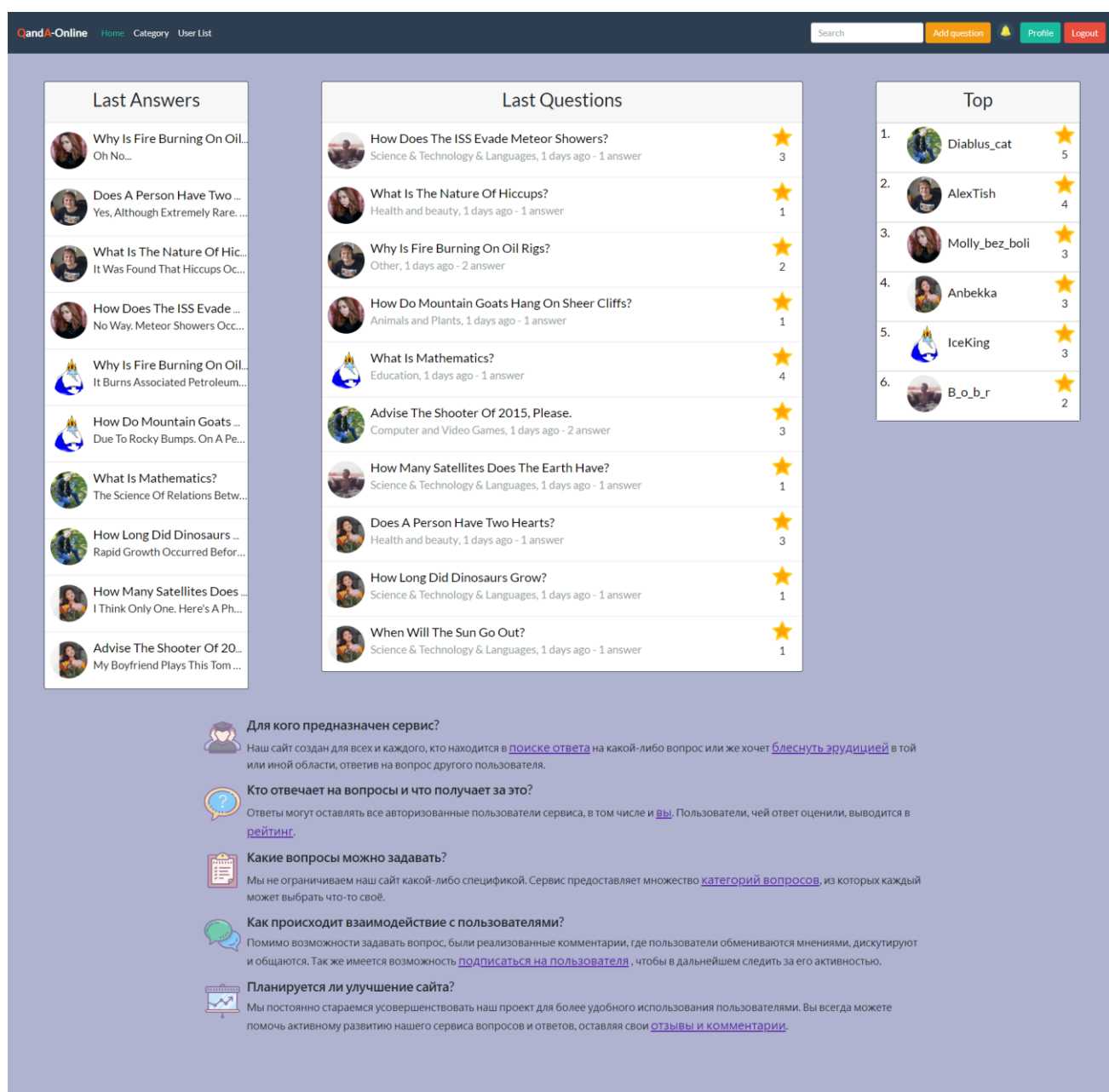


Рисунок Д.1 – Главная страница веб-сайта

При помощи навигационного меню можно осуществлять навигацию по разделам веб-сайта.

Первым в списке навигационного меню идет выпадающий список «Category», на которой содержится информация о вариантах категорий. На рисунке Д.2 изображен выпадающий список.

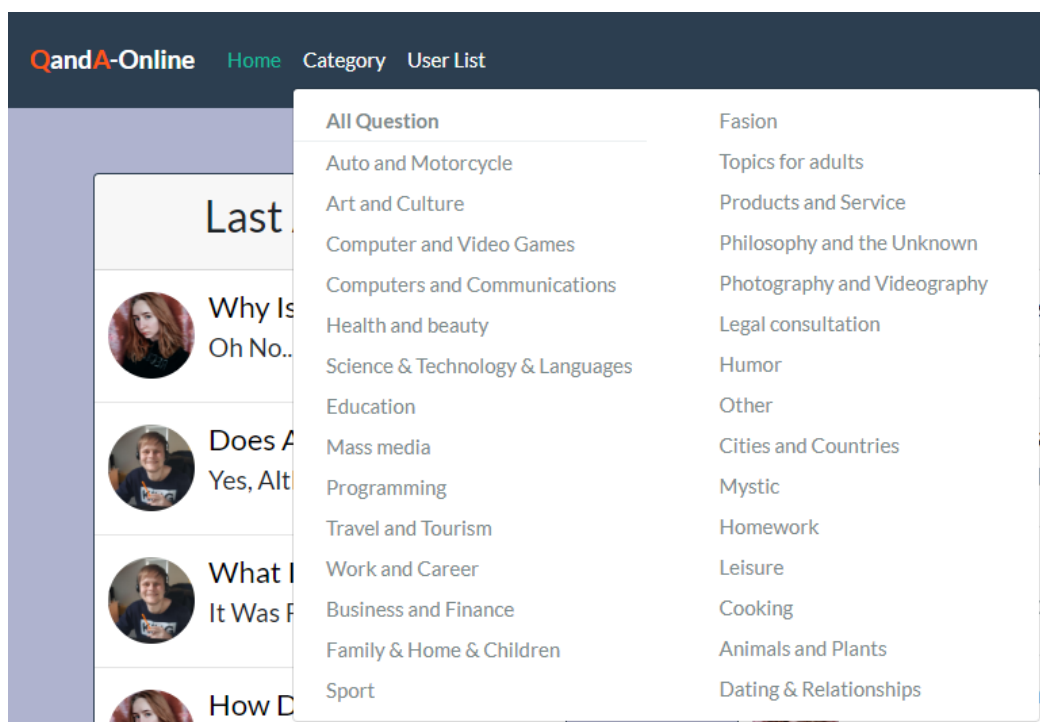


Рисунок Д.2 – Выпадающее меню

В разделе «User List», пользователь может увидеть список пользователей (рисунок Д.3).

