

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
Кафедра «Прикладная математика и информатика»

02.03.03 МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ И
АДМИНИСТРИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ

БАКАЛАВРСКАЯ РАБОТА

на тему **Разработка программного обеспечения контакт-центра
департамента урегулирования убытков страховой компании «РЕСО-
Гарантия»**

Студент	<u>А.О. Бритовский</u>	_____
Руководитель	<u>А.И. Туищев</u>	_____
Консультант по аннотации	<u>А.В. Кириллова</u>	_____

Допустить к защите
Заведующий кафедрой к.тех.н., доцент, А.В. Очеповский _____

« _____ » _____ 2017 г.

Тольятти 2017

АННОТАЦИЯ

Тема: «Разработка программного обеспечения контакт-центра департамента урегулирования убытков страховой компании Ресо-Гарантия».

Ключевые слова: РАЗРАБОТКА, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, КОНТАКТ-ЦЕНТР, УРЕГУЛИРОВАНИЕ УБЫТКОВ, СТРАХОВАЯ КОМПАНИЯ.

Целью данной бакалаврской работы является разработка программного обеспечения контакт-центра департамента урегулирования убытков страховой компании РЕСО-Гарантия.

Объект исследования - контакт-центр департамента урегулирования убытков страховой компании.

Предмет исследования - автоматизация работы контакт-центр департамента урегулирования убытков страховой компании.

Методы исследования: Web-технологии, методы администрирования информационных систем, объектно-ориентированное программирование.

На основе объектно-ориентированного подхода разработана логическая модель программного обеспечения (ПО) контакт-центра. На основе Web-технологии PHP+MySQL разработано ПО и даны рекомендации по его аппаратно-программному обеспечению.

Тестирование ПО подтвердило его работоспособность.

Работа включает: страниц 56 с приложением, рисунков 33, таблиц 1, источников 20.

ABSTRACT

The title of graduation work is Software Development of the Loss Adjustment Department Call-center of the Insurance Company Reso-Garantia. This graduation work is dedicated to the programs created for insurance companies. These programs can solve problems that usually occur during communication and insurance of a customer and company via mobile phone.

In this graduation work we consider the system of the call-center of the loss adjustment department call-center of the insurance company. The key question of the graduation work is the development of a program that solves the problem with the connection of the customer and the manager of the insurance company. The work covers the following: description of the system of the call-center in the insurance company Reso-Garantia, analysis of call-center loss adjustment of insurance companies, identify gaps, call-centers, insurance companies, representation of the decision problem for the contact center.

In this work it is shown that the customer cannot communicate with the insurance company via mobile phone, as it revealed the discomfort and much time needed. The program presented in this graduation work solves these problems.

The work is of interest for a wide range of customers. The results show that the existing system is inefficient and requires improvements to optimize the process.

The work includes: pages 56 with applications, figures 33, tables 1, references 20.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	4	
Глава 1. АНАЛИЗ ДЕЯТЕЛЬНОСТИ КОНТАКТ-ЦЕНТРА ДЕПАРТАМЕНТА УРЕГУЛИРОВАНИЯ УБЫТКОВ СТРАХОВОЙ КОМПАНИИ		6
1.1 Технико-экономическая характеристика предметной области	6	
1.2 Назначение контакт-центра и его особенности.....	8	
1.3 Анализ имеющихся отделов по урегулированию убытков.....	9	
1.4 Назначение и цели создания сайта, функции выполняемые сайтом...	11	
1.5 Требования к программному обеспечению	12	
Глава 2. РАЗРАБОТКА ПРОЕКТНЫХ РЕШЕНИЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОНТАКТ-ЦЕНТРА ДЕПАРТАМЕНТА УРЕГУЛИРОВАНИЯ УБЫТКОВ СТРАХОВОЙ КОМПАНИИ.....		14
2.1 Разработка логической модели контакт-центра департамента урегулирования убытков страховой компании.....	14	
2.2 Диаграмма вариантов использования	14	
2.3 Диаграмма классов.....	16	
2.4 Диаграмма последовательности подачи заявки.....	17	
Глава 3. РЕАЛИЗАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОНТАКТ- ЦЕНТРА ДЕПАРТАМЕНТА УБЫТКОВ СТРАХОВОЙ КОМПАНИИ.....		19
3.1 Модель системы разработки	19	
3.1.1 Выбор средств реализации.....	19	
3.1.2 Выбор языков разметки.....	21	
3.1.3 Выбор СУБД.....	22	
3.1.4 Выбор сервера	24	
3.1.5 Описание серверной части.....	26	
3.2 Реализация интерфейса контакт-центра департамента урегулирования убытков страховой компании	27	
3.3 Структура программного обеспечения и требования к аппаратному обеспечению	41	

ЗАКЛЮЧЕНИЕ	46
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ	47
ПРИЛОЖЕНИЕ А Исходный код контроллера регистрации	49
ПРИЛОЖЕНИЕ Б Реализация формы заявления о страховом случае	52

ВВЕДЕНИЕ

Урегулирование убытков – это комплекс мероприятий, проводимых страховщиком в целях выполнения обязательств перед страхователем при наступлении страхового случая.

Урегулирование убытка начинается с подачи страхователем заявления о страховом случае (иначе - страховой претензии) и завершается (если страховщик признал, что действительно имел место страховой случай) осуществлением страховой выплаты выгодоприобретателю.

Контакт-центр – является централизованной службой, которая используется для приема или передач большого объема запросов по телефону и иным каналам связи. Входящий вызов центр управляется компанией для управления входящими поддержки продукта или информационные запросы со стороны потребителей. Контакт-центр является местом для централизованной обработки индивидуальных сообщений, в том числе письма, факсов , программного обеспечения онлайн поддержки , социальных средств массовой информации, обмен мгновенных сообщений и электронной почты.

Было решено разработать программное обеспечение контакт-центра, работающее через электронную почту мобильного телефона, что делает данную тему актуальной.

Объектом исследования является контакт-центр департамента урегулирования убытков страховой компании.

Предметом исследования является автоматизация работы контакт-центр департамента урегулирования убытков страховой компании.

Целью данной бакалаврской работы является разработка программного обеспечения контакт-центра департамента урегулирования убытков страховой компании РЕСО-Гарантия.

Для достижения поставленной цели, необходимо решить следующие задачи:

- провести анализ литературы по проведению урегулирования убытков;
- определить структуру страховой компании «РЕСО-Гарантия»;
- спроектировать ИС, реализующую функции контакт-центра департамента урегулирования страховой компании;
- разработать контакт-центр выбранными средствами реализации;
- тестирование разработанного контакт-центра с целью выявления недостатков и их устранения.

В ходе данной выпускной квалификационной работы (ВКР) будет разработано приложение для автоматизированного контакт-центра департамента урегулирования убытков страховой компании.

Выпускная квалификационная работа (ВКР) состоит из введения, трех глав и заключения.

Во введении описывается актуальность рассматриваемой темы, определяются объект и предмет ВКР работы, ставится цель и выявляются задачи.

В первой главе происходит анализ предметной области и формализация требований к программному продукту.

Во второй главе описывается разработка и реализация проектных решений программного обеспечения контакт-центра департамента урегулирования убытков страховой компании.

В третьей главе описывается реализация интерфейса и происходит тестирование разработанного приложения в соответствии с формализованными требованиями к пользовательскому интерфейсу.

В заключении подводятся итоги разработки программного обеспечения, формируются окончательные выводы по рассматриваемой теме.

В приложении представлены фрагменты программного кода.

Глава 1. АНАЛИЗ ДЕЯТЕЛЬНОСТИ КОНТАКТ-ЦЕНТРА ДЕПАРТАМЕНТА УРЕГУЛИРОВАНИЯ УБЫТКОВ СТРАХОВОЙ КОМПАНИИ

1.1 Технико-экономическая характеристика предметной области

Предметной областью ВКР работы является контакт-центр департамента урегулирования убытков страховой компании.

Контакт-центр – это специализированное подразделение в организации, занимающиеся обработкой обращений и информированием по голосовым каналам связи, электронной и обычной почте или факсу.

От того насколько хорошо автоматизирован процесс по заключению договоров страхования в страховых компаниях зависят многие показатели:

- скорость и качество обслуживания клиентов;
- введение новой информации и возможность редактирования;
- трудозатраты на информирование сотрудников;
- механизмы контроля и мониторинга;
- возможности быстрого получения необходимой информации [4].

Основная цель контакт-центра департамента урегулирования убытков страховой компании заключается в том, что бы помочь клиенту выбрать страховую программу или подать заявку на получение компенсации за страховой случай.

1.1.1 Характеристика предприятия

Страховое публичное акционерное общество «РЕСО-Гарантия» – универсальная страховая компания, созданная в 1991 году. Компания имеет лицензию на более 100 видов страховых услуг и перестраховочную деятельность.

Сайт компании: www.reso.ru.

В Группу РЕСО входит ряд страховых компаний СНГ - «НСК» (Казахстан), «Белросстрах» (Беларусь), «РЕСО» (Армения), также Холдинг имеет стратегического партнера и на территории Украины – «ПРОСТО-страхование».

Партнерами РЕСО-Гарантия по перестраховочным программам являются компании Munich Re, Hannover Re, SCOR, Sirius, Partner Re, Gen Re и многие другие.

Национальные и международные рейтинги. Начиная с 2009 рейтинговое агентство «Эксперт РА» ежегодно подтверждает РЕСО-Гарантия "Исключительно высокий уровень надежности" А ++ - максимально возможный на сегодня в России.

Служба кредитных рейтингов Standart & Poor's присвоила РЕСО-Гарантия рейтинг "BB".

На рисунке 1 представлена организационная структура СК «РЕСО-Гарантия»

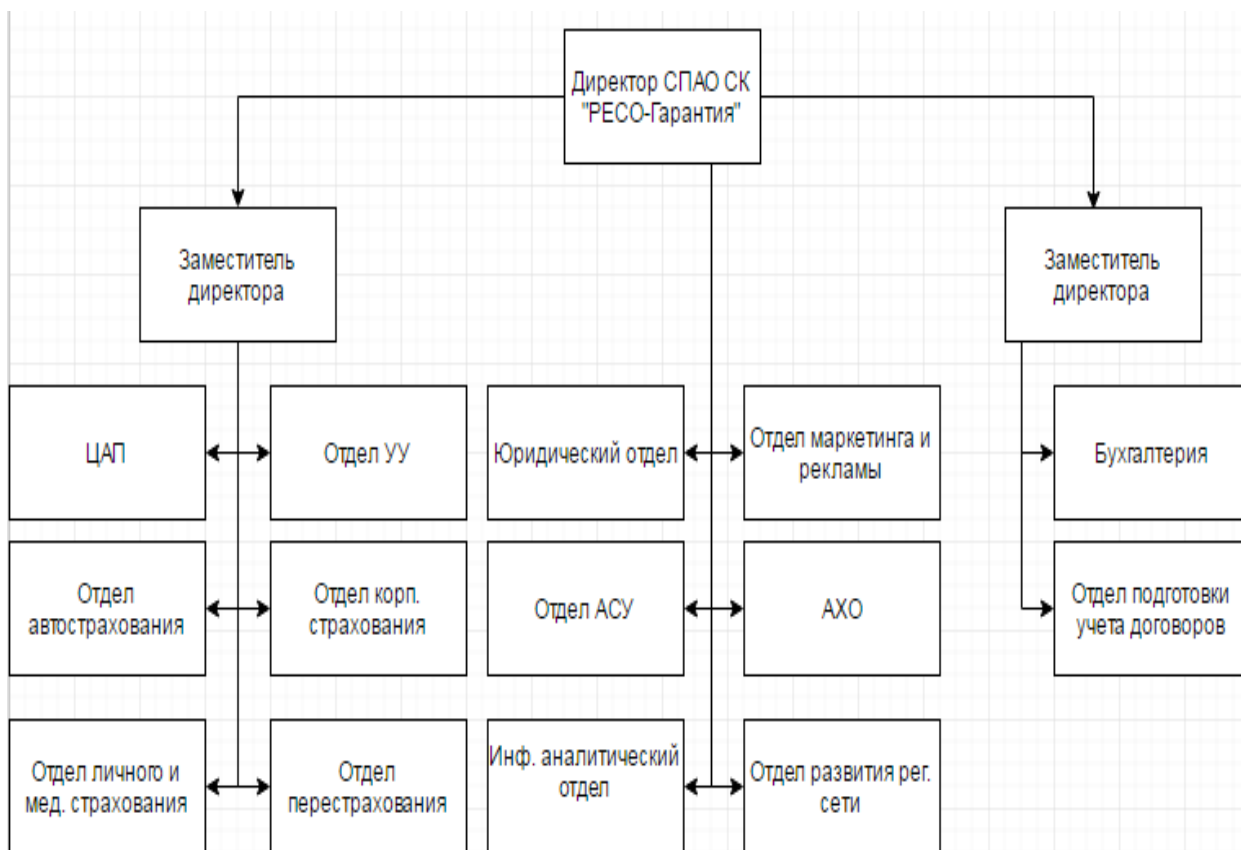


Рисунок 1 – Организационная структура СК

Выплатой убытков страховщикам занимается отдел по урегулированию убытков.

1.2 Назначение контакт-центра и его особенности

Большинство компаний используют центры по обработке звонков для общения со своими клиентами, для организации и сопровождения процессов продаж продуктов и услуг, организации сервисного обслуживания, «первой линии» технической поддержки. Также контакт-центрам может быть вменена функция взаимодействия с подрядчиками, поставщиками. Государственные, политические и общественные организации используют контакт-центры для организации телефонного взаимодействия с гражданами. Крупные предприятия и организации обслуживают некоторые внутренние функции с помощью контакт-центров, например, обеспечивают первую линию офисной и технической поддержки сотрудников.

Для организации обычно используются телефонные сети. В контакт-центрах также применяется широкий спектр программного обеспечения для поддержки взаимодействия через электронную почту, системы мгновенного обмена сообщениями.