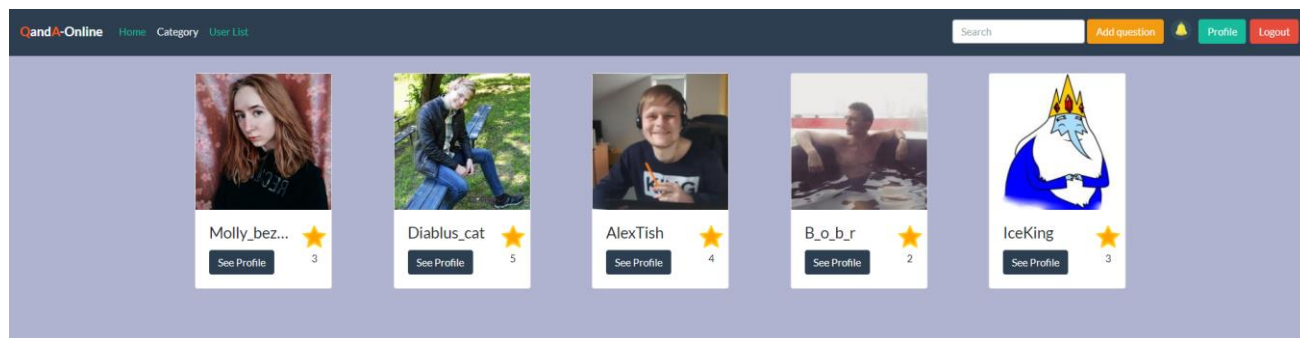


Рисунок Д.3 – Страница раздела «User List»

Для просмотра полной информации о пользователе необходимо нажать на аватарку одного из пользователей, после чего откроется страница профиля пользователя, представленная на рисунке Д.4.

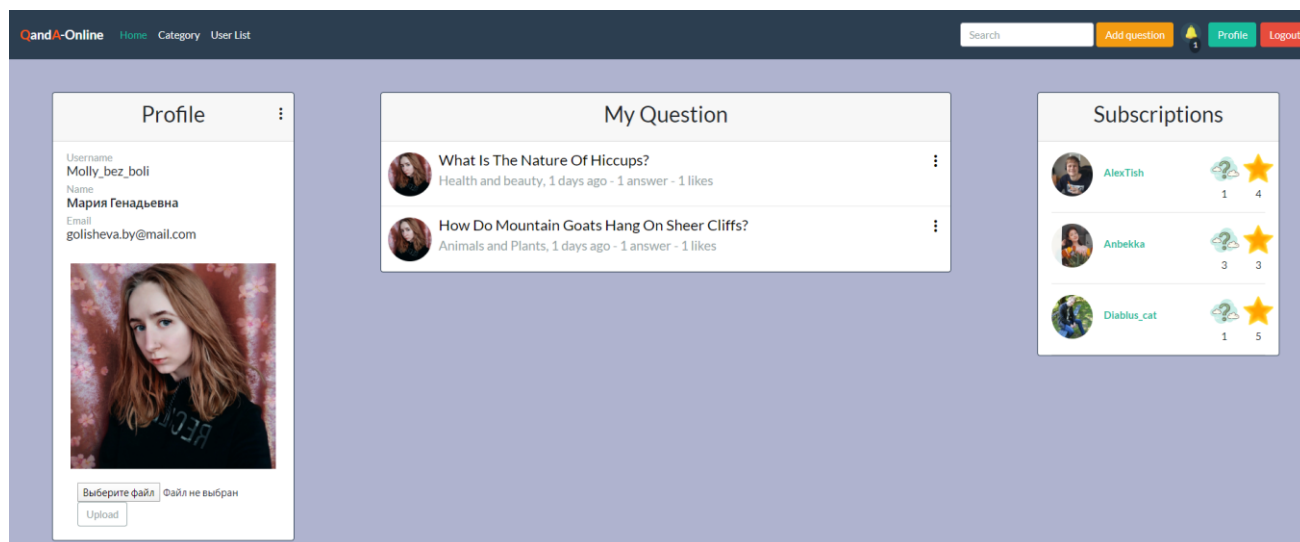


Рисунок Д.4 – Страница профиля

Для того чтобы у пользователя была возможность задавать вопросы и отвечать на них необходимо зарегистрироваться и авторизоваться в системе, нажав на кнопку «Sing In/Sing Up», после чего откроется страница регистрации (рисунок Д.5) и авторизации (рисунок Д.6).