Программно-аппаратные решения для организации контакт-центров могут обладать следующими функциональными возможностями:

- регистрация всех входящих и исходящих звонков;
- хранение информации о клиенте;
- статическая и интеллектуальная маршрутизация обращения: организация очереди, маршрутизация по номеру абонента (с использованием информации от автоматических определителей номеров и CallerID);
- визуализация на рабочей станции оператора информации о поступившем звонке и карточке клиента;
- отражение состояния операторов (занят, свободен, пауза);
- распределение звонков внутри группы по загрузке операторов и по порядку;
- формирование отчётов по выполненным и принятым звонкам;
- запись разговоров;

- контроль качества работы операторов.

1.3 Анализ имеющихся отделов по урегулированию убытков

Для успешного создания контакт-центра, был произведен анализ существующих отдела по урегулированию убытков страховых компаний для выявления достоинств и недостатков. Для анализа были взяты страховые компании «Надежда» и «Югория».

Сайт департамента урегулирования убытков страховой компании «Надежда» представлен на рисунке 2.

Ачинск ▾
8 800 350-78-10

ПОИСК

ОНЛАЙН-ПОКУПКА ПОЛИСА
ЭКСПРЕСС-ЗАЯВКА НА СТРАХОВАНИЕ
ПРОВЕРКА ВЫПЛАТ
ЗАПИСЬ НА ВЫПЛАТУ
ЗАДАТЬ ВОПРОС

ЧАСТНЫМ ЛИЦАМ БИЗНЕСУ АКЦИИ О КОМПАНИИ КОНТАКТЫ

Главная

Урегулирование убытков

Страховое общество «Надежда» сохраняет в течение многих лет верхние позиции в российском рейтинге самых оперативных страховщиков по выплатам.

Мы – региональная компания, поэтому все вопросы, связанные с порядком, сроками, размером страховых выплат решаются на месте, в Красноярске, в максимальном короткое сроки.

ОФИСЫ УРЕГУЛИРОВАНИЯ УБЫТКОВ

ЛЕВЫЙ БЕРЕГ	ПРАВЫЙ БЕРЕГ
г. Красноярск, ул. Калинина, 47а – Прием документов по предварительной записи по т. 205-23-08 – Понедельник-четверг 8.30-17.00 – Пятница 8.30 – 16.00	г. Красноярск, ул. Павлова, 14 – Понедельник-четверг 8.30-17.00 – Суббота 8.30 – 16.00

Онлайн запись на выплату

Рисунок 2 – Сайт департамента урегулирования убытков компании «Надежда»

Из достоинств можно выделить:

- возможность создать онлайн запись на выплату;
- понятная навигация сайта.

Из недостатков можно выделить следующее:

- нет возможности авторизации пользователя.

На рисунке 3 представлен сайт департамента урегулирования убытков страховой компании «Югория».

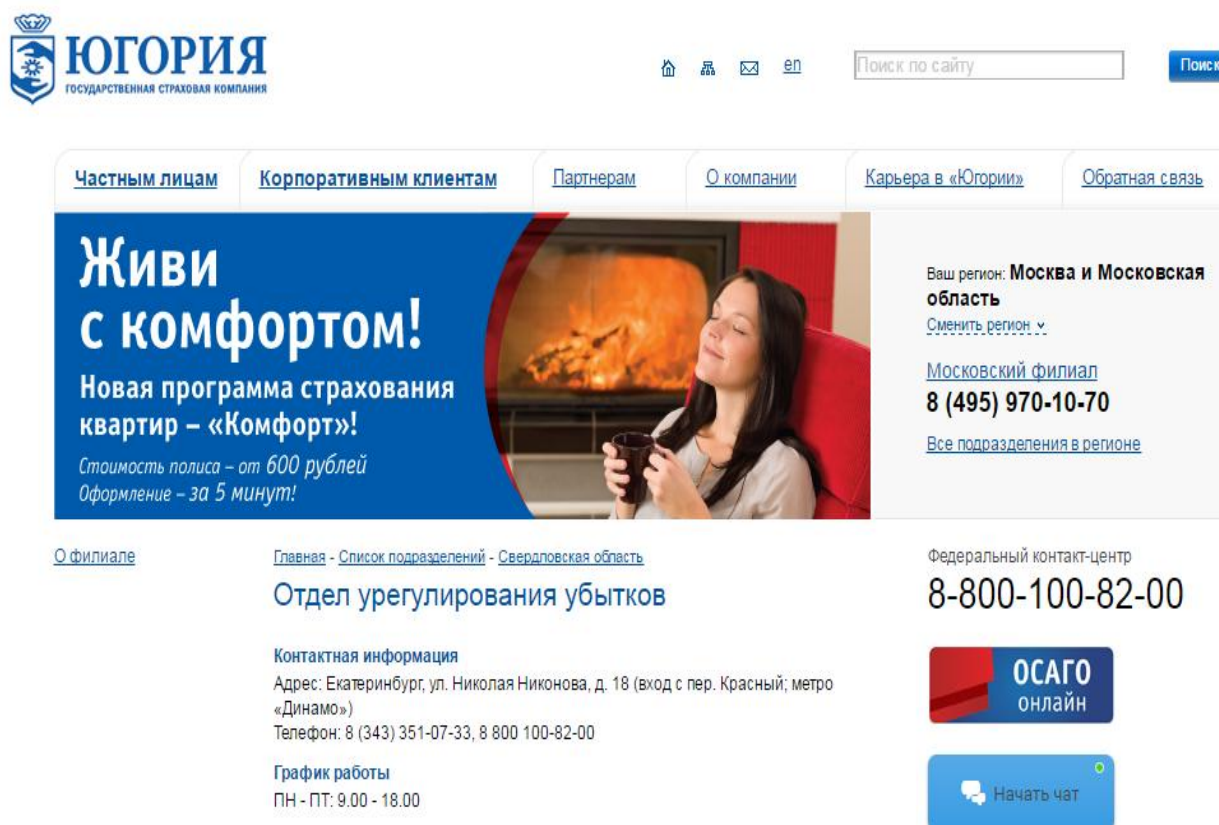


Рисунок 3 – Сайт департамента урегулирования убытков компании «Югория»

Из достоинств данного сайта можно выделить следующее:

- понятный интерфейс;
- возможность обратной связи.

Недостатки данного сайта:

- нет возможности авторизации пользователя;
- нет возможности создать заявку на сайте.

В итоге было проанализировано два отдела по урегулированию убытков страховых компаний. Недостатком является отсутствие личного кабинета, отсутствие возможности подачи заявки. Полученные данные будут использоваться при создании контакт-центра.

1.4 Назначение и цели создания сайта, функции выполняемые сайтом

Для поддержания контакт-центра и эксплуатации интерфейса системы управления сайтом, достаточно иметь общих навыков работы с персональным компьютером и браузером.

Пользователей отдела по урегулированию убытков можно разделить на два типа в соответствии с правами доступа:

- посетители;
- администратор (сотрудник компании).

Посетители имеют доступ исключительно к общедоступной части информации.

Администратор имеет права на добавление, редакцию и удаление разделов, а так же просмотр заявок.

Контакт-центр департамента урегулирования убытков должен состоять из следующих разделов:

- главная страница;
- заключение договоров;
- выплата убытков;
- регистрация и авторизация пользователя;
- список заявок;
- автоматизированная отправка сообщения пользователю на электронную почту, после регистрации и заключения договора.

Стиль сайта департамента урегулирования убытков, должен быть деловым. В качестве фонового цвета будут использоваться светлые цвета. После посещения сайта пользователь должен остаться доволен.

Оформление сайта не должно быть загроможденным, интерфейс должен быть удобным и понятным пользователю.

1.5 Требования к программному обеспечению

Для решения проблемы проекта необходимо разработать контакт-центр департамента урегулирования убытков страховой компании, задачей, которого является взаимодействие клиента с менеджером по урегулированию убытков.

Требования для приложения:

- минимальное время ответа на запросы пользователя;
- простота работы и обслуживания;
- возможность быстрого обучения пользователя.

Требования к функциям контакт-центра:

- удобный интерфейс;
- возможность для менеджера просмотра, редактирования и удаления заявки;

- оптимальная структура базы данных.

Требования к функциональным характеристикам контакт-центра:

- возможность создания заявки на выплату убытков для пользователя;
- пользователь должен иметь определенную возможность ввода, и просмотра имеющейся информации;
- возможность ввода новой информации менеджеру;
- возможность просмотра новых заявок менеджеру;
- автоматизированная отправка сообщения пользователю на почту, после его регистрации и дальнейшего перехода на сайт;
- автоматизированная отправка новых заявок менеджеру на почту.

Выводы по главе

В ходе анализа предметной области были выявлены достоинства и недостатки существующих аналогов отделов по урегулированию убытков, выработаны требования к программному обеспечению.

Анализ показал, что известные ИТ-решения для поддержки контакт-центров страховых компаний не удовлетворяют всем требованиям, поэтому принято решение о разработке нового программного обеспечения контракт-центра страховой компании «РЕСО-Гарантия».

Глава 2. РАЗРАБОТКА ПРОЕКТНЫХ РЕШЕНИЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОНТАКТ-ЦЕНТРА ДЕПАРТАМЕНТА УРЕГУЛИРОВАНИЯ УБЫТКОВ СТРАХОВОЙ КОМПАНИИ

2.1 Разработка логической модели контакт-центра департамента урегулирования убытков страховой компании

На логическом уровне контакт-центр представляется в виде визуальной объектной модели, разработанной с помощью объектно-ориентированного подхода. В процессе проектирования происходит отображение концептуальной модели на логическую схему ее базы данных.

Для разработки логической модели используется унифицированный язык программирования UML.

Совокупность диаграмм описывающие аспекты исследуемой системы образует модель UML.

При логическом моделировании используются диаграммы классов, взаимодействия и использования.

2.2 Диаграмма вариантов использования

Диаграммы вариантов использования описывают взаимоотношения между вариантами использования и действующими лицами.

Диаграммы вариантов использования для упрощения взаимодействия с пользователями системы и клиентами системы, а так же для определения характеристик системы.

Вариант использования (use case) описывает группу действий в системе. Варианты использования – это описание взаимодействий между системой и ее пользователем.

Варианты использования отображают интерфейс системы и указывают форму системы, что та должна сделать.

Варианты использования должны выполнять следующее:

- вариант использования принадлежит, хотя бы к одному действующему лицу;

- вариант использования приводит к результату;
- у каждого варианта есть инициатор.

Варианты использования идентифицируются исходя из следующих соображений: каждый вариант использования представляет собой некоторую функцию, выполняемую системой в ответ на воздействие действующего лица (актера), и характеризует конкретный способ применения системы, диалог между актером и системой.

Действующее лицо (actor) является источником, взаимодействующим с системой с помощью варианта использования. Действующие лица могут быть пользователем системы или компьютерной системой.

На рисунке 4 продемонстрирована диаграмма вариантов использования подачи заявления о страховом случае.

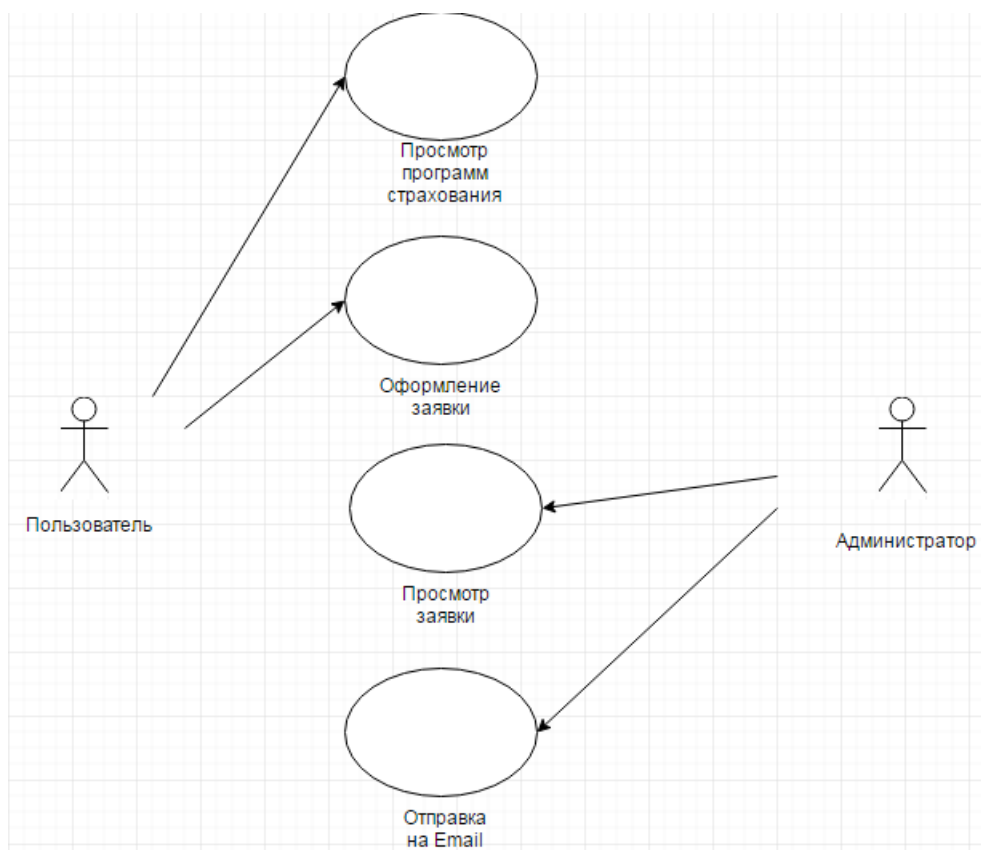


Рисунок 4 – Диаграмма вариантов использования подачи заявления о страховом случае

Клиент просматривает программы страхования или подает заявку для получения информации о выплате убытков. Далее менеджер получает заявку и связывается с клиентом через почту.

2.3 Диаграмма классов

Диаграмма классов – диаграмма, демонстрирующая классы системы, их методы, атрибуты, и взаимосвязи между ними.

На основании диаграммы классов разрабатываются компоненты логической модели системы. Схема наследования типов ее объектов и спецификация.

Разработка диаграммы классов контакт-центра:

- в объектно-структурной модели выделяются классы методом декомпозиции;
- разрабатывается спецификация и строится схема наследования классов.

На рисунке 5 представлена диаграмма классов контакт-центра.

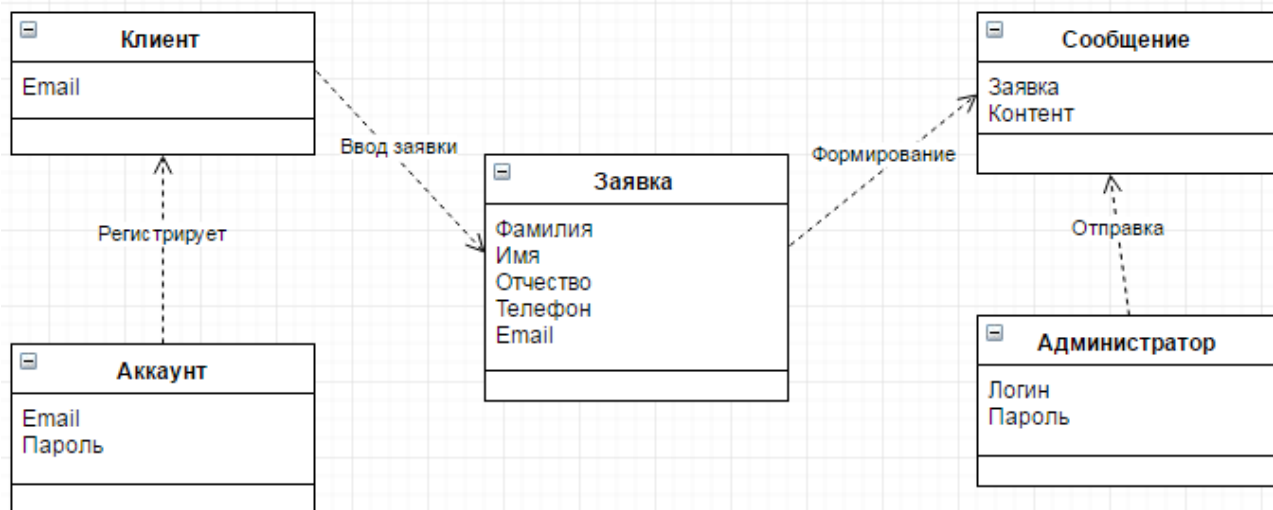


Рисунок 5 – Диаграмма классов контакт-центра страховой компании

Диаграмма классов контакт-центра является основой для разработки ее логической модели данных.

Спецификация классов:

- заявка – класс подачи заявки менеджеру на выплату убытков;
- менеджер – класс пользователей контакт-центра;
- клиент – класс лиц, делающих заявку на выплату убытков.

2.4 Диаграмма последовательности подачи заявки

Диаграмма последовательности – диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл какого-либо определённого объекта и взаимодействие актёров. ИС в рамках какого-либо определённого прецедента отправка запросов и получение ответов, используется в языке UML.

Основными элементами диаграммы последовательности являются обозначения объектов (прямоугольники с названиями объектов), вертикальные линии, отображающие течение времени, прямоугольники, отражающие деятельность объекта или исполнение им определенной функции (прямоугольники на пунктирной линии) и стрелки, показывающие обмен сигналами или сообщениями между объектами.

На рисунке 6 представлена диаграмма последовательности подачи заявления о страховом случае.

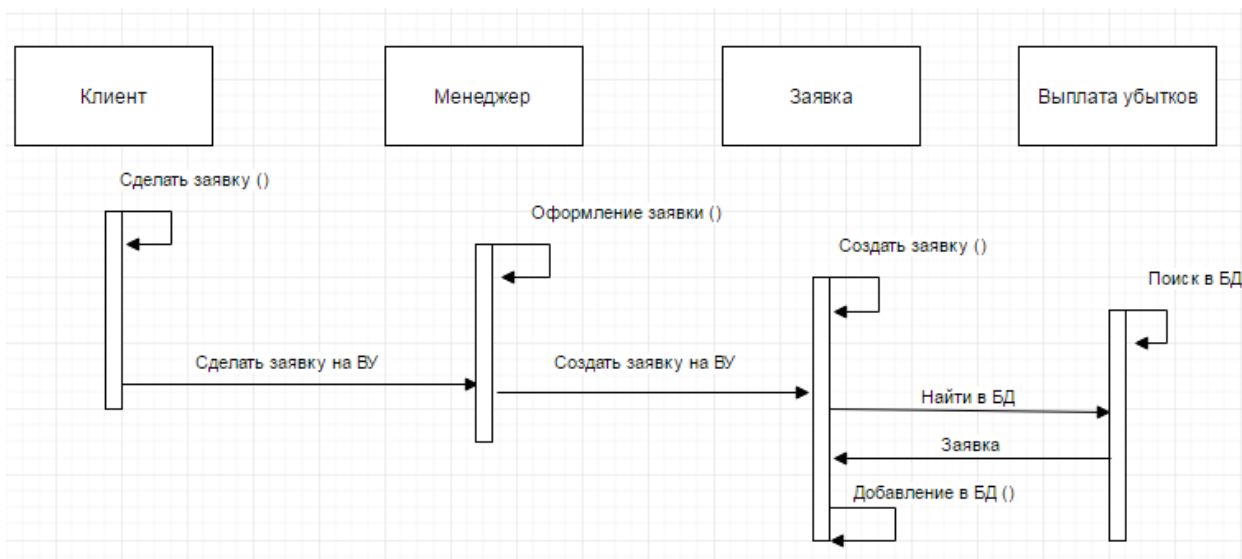


Рисунок 6 – Диаграмма последовательности подачи заявки о страховом случае

Объект клиент обращается к менеджеру, для просмотра заявки, менеджер просматривает заявку на выплату убытков.

Выводы по главе

Во второй главе было произведено проектирование АИС, включающее логическое моделирование.

Разработаны следующие диаграммы:

- вариантов использования;
- классов;
- последовательности.

Были выделены некоторые функции выполняющие системой.

Глава 3. РЕАЛИЗАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОНТАКТ-ЦЕНТРА ДЕПАРТАМЕНТА УБЫТКОВ СТРАХОВОЙ КОМПАНИИ

3.1 Модель системы разработки

3.1.1 Выбор средств реализации

Были рассмотрены самые популярные технологии разработки информационным ресурсом.

У каждого ресурса есть свои требования, в зависимости от поставленной задачи. В них входит структура, работа с информацией и характеристика технологии. Рассмотрим следующие технологии: Perl, Java и PHP.

Рассмотрим язык программирования Perl, который является интерпретируемым. Сложный в изучении, но гибкий язык.

Perl представляет собой семейство высокого уровня, общего назначения, интерпретируемых, динамических языков программирования. Языки в этом семействе включают Perl 5 и Perl 6.

Хотя Perl официально не аббревиатура, существуют различные бэкронимы в использовании, самым известным является «Практическая добыча и отчетность Язык». Perl изначально была разработана Ларри Уолл в 1987 году в качестве общего назначения Unix скриптового языка, чтобы сделать отчет обработки проще. С тех пор она претерпела множество изменений и изменений. Perl 6, которая началась как редизайн Perl 5 в 2000 году, в конце концов, превратилась в отдельный язык. Оба языка продолжают развиваться независимо друг от друга различных групп разработчиков и свободно заимствовать идеи друга от друга.

Язык Perl заимствовал функции из других языков программирования, включая C. Они обеспечивают мощные средства обработки текста без произвольных ограничений данных длины многих современных командной строки инструментов Unix, что облегчает манипуляции с текстовыми

файлами. Perl 5 получил широкую известность в конце 1990 - х годов в качестве сценариев CGI языка[5].

Следующим рассматриваемым языком программирования будет Java.

Java является универсальным языком программирования, который является одновременно, на основе классов, объектно-ориентированный, и специально разработанный, чтобы иметь как несколько зависимостей реализации, как это возможно. Java приложения обычно компилируется в байт - код , который может работать на любой виртуальной машине Java (JVM) независимо от компьютерной архитектуры. По состоянию на 2016 год, Java является одним из наиболее популярных языков программирования в использовании, особенно для веб-приложений. Java первоначально была разработана Джеймсом Гослинг в Sun Microsystems (которая впоследствии была приобретена корпорацией Oracle) и выпущен в 1995 году в качестве одного из основных компонентов компании Sun Microsystems Java платформы. Язык получает большую часть его синтаксиса от C и C++, но она имеет меньше низкоуровневых объектов, чем любой из них.

Последним рассматриваемым языком программирования был выбран PHP.

PHP – скриптовый язык, применяемый для разработки web-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических web-сайтов. PHP – является серверным языком программирования.

Популярность в области построения web-сайтов определяется наличием большого набора встроенных средств для разработки web-приложений. Основные из них:

- автоматическое извлечение POST и GET-параметров;
- взаимодействие с большим количество СУБД, в том числе MySQL и MySQLi;
- автоматизированная отправка HTTP-заголовков;

- работа с cookies и сессиями;
- обработка файлов, загружаемых на сервер.

PHP – язык с динамической, слабой и неявной типизацией. При использовании слабой типизации PHP позволяет абстрагироваться от типизации и полностью сосредоточиться на задаче. К плюсам можно отнести краткость записи и удобство использования смешанных выражений для слабой типизации. Используя неявную типизацию можно также выделить, что сокращается запись функций. Так же язык не имеет единой стандартизации в отличии, например, от C++ и C#.

Язык программирования PHP является интерпретируемым языком. Код PHP встраивается в HTML-документ, посредством PHP-вставок “<?php” и “?>”. Код вне вставок обрабатывается интерпретатором как обычный HTML. Для корректной интерпретации исходный документ должен иметь расширение «.php». Синтаксис PHP очень близко схож с синтаксисом языков C или Perl, однако сам язык изучать гораздо проще, чем два других[17].

Сам язык PHP используется для написания серверных скриптов, которые обеспечивают максимальную динамичность и интерактивность сайту. В частности:

- генерация динамических web-страниц;
- обработка данных различных форм;
- отсылка и прием cookies.

PHP является довольно простым языком для освоения новичками и поддерживается подавляющим числом хостинг-провайдеров, что без сомнения является заслугой самого языка.

В результате обзора технологий создания интернет-страниц был выбран язык программирования PHP.

3.1.2 Выбор языков разметки

Языки разметки используются везде, где требуется получения форматированного текста на основании одного лишь текста.

Во Всемирной паутине таким является HTML. Большинство веб-страниц содержат описание разметки на языке HTML. Так же язык HTML поддерживается большинством современных браузеров. Речь будет идти по пятую версию языка – HTML5.

HTML-документ, получаемый пользователем при использовании веб-браузера, может содержать различные элементы, такие как:

- блочные элементы `<h1>`, `<h2>`, `<div>`, `<form>`, `<table>`;
- ссылки `<a>`;
- изображения ``;
- разносторонние элементы формы `<input>`.

CSS – формальный язык, описывающий внешний вид документа, написанного с использованием языка разметки HTML. Преимущественно используется как средство описания, оформления внешнего вида веб-страницы, написанной, в основном, с помощью языка разметки HTML.

CSS-стили являются удобным и эффективным инструментом при верстке веб-страниц и оформлении текста, ссылок, изображения и других элементов. С использованием стилей появилась идея о том, чтобы HTML-код был освобожден от элементов оформления, а вид элементов задавался через стили. При таком подходе работа над дизайном и версткой может идти параллельно. Так же при хранении стилей в отдельном файле, он кэшируется и при повторном обращении к нему извлекается из кэша браузера. За счет кэширования и того, что стили хранятся в отдельном файле, уменьшается код веб-страницы и снижается время загрузки документа. Благодаря тому, что ссылки на файлы хранятся во всех документах удобно править стиль в одном месте, при этом оформление элементов автоматически меняется на всех страницах [20].

3.1.3 Выбор СУБД

Для разработки контакт-центра необходим сервер баз данных. Система управления базами данных (СУБД) – совокупность программных и

лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

Функции СУБД:

- управление данными во внешней памяти (на дисках);
- управление данными в оперативной памяти с использованием дискового кэша;
- журнализация изменений, резервное копирование и восстановление базы данных после сбоев;
- поддержка языков БД.

Самыми популярными являются MySQL, MS SQL Server и PostgreSQL.

PostgreSQL является объектно-реляционная система управления базами данных (ORDBMS) с акцентом на расширяемость и соответствие стандартам. В качестве сервера базы данных, ее основные функции для безопасного хранения данных и возвращают эти данные в ответ на запросы от других приложений. Он может обрабатывать рабочие нагрузки, начиная от небольших приложений одной машины до крупных интернет-приложений с большим количеством одновременно работающих пользователей; на MacOS сервер, PostgreSQL является базой данных по умолчанию, и она также доступна для Microsoft Windows и Linux (поставляется в большинстве дистрибутивов).

PostgreSQL имеет обновляемые взгляды и материализованные представления, триггеры, внешние ключи.

PostgreSQL разрабатывается глобальной группой PostgreSQL развития, различные группы многих компаний и отдельных участников. Это бесплатная СУБД с открытым исходным кодом, выпущенная в соответствии с условиями лицензии PostgreSQL.

OracleExpress – открытая реляционная система управления базами данных, разрабатываемая и поддерживаемая компанией Oracle.

OracleDatabase не имеет ограничений в применении – от простой базы данных, обслуживающей сайт или небольшую компанию, до огромных и

мощных хранилищ данных со встроенными решениями задач класса OLAP или DataMinig, хранящих любые данные: от простых таблиц до документов, видео-файлов, геоинформационных данных и т. п. Однако можно выделить недостатки, такие как: жесткие требования к операционной системе (Linux и Windows), дорогостоящая техническая поддержка Oracle, рискованное использование в коммерческих проектах.