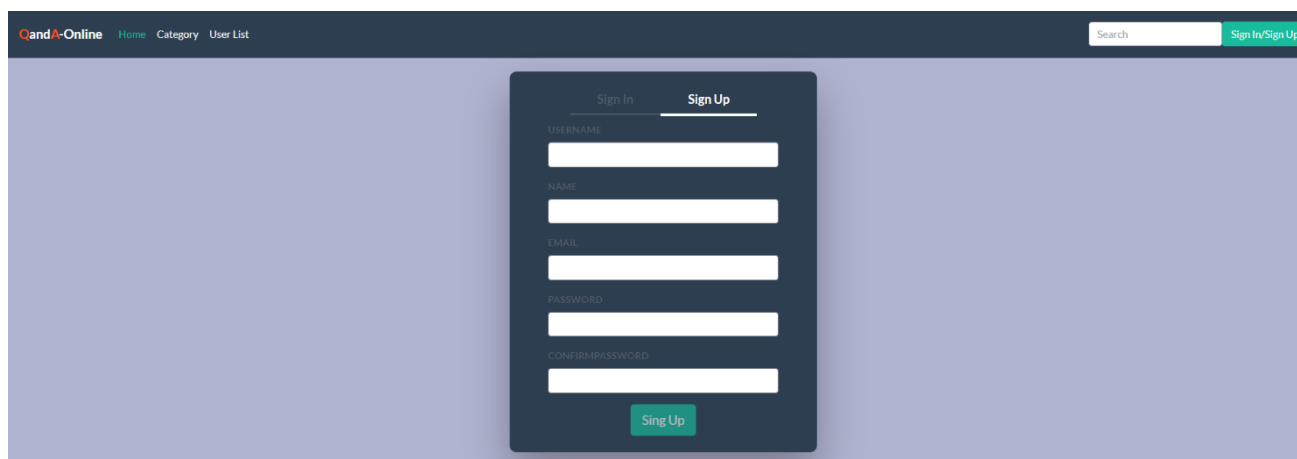


Рисунок Д.5 – Страница регистрация

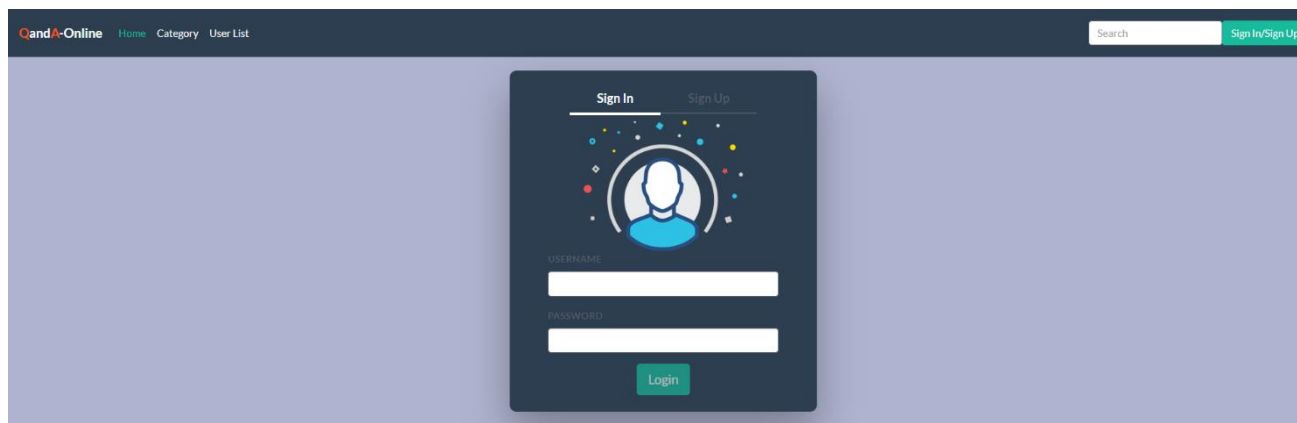


Рисунок Д.6 – Страница авторизации

После успешной авторизации у пользователя появится возможность задавать вопросы и отвечать на них, оставлять лайки и подписываться на других пользователей. Страница с добавлением вопроса представлена на рисунок Д.7. Страница с просмотром вопроса и добавлением ему ответа представлена на рисунке Д.8.