MySQL – свободная реляционная система управления базами данных.

MySQL является самой популярной СУБД и имеет большее количество преимуществ. Ее основными преимуществами является простота в работе и инсталляции, гибкость, богатый функционал, безопасность данных, масштабируемость, скорость работы.

MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверов Денвер, XAMPP, VertrigoServ. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Большое количество таблиц поддерживается СУБД MySQL. Такие как MyISAM и InnoDB. MySQL поставляется с таблицами EXAMPLE[7].

Было решено выбрать СУБД MySQL для создания контакт-центра.

3.1.4 Выбор сервера

Для создания контакт-центра требуется сервер с поддержкой PHP и MySQL.

Web-сервер служит основной систем, работающих в среде web. Он обеспечивает доставку клиенту результатов работы скриптов и статических документов, принимает запросы клиента и перенаправляет их скриптам в нужном виде. Влияние его на работу системы (сайта) минимально, но при отсутствие надежного web-сервера система будет неработоспособна. В данной работе используется такой web-сервер, как Nginx.

Nginx - это HTTP-сервер и обратный прокси-сервер, почтовый прокси-сервер, а также TCP/UDP прокси-сервер общего назначения. Уже длительное время он обслуживает серверы многих высоконагруженных российских сайтов, таких как Яндекс, mail.ru, ВКонтакте и Рамблер[12].

Nginx позиционируется производителем как простой, быстрый и надёжный сервер, не перегруженный функциями. Применение nginx целесообразно прежде всего для статических web-сайтов и как прокси-сервера перед динамическими сайтами.

Основная функциональность HTTP-сервера:

- обслуживание статических запросов, индексных файлов, автоматическое создание списка файлов, кэш дескрипторов открытых файлов;
- акселерированное обратное проксирование с кэшированием, распределение нагрузки и отказоустойчивость;
- акселерированная поддержка FastCGI, uwsgi, SCGI и memcached серверов с кэшированием, распределение нагрузки и отказоустойчивость;
- модульность, фильтры, в том числе сжатие (gzip), byte-ranges (докачка), chunked ответы, XSLT-фильтр, SSI-фильтр, преобразование изображений; несколько подзапросов на одной странице, обрабатываемые в SSI-фильтре через прокси или FastCGI/uwsgi/SCGI, выполняются параллельно;
- Поддержка SSL и расширения TLS SNI;
- Поддержка HTTP/2 с приоритизацией на основе весов и зависимостей.

Функциональность почтового прокси-сервера:

- перенаправление пользователя на IMAP- или POP3-сервер с использованием внешнего HTTP-сервера аутентификации;
- проверка пользователя с помощью внешнего HTTP-сервера аутентификации и перенаправление соединения на внутренний SMTP-сервер.

3.1.5 Описание серверной части

В данной ВКР работе используется фреймворк laravel.

Laravel - бесплатный веб-фреймворк с открытым кодом, предназначенный для разработки с использованием архитектурной модели MVC (Model View Controller - модель-представление-контроллер). Laravel выпущен под лицензией MIT. Исходный код проекта размещается на GitHub.

Ключевые особенности, лежащие в основе архитектуры Laravel:

- пакеты (packages) - позволяют создавать и подключать модули в формате composer к приложению на Laravel. Многие дополнительные возможности уже доступны в виде таких модулей;
- eloquent ORM — реализация шаблона проектирования ActiveRecord на PHP. Позволяет строго определить отношения между объектами базы данных. Стандартный для Laravel построитель запросов Fluent поддерживается ядром Eloquent;

В данной ВКР работе используются следующие функции laravel.

Большинство роутов (маршруты, routes) приложения определены в файле `app/Http/routes.php`, который загружается сервис-провайдером `App\Providers\RouteServiceProvider`. В Laravel простейший роут состоит из URI (пути) и функции-замыкания.

HTTP Middleware (посредники) - это фильтры обработки HTTP-запроса. Так, например, в Laravel включены middlewares для проверки аутентификации пользователя. Если пользователь не залогинен, middleware перенаправляет его на страницу логина. Если же залогинен - middleware не вмешивается в прохождение запроса, передавая его дальше по цепочке middleware-посредников к собственно приложению.

Views (представления, отображения, шаблоны) обычно содержат HTML-код приложения и представляют собой удобный способ разделения

бизнес-логики и логики отображения информации. Шаблоны хранятся в папке resources/views.

Laravel предоставляет простой интерфейс к популярной библиотеке SwiftMailer. Главный файл настроек - app/config/mail.php - содержит всевозможные параметры, позволяющие менять SMTP-сервер, порт, логин, пароль, а также устанавливать глобальный адрес from для исходящих сообщений. Можно использовать любой SMTP-сервер, либо стандартную функцию PHP mail - для этого необходимо установить параметр driver в значение mail. Кроме того, доступен драйвер sendmail.

3.2 Реализация интерфейса контакт-центра департамента урегулирования убытков страховой компании

Интерфейс контакт-центра департамента урегулирования убытков страховой компании содержит следующие разделы:

- главная страница;
- заключение договоров;
- выплата убытков;
- регистрация и авторизация пользователя;
- список заявок;
- автоматизированная отправка сообщения пользователю на

электронную почту, после регистрации и заключения договора.

Разработанные элементы интерфейса полностью соответствуют изначальным требованиям к программе.

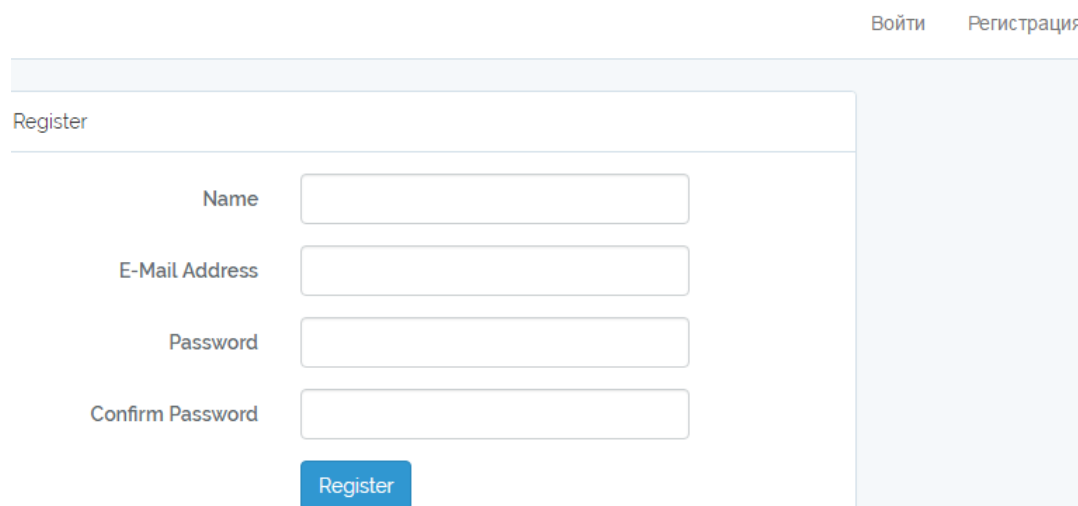
Регистрация пользователя проходит следующим образом. Пользователь заходит на страницу регистрации и вводит необходимые данные, а именно имя, адрес электронной почты, пароль и подтверждение пароля. Далее зарегистрированный пользователь авторизуется, введя адрес электронной почты и пароль. После чего пользователь попадает на главную страницу сайта.

После регистрации и авторизации пользователь может перейти непосредственно к основным функциям департамента по урегулированию убытков страховой компании, а именно к выплате убытков или заключению договоров.

На странице выплаты убытков пользователю предоставлена информация по всем программам страхования.

Перейдя на страницу заключения договоров, пользователь заполняет форму и ожидает ответа от менеджера компании.

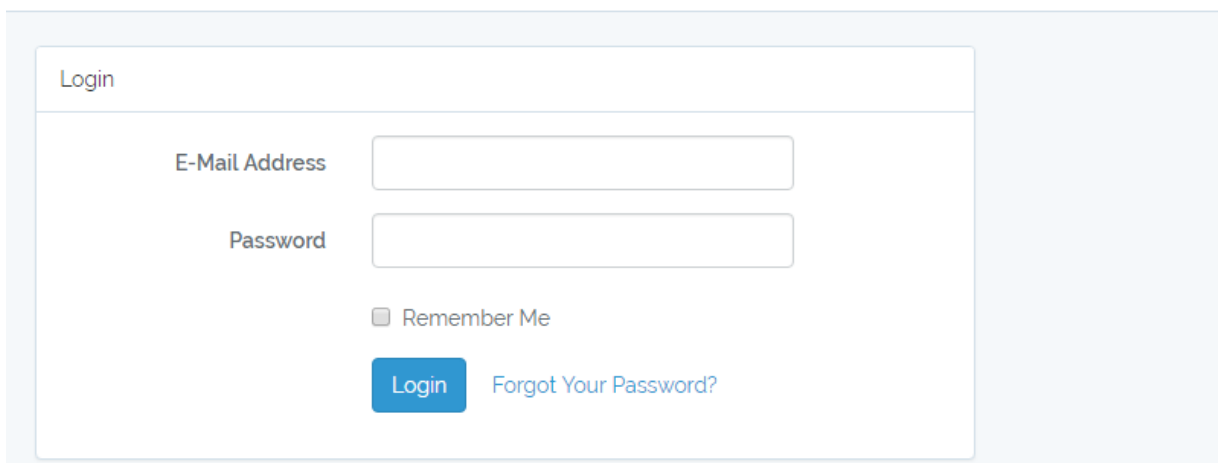
На рисунке 7 продемонстрирована форма регистрации. Исходный код контроллера регистрации представлен в приложении А.



The image shows a web interface for user registration. At the top right, there are two links: "Войти" (Login) and "Регистрация" (Registration). Below these is a light blue header bar. The main content area is titled "Register" and contains a form with four input fields: "Name", "E-Mail Address", "Password", and "Confirm Password". Each field is represented by a white rectangular box with a thin border. Below the "Confirm Password" field is a blue button with the text "Register" in white. The form is set against a light gray background.

Рисунок 7 – Форма регистрации пользователя

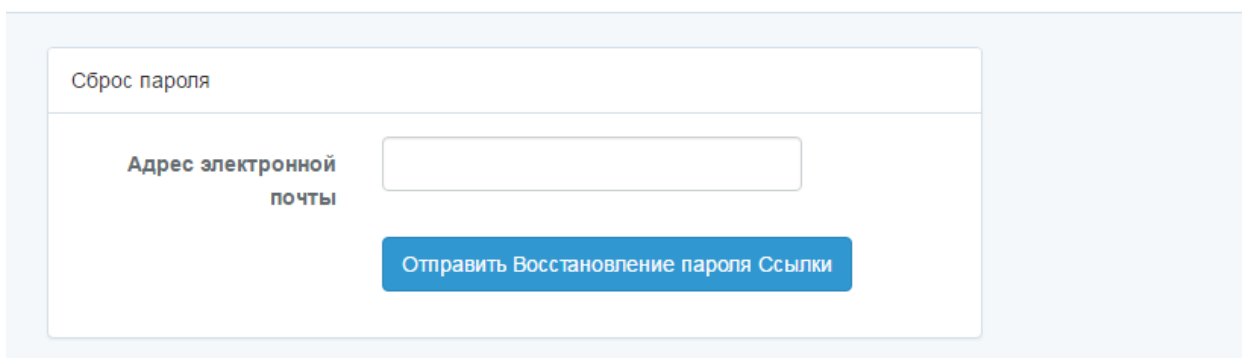
Так же на рисунке 8 показана форма авторизации. Исходный код контроллера авторизации представлен в приложении Б, код контроллера пароля (приложение В).



The screenshot shows a login form titled "Login". It contains two input fields: "E-Mail Address" and "Password". Below the password field is a checkbox labeled "Remember Me". At the bottom of the form, there is a blue "Login" button and a link labeled "Forgot Your Password?".

Рисунок 8 – Форма авторизации пользователя

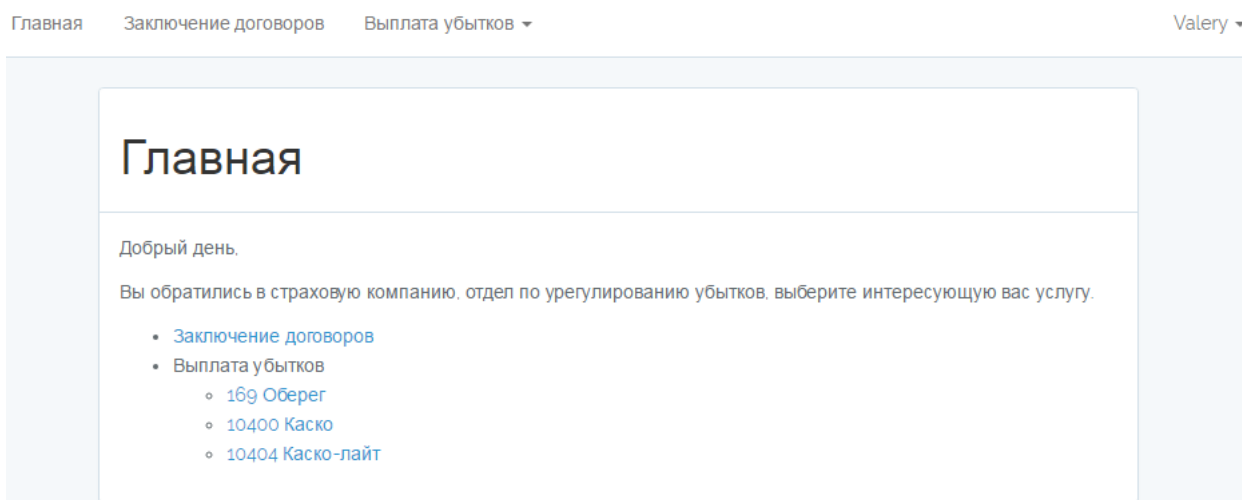
На рисунке 9 показана форма восстановления пароля, исходный код контроллера сброса пароля представлен в приложении Г.