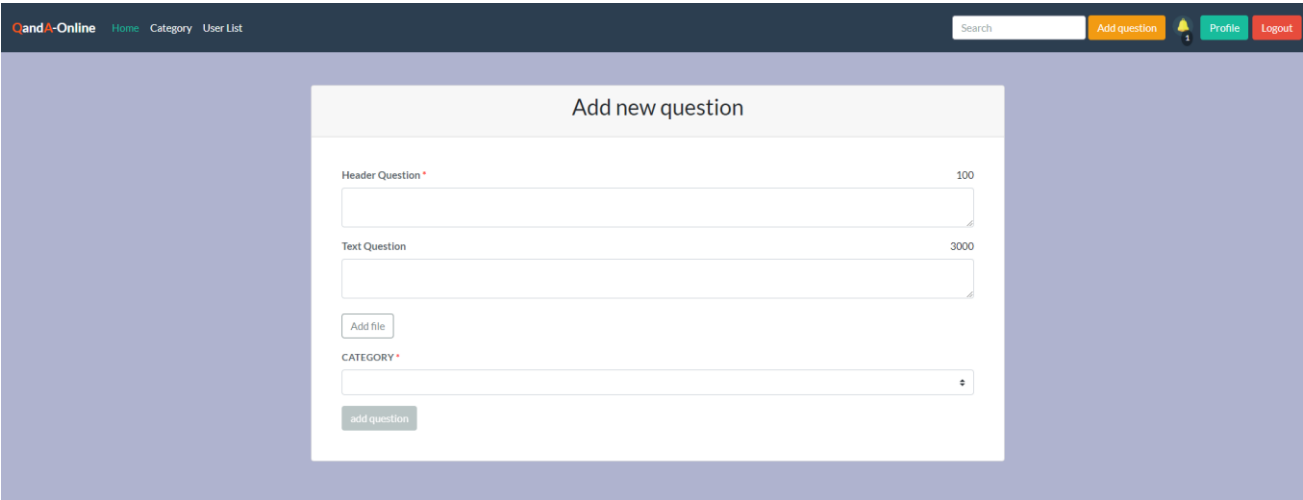


Рисунок Д.7 – Страница с добавлением вопроса

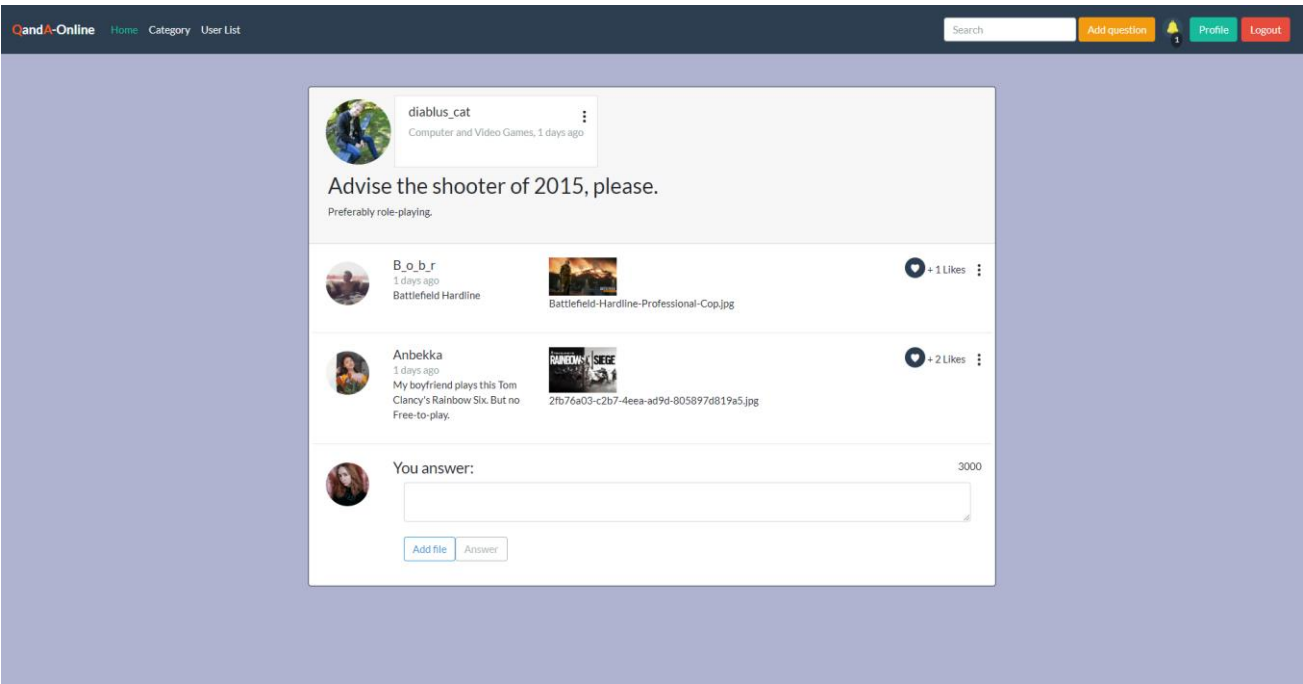


Рисунок Д.8 – Страница просмотра вопроса

ПРИЛОЖЕНИЕ Е
(обязательное)
РУКОВОДСТВО АДМИНИСТРАТОРА

Е.1 Назначение программы

«Web-приложения «Служба вопрос-ответ QandA-Online» для социального взаимодействия пользователей» предоставляет администратору следующие возможности:

- просмотр информации о пользователях, его вопросах, ответах, подписках;
- возможность редактирования профиля пользователя;
- удаления пользователя, вопроса, ответа;
- просмотр жалоб.

Е.2 Условия выполнения программы

Для того чтобы работа с программным средством на компьютере администратора, компьютер должен иметь минимальную аппаратную комплектацию: монитор, системный блок, клавиатура, мышь, доступ к сети Интернет.

Минимальная аппаратная конфигурация системы пользователя, при которой обеспечивается стабильная работа программного средства, должна быть не ниже следующей:

- процессор с тактовой частотой 500 МГц и выше;
- процессор с тактовой частотой 1350 МГц и выше;
- оперативная память 2 Гб и выше;
- соединение с сетью интернет (со скоростью соединения не менее 1 мбит/с).

Сервер обработки запросов должен соответствовать следующей конфигурации:

- процессор с тактовой частотой не менее 1350 МГц;
- не менее 2 Гб оперативной памяти;
- наличие свободного места на жестком диске не менее 1 Гб.

Если данные требования будут соблюдены, то программное средство будет корректно работать на компьютере пользователя.

Для работы с программным средством пользователь должен обладать минимальными навыками работы в сети Интернет.

Е.3 Выполнение программы

Программный продукт выполнен в виде веб-приложения, в зависимости от запросов сервер генерирует необходимые данные и отправляет их пользователю.

Для начала использования приложения следует открыть веб-браузер и ввести в адресную строку адрес веб-сайта. Далее произойдет загрузка главной веб-страницы. Для авторизации пользователя нужно нажать на кнопку «Sing In/Sing Up» и ввести данные администратора.

После успешной авторизации администратора, осуществится переход на профиль авторизованного пользователя. На данной странице можно увидеть, навигационное меню, при помощи которого можно осуществлять навигацию по разделам панели управления (рисунок Е.1).



Рисунок Е.1 – Навигационное меню для администратора

На странице просмотра вопроса можем наблюдать три вертикальные точки, при нажатии на них открывается выпадающее меню с действиями, такая же кнопка имеется у ответов (рисунок Е.2).

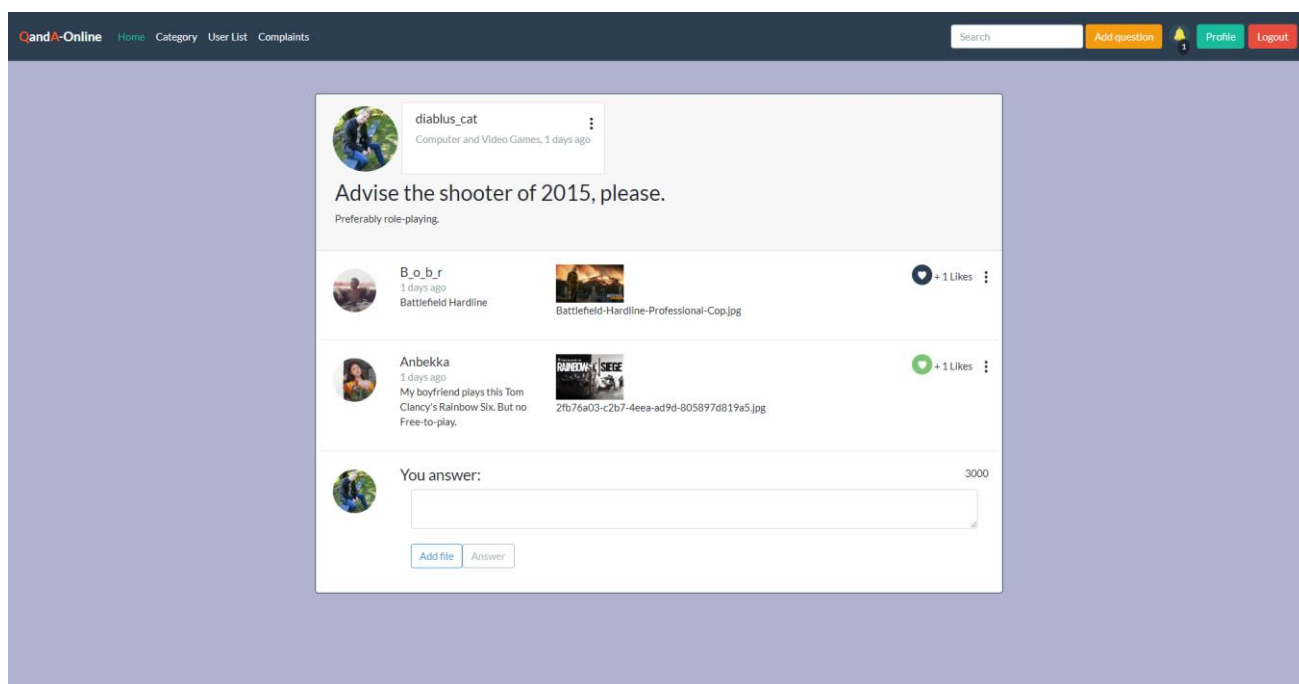


Рисунок Е.2 – Страница просмотра вопроса

При переходе к странице пользователя так же есть возможность дополнительных действий (рисунок Е.3):

- редактирование профиля (рисунок Е.4);
- удаление профиля;
- просмотр жалоб (рисунок Е.5-Е.7).