The screenshot shows a password reset form titled "Сброс пароля". It contains one input field labeled "Адрес электронной почты". Below the input field is a blue button labeled "Отправить Восстановление пароля Ссылки".

Рисунок 9 – Форма сброса пароля для получения сообщения

После авторизации пользователь автоматически попадает на главную страницу сайта (рисунок 10).



The screenshot shows the main page of the website. At the top, there is a navigation menu with links: "Главная", "Заключение договоров", "Выплата убытков", and a user profile "Valery". The main content area has a heading "Главная" and a greeting "Добрый день." Below the greeting, there is a message: "Вы обратились в страховую компанию, отдел по урегулированию убытков, выберите интересующую вас услугу." A list of services is provided:

- Заключение договоров
- Выплата убытков
 - 169 Оберег
 - 10400 Каско
 - 10404 Каско-лайт

Рисунок 10 – Главная страница сайта пользователя

С главной страницы пользователь может перейти во вкладки: «заключение договоров», «169 Оберег», «10400 Каско», «10404 Каско-Лайт», либо вернуться на главную страницу, воспользовавшись вкладкой «главная».

На рисунке 11 показана форма договора на выплату убытков, на которой пользователю необходимо заполнить все поля, после чего менеджер свяжется с пользователем.

Главная Заключение договоров Выплата убытков Valery

Договор на выплату убытков

Пожалуйста заполните форму ниже и мы свяжемся с вами.

Добавление заявки

Фамилия * Пожалуйста укажите вашу фамилию

Имя * Пожалуйста, укажите ваше имя

Отчество * Пожалуйста, укажите ваше отчество

Телефон * Пожалуйста, укажите ваш телефон

Email * Пожалуйста, укажите ваш email адрес

Отправить заявку

Рисунок 11 – Форма договора для выплаты убытков

После нажатия на кнопку «отправить заявку» высвечивается сообщение, продемонстрированное на рисунке 12.

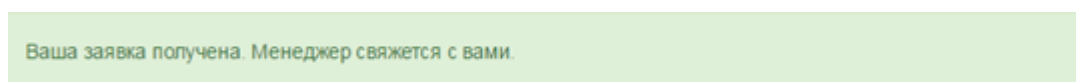


Рисунок 12 – Сообщение после успешной отправки заявки

Нажав на главной странице кнопку «выплата убытков», выплывет окно, как это показано на рисунке 13.

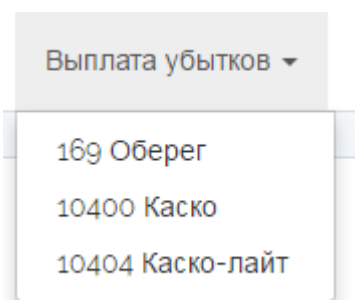


Рисунок 13 – Результат кнопки «выплата убытков»

Далее продемонстрированы результаты страниц «169 оберег», «10400 каско» и «10404 каско-лайт».

На рисунок 14 продемонстрирован результат страницы программы страхования «169 Оберег».

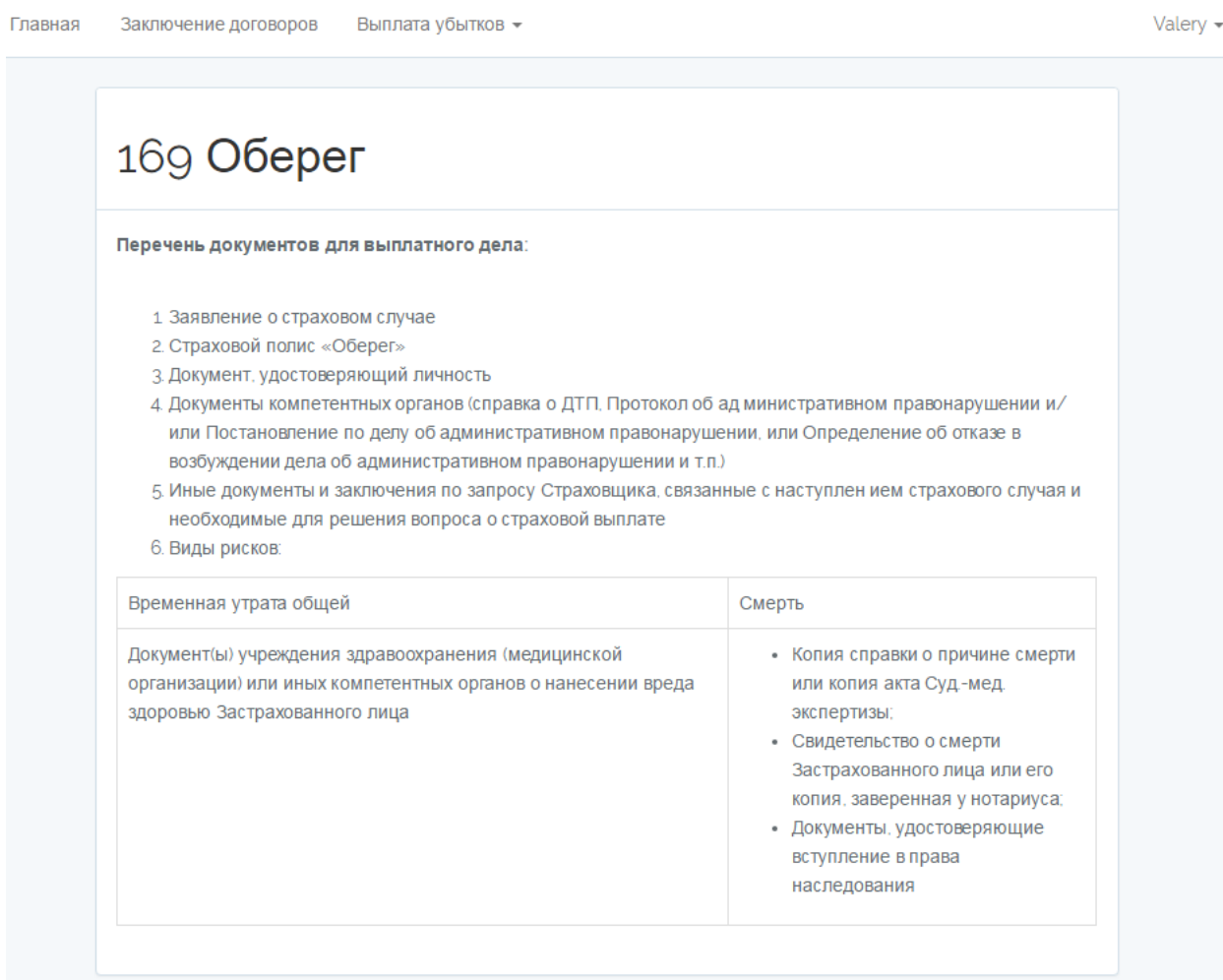


Рисунок 14 – Реализация страницы «169 Оберег»

На рисунках 15,16 показана страница программы страхования «10400 Каско».

Главная Заключение договоров Выплата убытков ▾ Valery ▾

10400 Каско

Перечень документов для выплатного дела:

1. Заявление о страховом случае.
2. Копии документов, подтверждающих право собственности на поврежденное имущество (Свидетельство о регистрации ТС или ПТС).
3. Документы, подтверждающие статус водителя транспортного средства как лица, управляющего данным транспортным средством на законном основании.
4. Полис (договор страхования).
5. Статус (лицо) страхователя

Физическое лицо

Собственник	Представитель страхователя	Наследник
Паспорт РФ	<ul style="list-style-type: none">• Паспорт РФ• Нотариально заверенная доверенность	<ul style="list-style-type: none">• Паспорт РФ• Свидетельство о праве на наследство

Юридическое лицо

- Путевой лист.
- Доверенность на право представления интересов в страховой компании;
- Распорядительное письмо на выплату по реквизитам.

6. Документы, составленные на месте ДТП:

Рисунок 15 – Реализация страницы «10400 Каско»

С вызовом компетентных органов

Ущерб	Угон	Стихийное бедствие	Пожар
<ul style="list-style-type: none"> • Справка о ДТП • Протокол об админ. правонарушении и/или постановление по делу об админ. правонарушении и/или определение об отказе в возбуждении дела (админ. материал) 	<ul style="list-style-type: none"> • Справка или иные акты ОВД, содержащие сведения по обст. события, о дате и времени обращения Стр-ля в ОВД о виновных лицах, если такие установлены; • Постановление о возбуждении и Постановлен. о приостановлении (прекращении) уголовного дела; • По требования Страховщика - справку ОВД с перечнем похищенного имущества; • Договор справка или иной документ из десп. центра - при установке на ТС поисковой системы; • Оригинал ключей; • Оригинал ПТС. 	<ul style="list-style-type: none"> • Справка из органов гидрометеослужбы; • Справка из полиции, РЭУ, ТСЖ с указанием перечня повреждений. 	<ul style="list-style-type: none"> • Заключение (справка) органа гос. противопож. служ с указанием очага и причин возгорания; • Оригинал ПТС.

7. Реквизиты расчетного счета потерпевшего и банка, в котором он открыт, либо СТО.

Рисунок 16 – Продолжение страницы «10400 Каско»

На рисунках 17, 18 показана реализация страницы программы страхования «10404».

Главная Заключение договоров Выплата убытков ▾ Valery ▾

10404 Каско-лайт

Перечень документов для выплатного дела:

1. Заявление о страховом случае.
2. Копии документов, подтверждающих право собственности на поврежденное имущество (Свидетельство о регистрации ТС или ПТС).
3. Водительское удостоверение лица, находящегося за рулем ТС, в момент ДТП.
4. Страховой полис.
5. Статус (лицо) страхователя:

Физическое лицо

Собственник	Доверенное лицо
Паспорт РФ	Паспорт РФ Нотариально заверенная доверенность
Реквизиты расчетного счета получателя страхового возмещения и банка, в котором он открыт	

Рисунок 17 – Реализация страницы «10404 Каско-лайт»

Юридическое лицо

- Путевой лист
- Доверенность на право представления интересов в страховой компании;
- Распорядительное письмо на выплату по реквизитам.

6. Документы, составленные на месте ДТП:

С ГИБДД	Без ГИБДД
<ul style="list-style-type: none"> • Извещение о ДТП • Справка о ДТП • Протокол об админ. правонарушении и/или постановление по делу об админ. правонарушении и/или определение об отказе в возбуждении дела (админ. материал) 	Извещение о ДТП, заполненное и подписанное обоими участниками

Рисунок 18 – Продолжение страницы «10404 Каско-лайт»

Далее будет продемонстрирован интерфейс администратора. Для того чтобы попасть на страницу, необходимо пройти по адресу <http://astrovolga.tigmig.ru/sadmin>. Откуда администратор сможет управлять разделами сайта и отслеживать заявки.

На рисунке 19 продемонстрирована форма авторизации администратора.

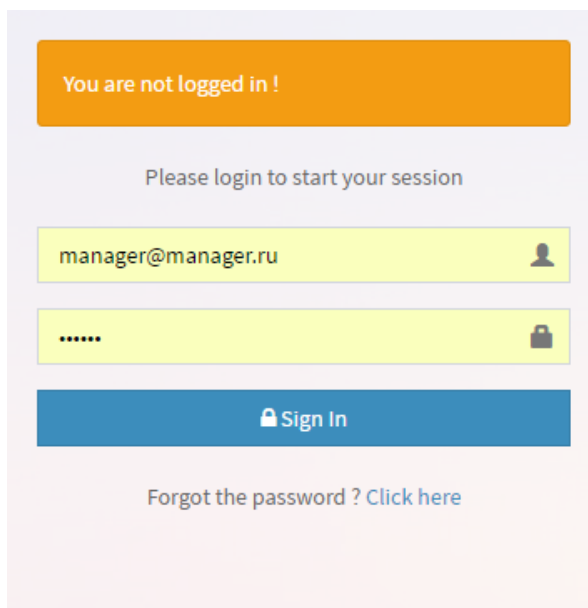


Рисунок 19 – Форма авторизации пользователя

После входа, администратор попадает на страницу менеджера. На рисунке 20 продемонстрирована страница управления сайтом, где администратор может просмотреть, редактировать, удалить или добавить страницу.

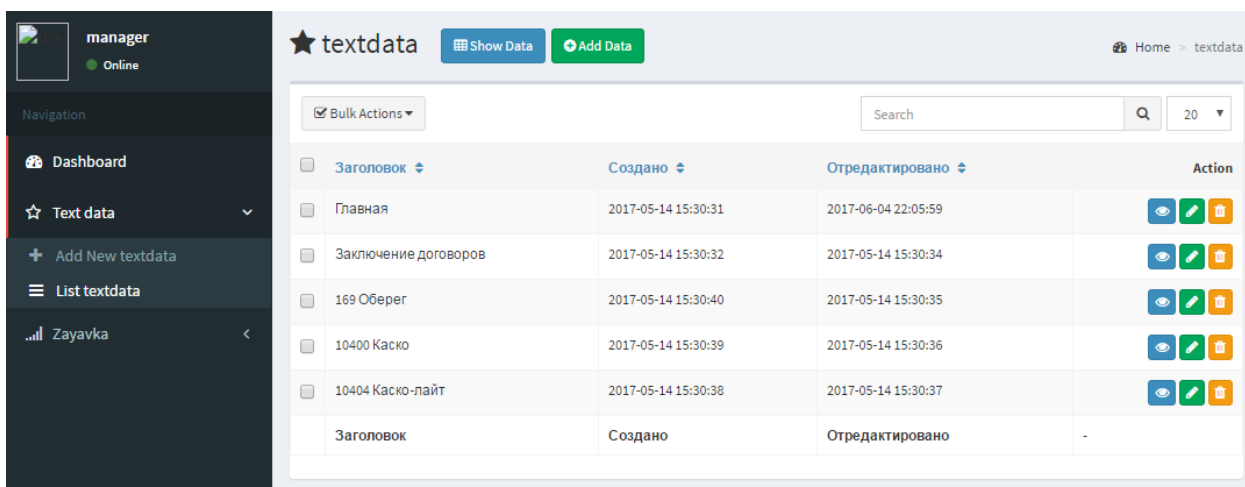


Рисунок 20 – Страница управления сайтом администратора

На рисунке 21 продемонстрирована страница заявок, где администратор может просмотреть детальную информацию, отредактировать или удалить заявку.