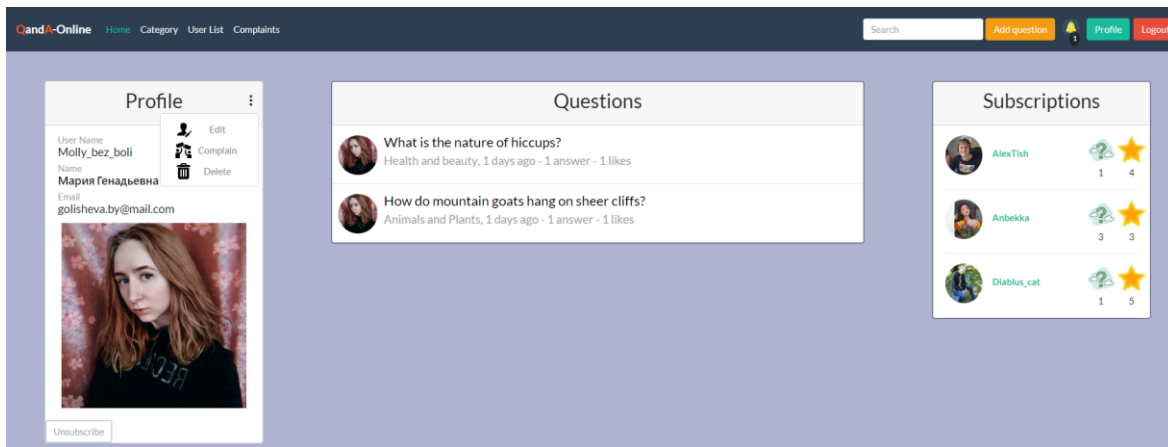


Рисунок Е.3 – Страница профиля пользователя

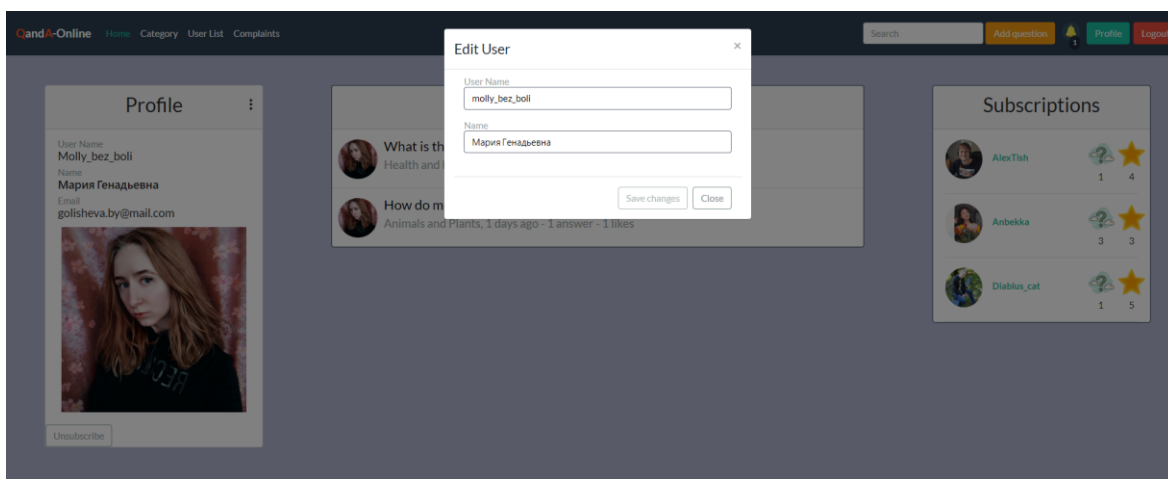


Рисунок Е.4 – Редактирование профиля пользователя

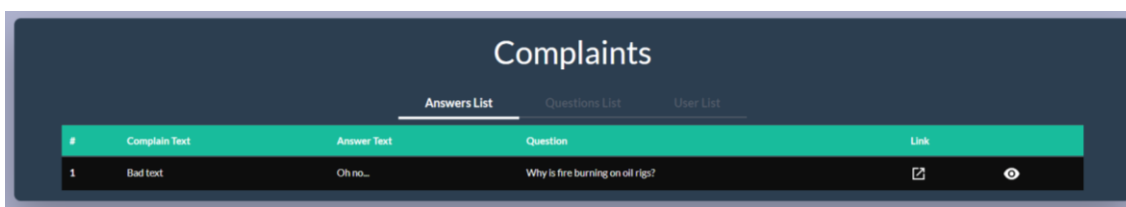


Рисунок Е.5 – Просмотр жалоб на ответы

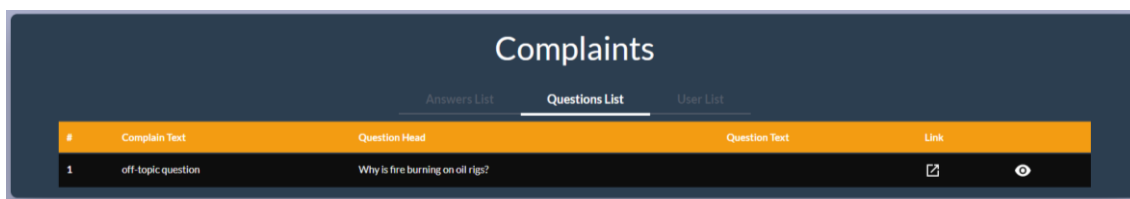


Рисунок Е.6 – Просмотр жалоб на вопросы

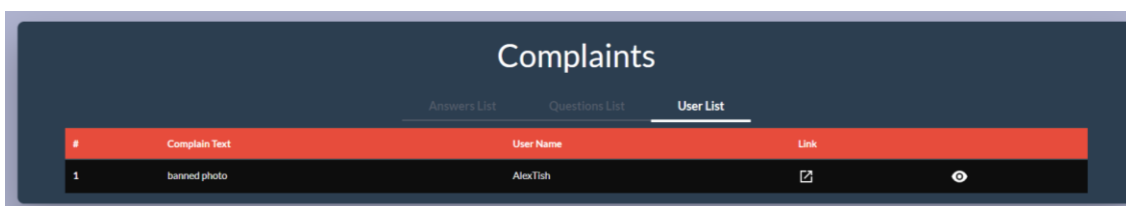
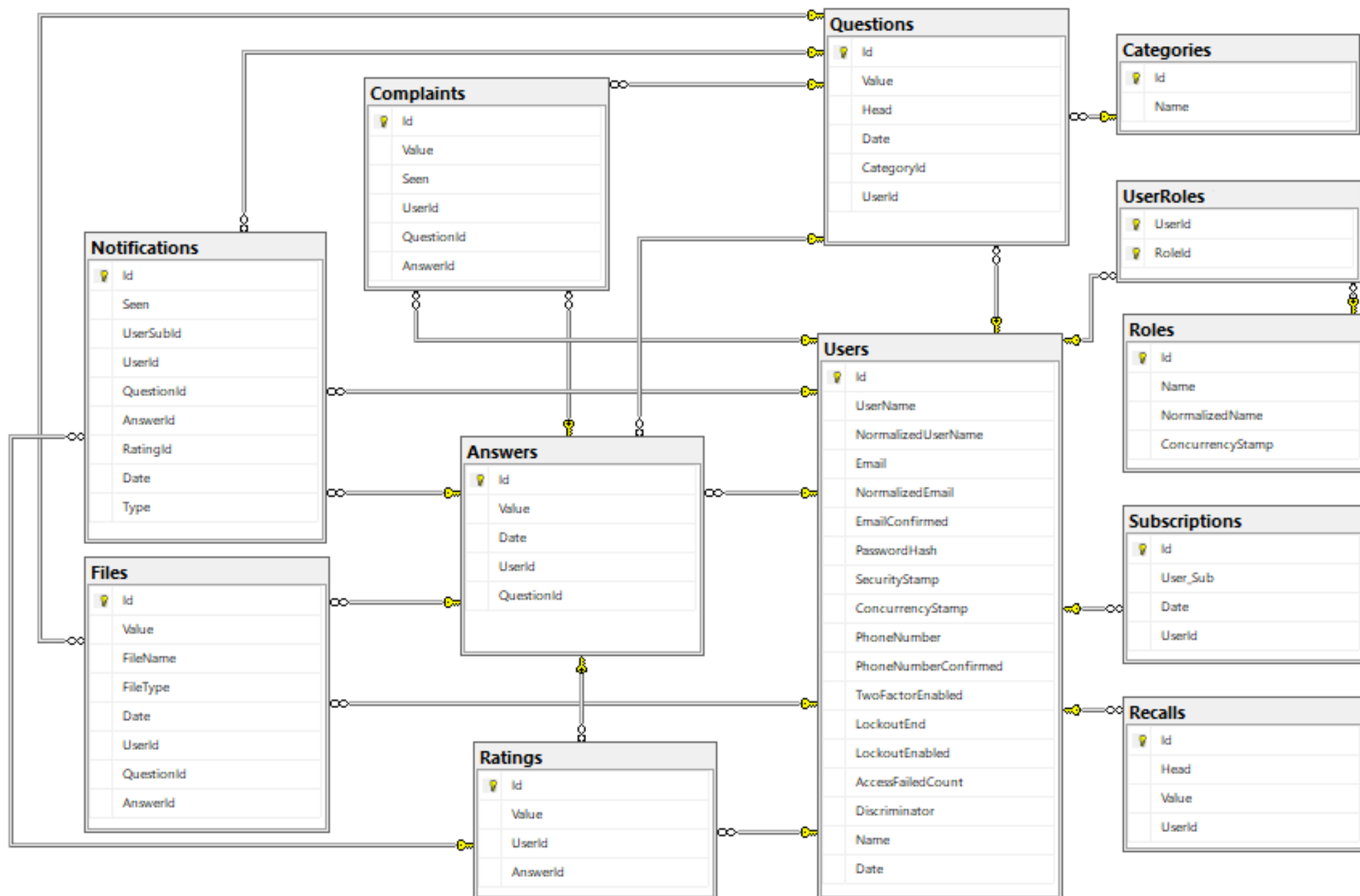