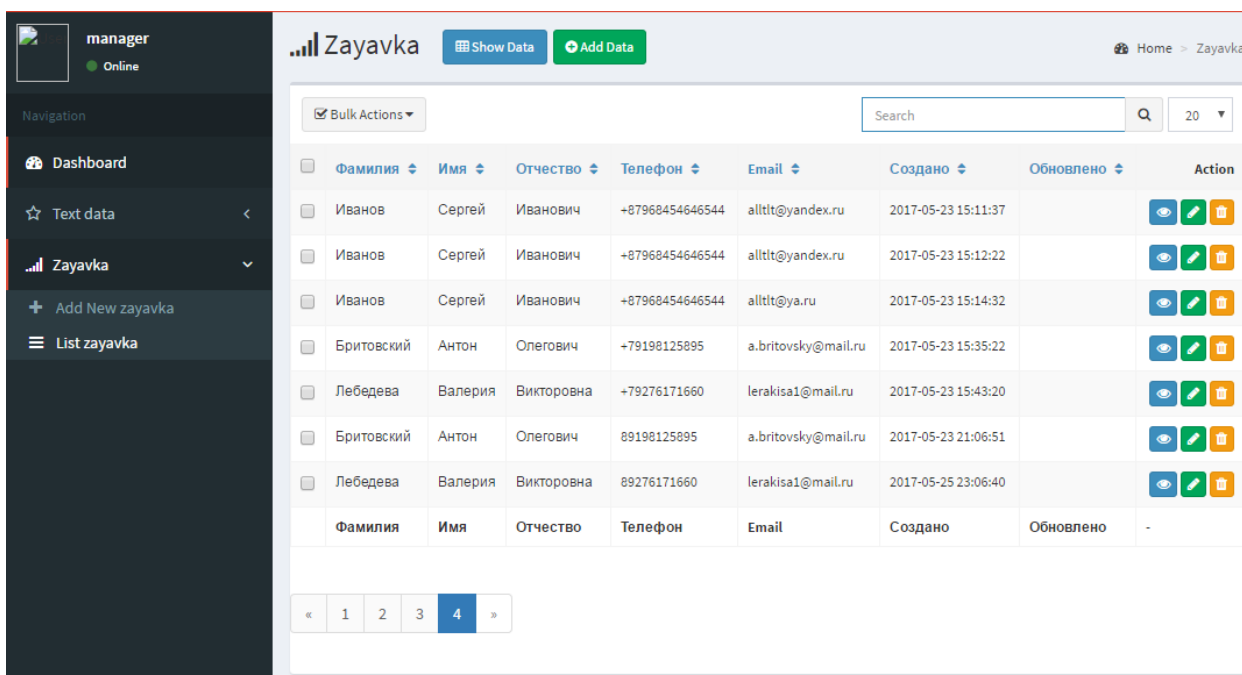


Рисунок 21 – Страница отслеживания заявок администратора

В ходе выполнения регистрации была предпринята попытка оставить одно из полей не заполненным. При этом появилось сообщение, о том, что необходимо заполнить поле, как это показано на рисунке 22, что соответствует требованиям к программе.

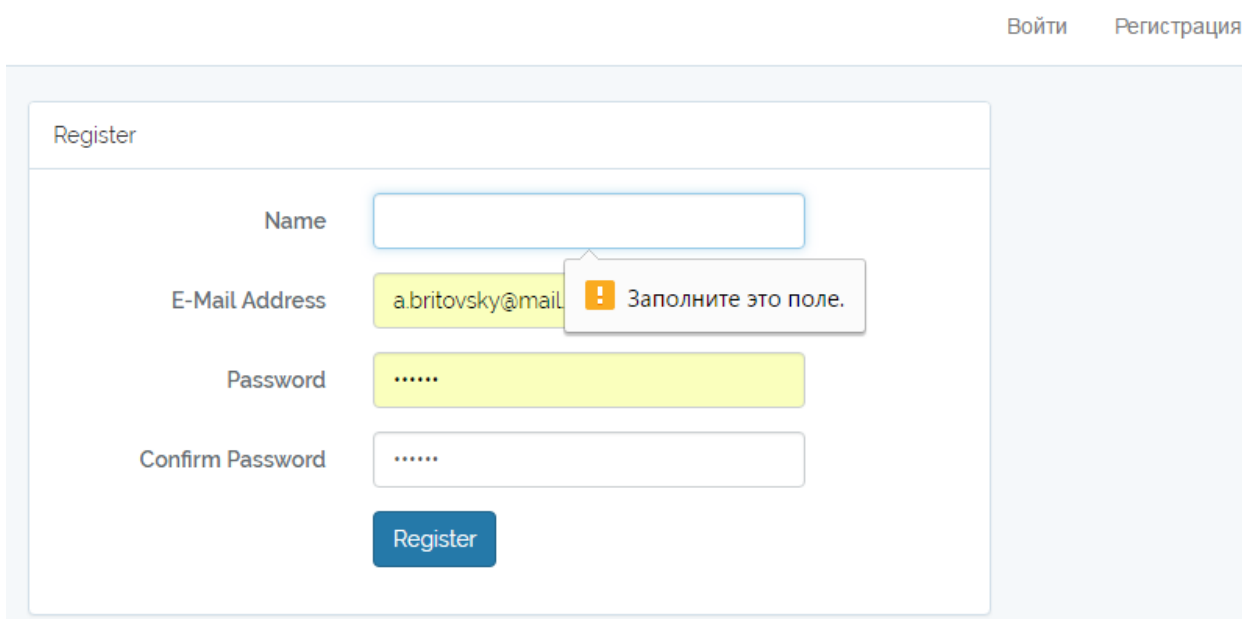


Рисунок 22 – Ошибка при не заполненном поле

После регистрации, пользователю, на указанный им электронный адрес, придет сообщение, как это показано на рисунке 23.

Спасибо за регистрацию

Сейчас вы можете зайти на сайт и выполнить следующие действия:

- [Главная](#)
- [Заключение договоров](#)
- [Выплата убытков](#)
 - [169 Оберег](#)
 - [10400 Каско](#)
 - [10404 Каско-лайт](#)

Нажмите, чтобы [Ответить](#), [Ответить всем](#) или [Переслать](#)

Рисунок 23 – Сообщение отправленное пользователю после регистрации

Так же была попытка зарегистрироваться с уже ранее созданной почтой, при этом появилось сообщение об ошибке, как это показано на рисунке 24, что соответствует требованиям к программе.

При регистрации была попытка ввести подтверждающий пароль отличный от выше введенного, на что появилось сообщение об ошибке (рисунок 25), которое так же соответствует требованиям к программе.

Войти Регистрация

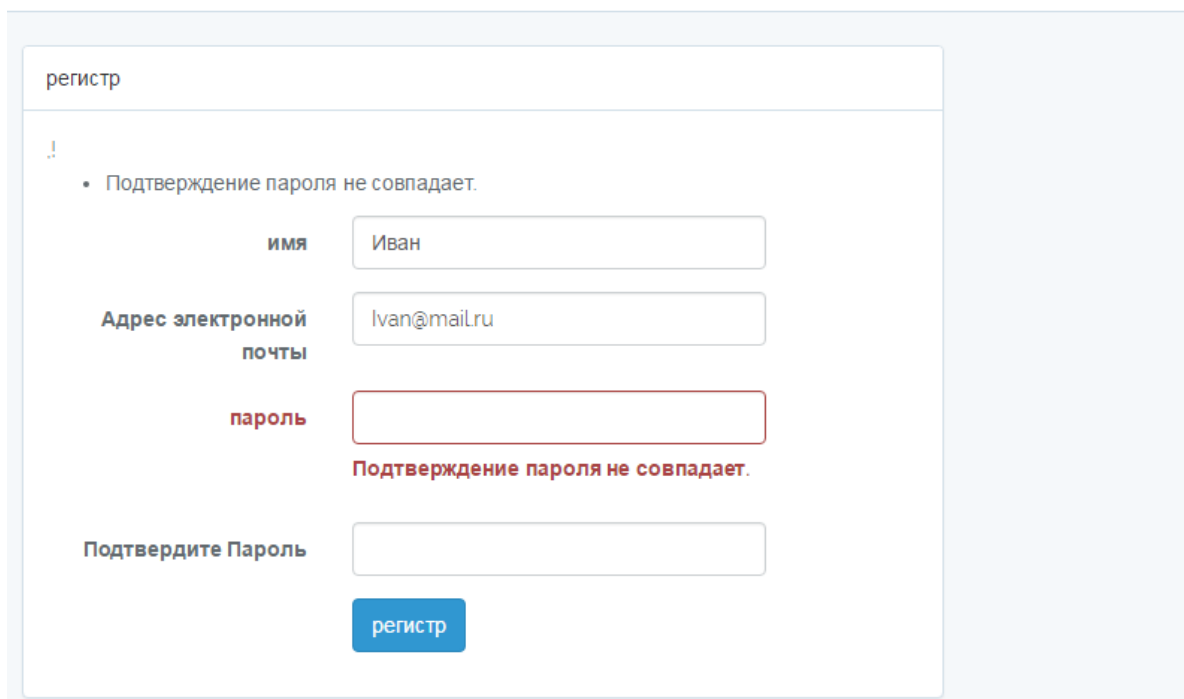
регистр

⚠

- Электронная почта уже принята.

имя	<input type="text" value="Антон"/>
Адрес электронной почты	<input type="text" value="a.britovsky@mail.ru"/> Электронная почта уже принята.
пароль	<input type="password"/>
Подтвердите Пароль	<input type="password"/>

Рисунок 24 – Ошибка при вводе уже существующей почты



регистр

!

- Подтверждение пароля не совпадает.

имя

Адрес электронной почты

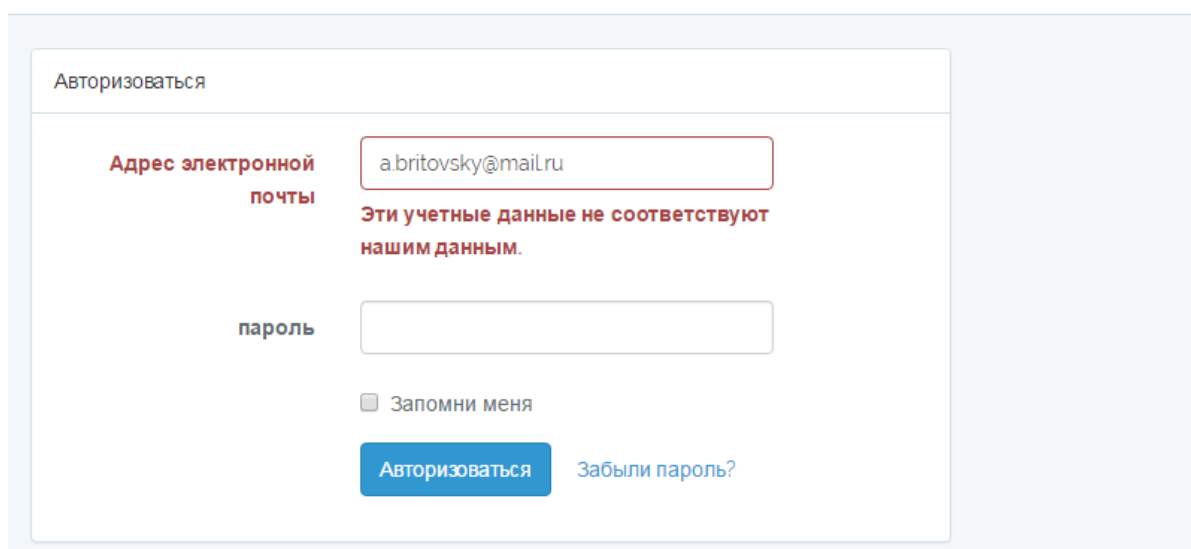
пароль

Подтвердите Пароль

Подтверждение пароля не совпадает.

Рисунок 25 – Ошибка подтверждения пароля

Далее при попытке авторизации была предпринята попытка ввода неверной почты или пароля, на что программа выдает сообщение об ошибке, продемонстрированное на рисунке 26, что соответствует требованиям к программе.



Авторизоваться

Адрес электронной почты

Эти учетные данные не соответствуют нашим данным.

пароль

Запомни меня

[Забыли пароль?](#)

Рисунок 26 – Ошибка ввода неверного адреса электронной почты или пароля

В ходе формы заполнения заявки, была предпринята попытка оставить не заполненным одно из обязательных полей. При этом появилось сообщение об ошибке (рисунок 27), что соответствует требованиям к программе.

Главная Заключение договоров Выплата убытков ▾ Valery ▾

Договор на выплату убытков


Пожалуйста заполните форму ниже и мы свяжемся с вами.

Добавление заявки

Фамилия * Васильев

Имя * Василий

Отчество * Пожалуйста, укажите ваше отчество

Телефон * +78625984554  Заполните это поле.

Email * vasya.vasilev1997@mail.ru

Рисунок 27 – Ошибка при пропуске поля для обязательного заполнения

После того как пользователь оформил и отправил заявку. На почту менеджеру придет сообщение о новой заявке (рисунок 28). Пользователю придет сообщение, продемонстрированное на рисунке 29.

Новая заявка с сайта



менеджер diesel@tigmig.ru
Вам ▾

23 мая в 23:06

На сайте новая заявка: Имя: Антон Отчество: Олегович Фамилия:
Бритовский Телефон: 89198125895 Email: a.britovsky@mail.ru



Нажмите здесь, чтобы [Ответить](#) или [Переслать](#)

Рисунок 28 – Сообщение о новой заявке на почте менеджера

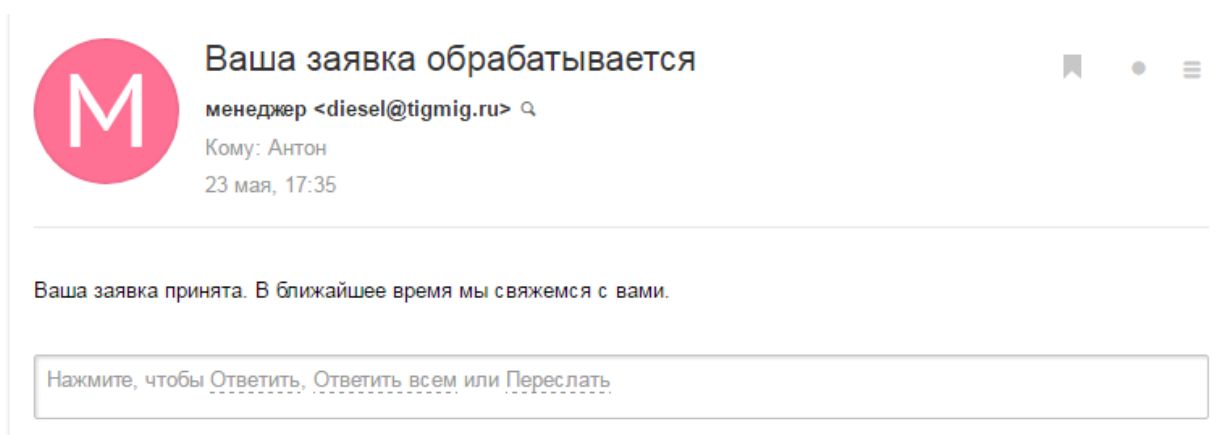


Рисунок 29 – Сообщение, подтверждающее подачу заявки
На рисунке 30 продемонстрирована форма сброса пароля.

[Войти](#) [Регистрация](#)

Сброс пароля

Адрес электронной почты	<input type="text"/>
пароль	<input type="password"/>
Подтвердите Пароль	<input type="password"/>

Рисунок 30 – Форма для сброса пароля

В результате разработки приложения была протестирована работа всех кнопок и функций на наличие вывода сообщений об ошибках, описанных в разработке интерфейса пользователя, а описание выводимых ошибок соответствует установленным требованиям к пользовательскому интерфейсу.

3.3 Структура программного обеспечения и требования к аппаратному обеспечению

Диаграмма компонентов – это схема, взаимодействия компонентов друг с другом с образованием более крупных компонентов или систем программного обеспечения. Они используются для иллюстрации структуры произвольно сложных систем.

Компонентом является то, что требуется для выполнения функции стереотипа. Примеры стереотипов компонентов включают в себя исполняемые файлы, документы, таблицы базы данных, файлы и файлы библиотек.

Компоненты соединены друг с другом с помощью соединителя сборки для подключения требуемого интерфейса одного компонента с помощью прилагаемого интерфейса другого компонента.

Соединитель в сборе представляет собой соединитель между двумя компонентами, который определяет, что один компонент предоставляет услуги, которые требует другой компонент.

На рисунке 31 продемонстрирована диаграмма компонентов контакт-центра страховой компании.

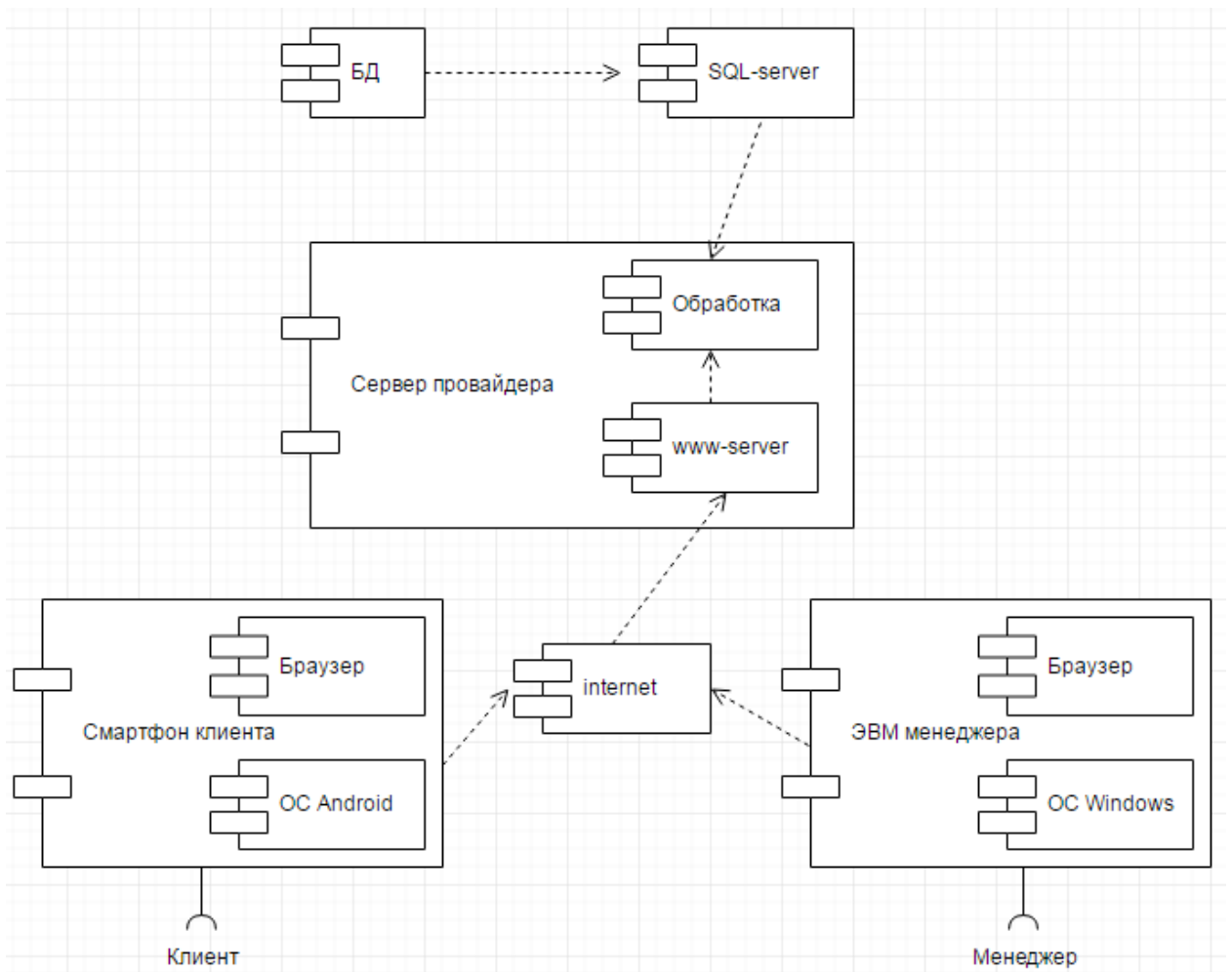


Рисунок 31 – Диаграмма компонентов контакт-центра

Рекомендуемые системные требования для смартфона клиента:

- веб-браузеры: Google Chrome, Firefox, Internet Explorer (8 и более поздняя версия) или Opera.
- ОС Android 4.2 или более поздняя версия;
- процессор с частотой от 1.2 ГГц;
- ОЗУ минимум 512 Ббайт;
- подключение высокоскоростного интернет-соединения.

Рекомендуемые системные требования для ЭВМ менеджера:

- веб-браузеры: Chrome (44 версия), Firefox (40 версия), Internet Explorer Edge (13 версия), Opera (31 версия), Safari (9 версия) или более поздние версии.
- ОС Windows XP/7;

- процессор с частотой от 2,3ГГц;
- монитор с разрешением 1440x900;
- ОЗУ от 2 Гб;
- клавиатура, мышь.

Конфигурация сервера: BL460c Gen8 10Gb FLB CTO Blade, процессор: 2 x E5-2660 (20M Cache, 2.20 GHz, 8 cores, 16 threads), Память: 4 x HP 8GB 2Rx4 PC3L-10600R.

Системное ОС: Microsoft Windows Server 2008 R2 Enterprise Edition (10CAL).

Требования по установке: Диски размечаются с максимально допустимым объемом и форматируются в NTFS. Сервер должен быть подключен по 1Gb/s Ethernet через сетевой коммутатор.

3.4 Тестирование программного обеспечения

Для того чтобы проверить стабильности работы разработанного контакт-центра, необходимо проведение нагрузочного тестирования.

Для проверки был выбран ApacheJmeter. Программа разработанная с целью проведения нагрузочных тестов, основная возможность которого является создание большого количества запросов.

Интересна возможность создания большого количества запросов с помощью нескольких компьютеров при управлении этим процессом с одного из них. Архитектура, поддерживающая плагины сторонних разработчиков, позволяет дополнять инструмент новыми функциями.

Н рисунках 32 – 35 показан результат тестирования контакт-центра в виде диаграммы, таблицы и дерева, а так же показан отчет.

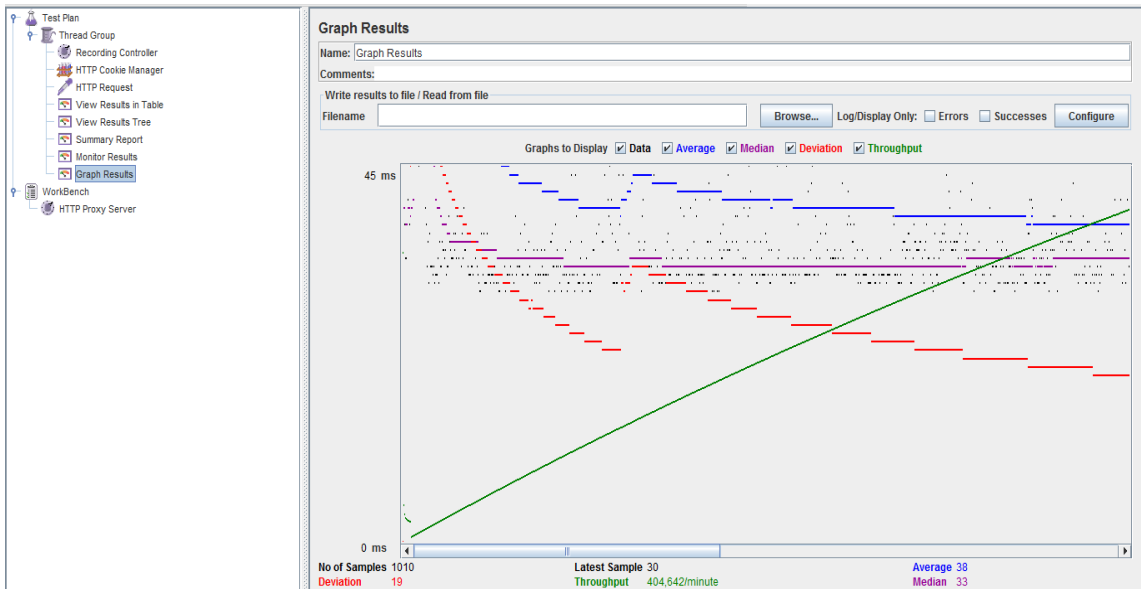


Рисунок 32 – Результат нагрузки в виде диаграммы

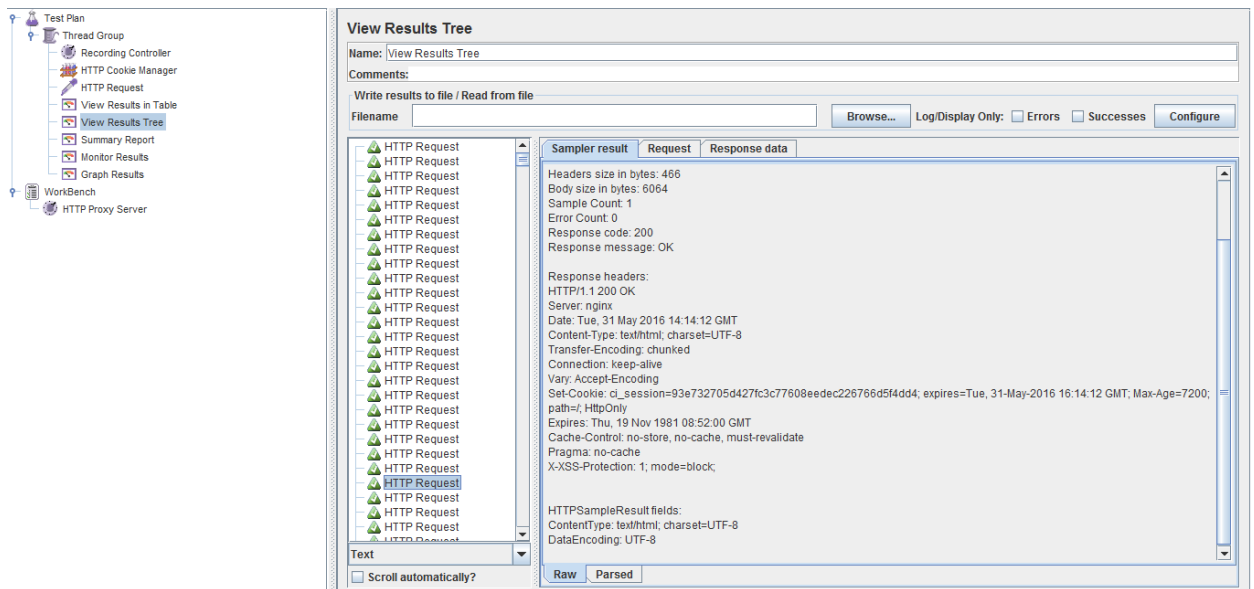


Рисунок 33 – Результат нагрузки в виде дерева

View Results in Table

Name: View Results in Table
 Comments:

Write results to file / Read from file
 Filename: Browse... Log/Display Only: Errors Successes Configure

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes
1	18:11:24.284	Thread Group 1-1	HTTP Request	192	▲	6530
2	18:11:27.236	Thread Group 1-2	HTTP Request	40	▲	6530
3	18:11:30.289	Thread Group 1-3	HTTP Request	38	▲	6530
4	18:11:33.287	Thread Group 1-4	HTTP Request	34	▲	6530
5	18:11:36.287	Thread Group 1-5	HTTP Request	34	▲	6530
6	18:11:39.288	Thread Group 1-6	HTTP Request	54	▲	6530
7	18:11:42.284	Thread Group 1-7	HTTP Request	41	▲	6530
8	18:11:45.286	Thread Group 1-8	HTTP Request	51	▲	6530
9	18:11:48.286	Thread Group 1-9	HTTP Request	38	▲	6530
10	18:11:51.285	Thread Group 1-10	HTTP Request	39	▲	6530
11	18:13:23.519	Thread Group 1-1	HTTP Request	180	▲	6530
12	18:13:23.549	Thread Group 1-2	HTTP Request	186	▲	6530
13	18:13:23.578	Thread Group 1-3	HTTP Request	156	▲	6530
14	18:13:23.610	Thread Group 1-4	HTTP Request	157	▲	6530
15	18:13:23.642	Thread Group 1-5	HTTP Request	134	▲	6530
16	18:13:23.723	Thread Group 1-7	HTTP Request	107	▲	6530
17	18:13:23.749	Thread Group 1-8	HTTP Request	97	▲	6530
18	18:13:23.744	Thread Group 1-6	HTTP Request	102	▲	6530
19	18:13:23.786	Thread Group 1-9	HTTP Request	79	▲	6530
20	18:13:23.830	Thread Group 1-11	HTTP Request	45	▲	6530
21	18:13:23.795	Thread Group 1-10	HTTP Request	98	▲	6530
22	18:13:23.855	Thread Group 1-12	HTTP Request	56	▲	6530
23	18:13:23.887	Thread Group 1-13	HTTP Request	63	▲	6530
24	18:13:23.925	Thread Group 1-14	HTTP Request	64	▲	6530
25	18:13:23.959	Thread Group 1-15	HTTP Request	73	▲	6530
26	18:13:23.973	Thread Group 1-16	HTTP Request	84	▲	6530
27	18:13:24.007	Thread Group 1-17	HTTP Request	56	▲	6530

Scroll automatically? Child samples? No of Samples 1010 Latest Sample 30 Average 38 Deviation 19

Рисунок 34 – Результат нагрузки в виде таблицы

Summary Report

Name: Summary Report
 Comments:

Write results to file / Read from file
 Filename: Browse... Log/Display Only: Errors Successes Configure

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg Bytes
HTTP Request	1010	38	30	380	19,66	0,00%	6,7/sec	43,01	6530,0
TOTAL	1010	38	30	380	19,66	0,00%	6,7/sec	43,01	6530,0

Include group name in label? Save Table Header

Рисунок 35 – Отчет результатов нагрузки тестирования

Как видно из отчета нагрузочного тестирования было сформировано 1010 запросов, ошибок тестирование не выявило. Следует вывод, что разработанный контакт-центр работает стабильно.

Выводы по главе

В третьей главе были выбраны средства проектирования программного обеспечения: язык программирования PHP, СУБД MySQL, языки разметки HTML и CSS, сервер Nginx. А так же было разработано, реализовано и протестировано программное обеспечение контакт-центра.

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы была подробно исследована предметная область, на основе которой были определены требования к функциональным характеристикам работы и были выделены основные задачи, которые предстояло реализовать.

При создании контакт-центра департаменту урегулирования убытков страховой компании были проанализированы уже существующие контакт-центры. Из анализа были выделены достоинства и недостатки отделов по урегулированию убытков и на основе этого анализа были сформулированы требования к контакт-центру.

Выбран комплекс технических и программных средств реализации. Система реализована на языке PHP. В качестве системы базы данных использована СУБД MySQL. К достоинствам разработки можно отнести простоту в эксплуатации и понятный интерфейс.

В ходе выполнения выпускной квалификационной работы был создан контакт-центр департаменту урегулирования убытков страховой компании, который полностью соответствует требованиям по разработке программного обеспечения.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

Нормативно-правовые акты

1. ГОСТ 34.320-96. Информационная технология. Система стандартов по базам данных. Концепции и терминология для концептуальной схемы и информационной базы.

Научная и методическая литература

2. Бабич А.В. Введение в UML / Бабич А.В. - М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 198 с.

3. Буренин С.Н. Web-программирование и базы данных: учебный практикум / Буренин С.Н. - М.: Московский гуманитарный университет, 2014.-120 с.

4. Гольдштейн Б.С. Call-центры и компьютерная телефония // Б.С. Гольдштейн, В. А. Фрейнкман. БХВ. - СПб, 2015. - 372 с.

5. Золотов С. Ю. Проектирование информационных систем : учеб. пособие / С. Ю. Золотов ; Томский гос. ун-т систем управления и радиоэлектроники. - Томск : Эль Контент, 2013. - 86 с.

6. Информационные аналитические системы : учебник / Т. В. Алексеева ; под ред. В. В. Дика. - Москва : Синергия, 2013. - 379 с.

7. Карпова И. П. Базы данных : курс лекций и материалы для практ. занятий : учеб. пособие для студентов техн. фак. / И. П. Карпова. – СПб. : Питер, 2013. - 240 с.

8. Мкртычев С.В. Моделирование системы электронного документооборота урегулирования убытков страховой компании / С.В. Мкртычев, А.В. Очеповский // Вектор науки Тольяттинского государственного университета. - 2014. - №1(27). – С. 53-57.

9. Хорстманн К. С. Java SE 8. Вводный курс = Java SE 8 for the Really Impatient / К. С. Хорстманн. - М.: «Вильямс», 2014. - 208 с.

10. Крахоткина Е.В. Методы и средства проектирования информационных систем и технологий : учебное пособие / Крахоткина Е.В. - Ставрополь: Северо-Кавказский федеральный университет, 2015. -152 с.

11. Трутнев Д. Р. Архитектуры информационных систем. Основы проектирования: учеб. пособие. – СПб.: НИУ ИТМО, 2012. – 66 с.

Электронные ресурсы

12. Веб-сервер Nginx. - URL: <https://nginx.ru/ru> (дата обращения 01.06.2017).

13. Средства автоматизации моделирования UML. - URL: <http://citforum.ru/database/case/index.shtml> (дата обращения 01.06.2017).

14. Страховая компания «РЕСО-Гарантия». - URL: <http://www.reso.ru/About> (дата обращения 01.06.2017).

15. Тестирование программного обеспечения. -URL: <http://www.protesting.ru> (дата обращения 01.06.2017).

Литература на иностранном языке

16. Denuit, M. Actuarial Modelling of Claim Counts / M. Denuit, X. Maréchal, S. Pitrebois, J.-F. Walhin. – London: Wiley, 2007. -356 p.

17. Gilmore W.J. Beginnig PHP and MySQL. -3rd Edition, 2011.

18. Kirilov, R. Effectiveness of the Computer Information Systems in Insurance / R. Kirilov // Economic Alternatives. - 2008. – Iss. 2. -P. 146-152.

19. Models and Analysis in Distributed Systems / ed. by S. Haddad, F. Kordon, L. Pautet, L. Petrucci. – London-Hoboken: Wiley-ISTE, 2011. -368 p.

20. Nixon R. Learning PHP, MySQL, JavaScript, CSS & HTML5 - 3rd Edition, 2014.

ПРИЛОЖЕНИЕ А

Исходный код контроллера регистрации

```
<?php
namespace App\Http\Controllers\Auth;
use App\User;
use Validator;
use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\RegistersUsers;
use App\Http\Controllers\HomeController;
use DB;
use Mail;
class RegisterController extends Controller
{
    /**
     *
     * -----
     * | Register Controller
     * -----
     *
     * | This controller handles the registration of new users as well as their
     * | validation and creation. By default this controller uses a trait to
     * | provide this functionality without requiring any additional code.
     *
     */
    use RegistersUsers;

    /**
     *
     * * Where to redirect users after login / registration.
     *
     * @var string
     */
}
```

```

protected $redirectTo = '/';
/**
 * Create a new controller instance.
 *
 * @return void
 */
public function __construct()
{
    $this->middleware('guest');
}
/**
 * Get a validator for an incoming registration request.
 *
 * @param array $data
 * @return \Illuminate\Contracts\Validation\Validator
 */
protected function validator(array $data)
{
    return Validator::make($data, [
        'name' => 'required|max:255',
        'email' => 'required|email|max:255|unique:users',
        'password' => 'required|min:6|confirmed',
    ]);
}
/**
 * Create a new user instance after a valid registration.
 *
 * @param array $data
 * @return User

```

```

*/
protected function create(array $data)
{
    $titles = DB::table('textdata')
        ->select('id','title')
        ->orderBy('id', 'asc')
        ->get();

    $this->userSubject = 'Информация зарегистрировавшегося';
    $this->userEmail = $data['email'];
    $this->userName = $data['name'];
    $this->managerEmail = 'britovsky.anton@yandex.ru';
    $this->managerName = 'АНТОН';

    Mail::send('emails.register', array('titles' => $titles),
function($message){
        $message->from($this->managerEmail, $this-
>managerName);

        $message->to($this->userEmail, $this->userName)-
>subject($this->userSubject);

    });

    return User::create([
        'name' => $data['name'],
        'email' => $data['email'],
        'password' => bcrypt($data['password']),
    ]);
}
}

```

ПРИЛОЖЕНИЕ Б

Реализация формы заявления о страховом случае

```
@extends('layouts.app')
@section('content')
<div class="container">
  <div class="row">
    <div class="col-md-10 col-md-offset-1 col-lg-10 col-lg-offset-1 col-sm-12">
      <div class="panel panel-default">
        <div class="panel-heading"><h1>{{ $values->title }}</h1></div>
        <div class="panel-body">
          {!!$values->textdata!!}
          @if ($values->id ===1)
            <ul>
              @if (count($titles) > 0)
                @foreach ($titles as $title)
                  @if ( $title->id === 2 )
                    <li><a href="/home/{ {
$title->id } }">{{ $title->title }}</a></li>
                    <li>Выплатаубытков</li>
                  @elseif ( $title->id > 2 )
                    <li><a href="/home/{ {
$title->id } }">{{ $title->title }}</a></li>
                  @endif
                @endforeach
              </ul>
            </li>
          @endif
        </div>
      </div>
    </div>
  </div>
</div>
```

```

        @elseif ($values->id ===2)

        </br>
@if (\Session::has('success'))
    <div class="alert alert-success">
        @if (is_array(\Session::get('success')))
            @foreach (\Session::get('success') as $textSuccess)
                {!! $textSuccess !!}
            @endforeach
        @else
            {!! \Session::get('success') !!}
        @endif
    </div>
@endif

<div class="panel panel-default">
    <div class="panel-heading">
        <strong><i class='fa fa-signal'></i> Добавление заявки</strong>
    </div>
    <div class="panel-body" style="padding:20px 0px 0px 0px">
        <form class='form-horizontal' method='post' id="form"
enctype="multipart/form-data" action='/home/2'>
            <input type="hidden" name="_token" value="{{ csrf_token() }}">
                <div class="box-body">
                    <div class='form-group header-group-0 '
id='form-group-familia' style="">
                        <label class='control-label col-sm-
2'>Фамилия <span class='text-danger' title='This field is
required'>*</span></label>
                            <div class="col-sm-10">

```

```

        <input type='text' title="Фамилия"
required placeholder='Пожалуйста укажите вашу фамилию' maxlength=50
class='form-control' name="familia" id="familia" value=""/>
        <div class="text-danger"></div>
        <p class='help-block'></p>
    </div>
</div>
<div class='form-group header-group-0 '
id='form-group-imy' style="">
        <label class='control-label col-sm-
2'>Имя <span class='text-danger' title="This field is required">*</span></label>
        <div class="col-sm-10">
            <input type='text' title="Имя" required
placeholder='Пожалуйста, укажите ваше имя' maxlength=50 class='form-
control' name="imy" id="imy" value=""/>
            <div class="text-danger"></div>
            <p class='help-block'></p>
        </div>
    </div>
<div class='form-group header-group-0 '
id='form-group-otchestvo' style="">
        <label class='control-label col-sm-
2'>Отчество <span class='text-danger' title="This field is
required">*</span></label>
        <div class="col-sm-10">
            <input type='text' title="Отчество"
required placeholder='Пожалуйста, укажите ваше отчество' maxlength=50
class='form-control' name="otchestvo" id="otchestvo" value=""/>

```



```

        <div class="text-danger"></div>
        <p class='help-block'></p>
    </div>
</div>
<div class='form-group header-group-0 '
id='form-group-telefon' style="">
        <label class='control-label col-sm-
2'>Телефон <span class='text-danger' title='This field is
required'>*</span></label>
        <div class="col-sm-10">
            <input type='text' title="Телефон"
required placeholder='Пожалуйста, укажите ваш телефон' maxlength=15
class='form-control' name="telefon" id="telefon" value=""/>
            <div class="text-danger"></div>
            <p class='help-block'></p>
        </div>
    </div>
<div class='form-group header-group-0 '
id='form-group-email' style="">
        <label class='control-label col-sm-
2'>Email <span class='text-danger' title='This field is required'>*</span></label>
        <div class="col-sm-10">
            <div class="input-group">
                <span class="input-group-addon"><i class="fa fa-
envelope"></i></span>
                <input type='email' title="Email" required
placeholder='Пожалуйста, укажите ваш email адрес' maxlength=255
class='form-control' name="email" id="email" value=""/>
            </div>
            <div class="text-danger"></div>

```