Рисунок Е.7 – Просмотр жалоб на пользователей

СХЕМА БАЗЫ ДАННЫХ



ЭКРАННЫЕ ФОРМЫ ПРИЛОЖЕНИЯ

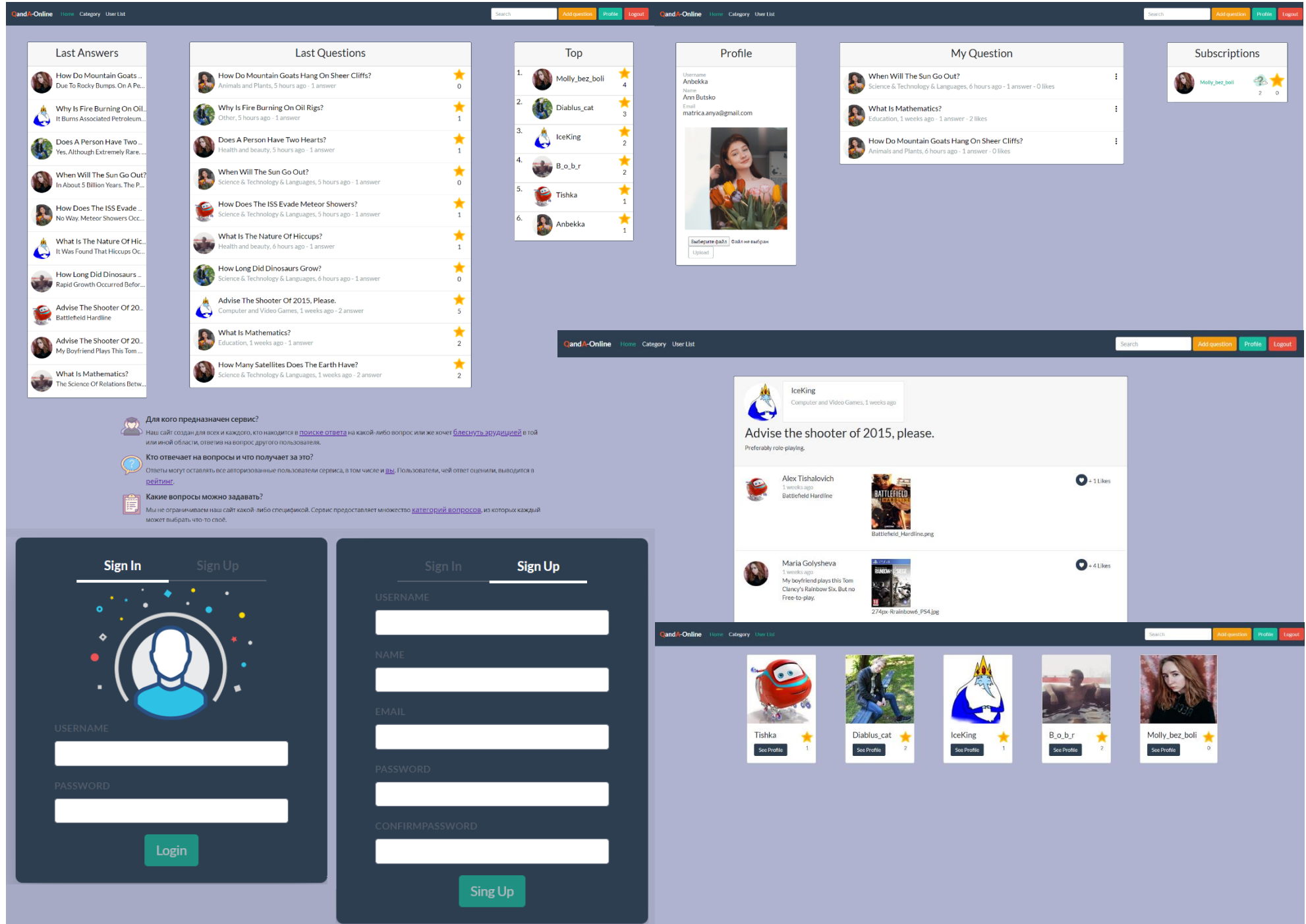


ДИАГРАММА СЕБЕСТОИМОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Статья затрат	Сумма затрат, руб.
Материалы и комплектующие M	20,50
Электроэнергия \mathcal{E}	180,48
Основная заработная плата \mathcal{Z}_o	2207,42
Дополнительная заработная плата \mathcal{Z}_d	331,11
Отчисления на социальные нужды $\mathcal{Z}_{сз}$	878,33
Амортизация A	244,84
Накладные расходы P_n	172
Плановая себестоимость:	4034,68

Диаграмма распределения статей затрат на разработку ПО

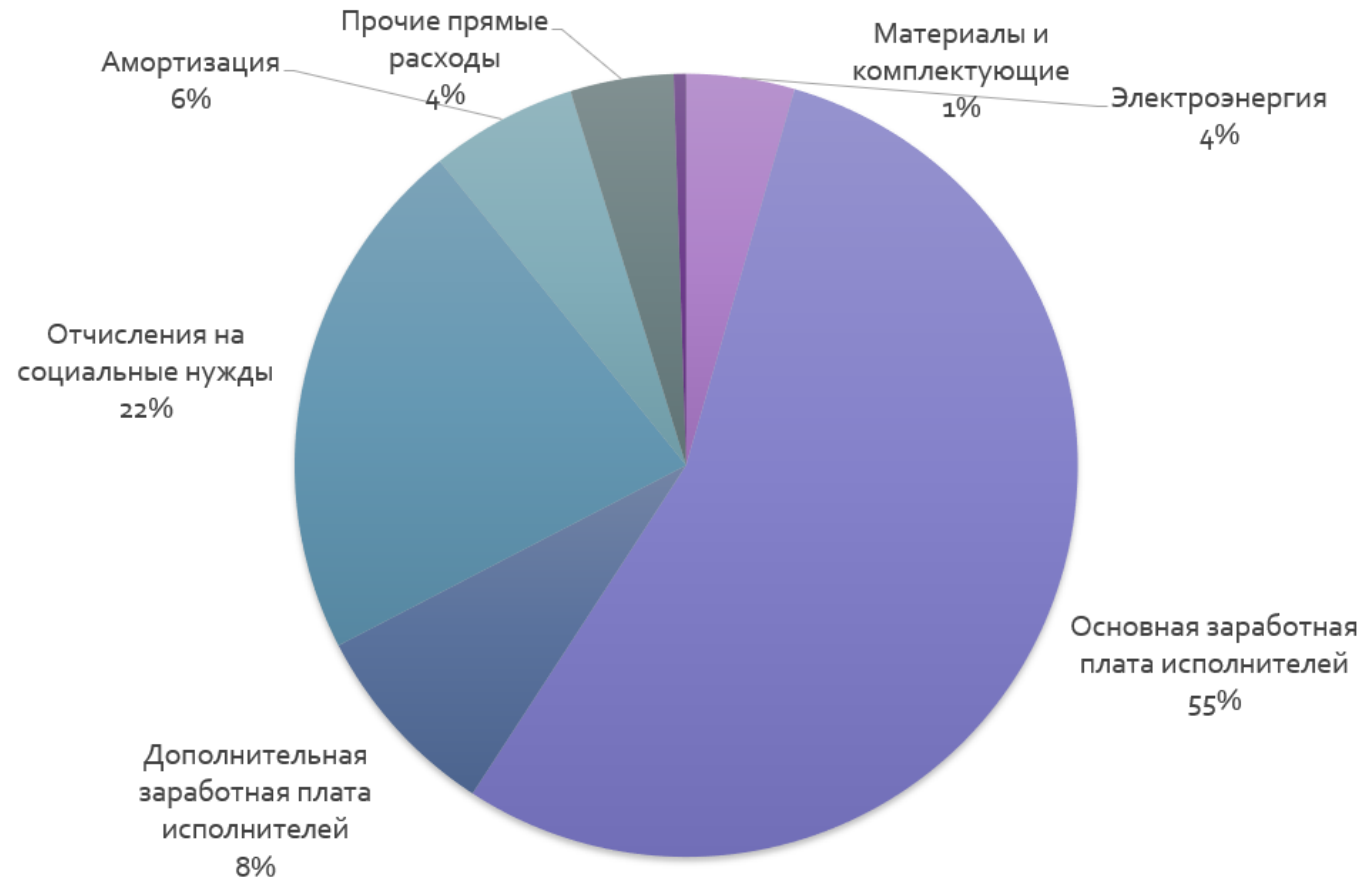
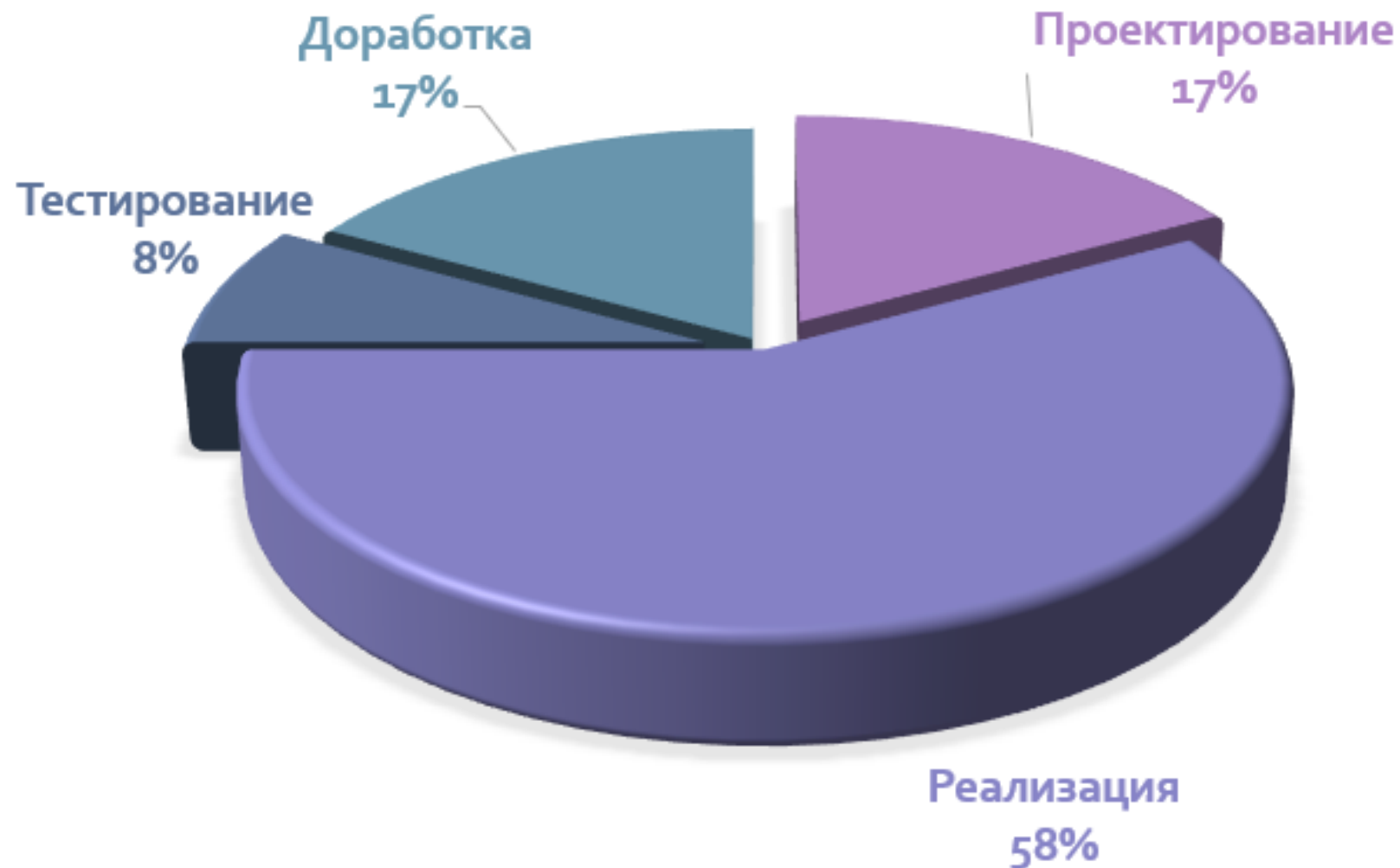


ДИАГРАММА РАСПРЕДЕЛЕНИЯ ВРЕМЕНИ РАБОТЫ ПРИ РАЗРАБОТКЕ ПРОДУКТА



[illegible]