

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

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование)

01.03.02. Прикладная математика и информатика
(код и наименование направления подготовки, специальности)

Системное программирование и компьютерные технологии
(направленность (профиль)/ специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)**

на тему: «Разработка математической модели для управления цифровым порталом по
продаже ХААs-продуктов»

Студент

Н.А. Зорин

(И.О. Фамилия)

(личная подпись)

Руководитель

М.А. Тренина

(ученая степень, звание, И.О. Фамилия)

Консультант

К.А. Селиверстова

(ученая степень, звание, И.О. Фамилия)

Тольятти 2020

Аннотация

Тема выпускной квалификационной работы: «Разработка математической модели для управление цифровым порталом по продаже ХААS-продуктов».

Целью ВКР является разработка математической модели для оптимального управления цифровым порталом телекоммуникационной компании по продаже ХааS-продуктов.

Структура работы состоит из: введения, трёх глав, заключение и списком литературы.

Во введении описывается актуальность темы, формируются цель и задачи, которые необходимо выполнить.

В первой главе была выбрана технология концептуального моделирования. Дано определение WEB-портала, что такое ХааS-продукт и необходимость в реализации портала для телекоммуникационной компании.

Во второй главе описываются основные характеристики WEB-портала, разрабатываемого для телекоммуникационной компании с помощью структурного и объектно-ориентированного подходов.

В третьей главе представлена технология разработки и принципы работы WEB-портала. Произведено тестирование функциональных требований, предоставлены скриншоты выполненной работы и обоснована эффективность использования WEB-портала.

В заключении представлены результаты и выводы о выполненной работе.

Результатом выпускной квалификационной работы является внедрённая математическая модель на цифровой портал по продаже ХааS продуктов.

В работе представлено 11 таблиц 18 рисунков, список использованной литературы содержит 20 источников. Общий объем выпускной квалификационной работы составляет 54 страницы.

Abstract

The title of the graduation project is "Development of a mathematical model for the management of a digital portal for the sale of XAAS products."

The aim is to develop a mathematical model for managing the digital portal of a telecommunication company selling XaaS products.

The structure of the work is presented by introduction, three chapters, conclusion, list of references and appendix.

The graduation project describes in details the need to automate the registration of the vendor and putting up for sale its software by creating a mathematical model. The vendor's software consists of XaaS products and will be laid out through the created WEB portal for sale to customers.

The graduation project consists of an explanatory note on 55pages, introduction, including 18 figures, 12 tables, the list of 20 references including 5 foreign sources.

The subject of the project is a mathematical model for selling XaaS products on the WEB portal. The object of the project is the WEB portal of a telecommunication company selling XaaS products.

In the first chapter, conceptual modeling technology is selected. A definition of the mathematical model for WEB portal is given and the need to implement it on the portal for a telecommunication company is described.

The second chapter describes the main characteristics of a WEB portal developed for a telecommunication company using object-oriented approaches.

The third chapter presents the development technology and principles of the WEB portal. The functional requirements were tested, screenshots of the work performed were provided, and the effectiveness of using the WEB portal was substantiated.

The result of the graduation project work will be a developed product - a digital portal for vendors selling XaaS products and this result showed that mathematical model had a positive impact on increasing of selling products on WEB portal.

Оглавление

Введение.....	6
Глава 1. Функциональное моделирование телекоммуникационной компании	8
1.1 Техничко-экономическая характеристика компании	8
1.2 Определение ХaaS портала и его необходимость реализации.....	9
1.3 Концептуальное моделирование бизнес-процесса	11
1.3.1 Выбор технологии концептуального моделирования автоматизированногоцифрового портала для управления заказами.....	11
1.3.2 Анализ модели бизнес-процесса управления регистрации вендоров телекоммуникационной компании «КАК ЕСТЬ»	14
1.3.3 Обоснование необходимости автоматизированного варианта решения и формирование требований к новой технологии	18
1.4 Постановка задачи на разработку математической модели для ХaaS- портала	20
Глава 2. Проектирование математической модели	23
2.1 Проектирование математической модели для WEB-портала с помощью UseCaseдиаграммы.....	23
2.2 Описание основных UseCaseWEB-портала	24
2.3 Проектирование архитектуры данных.....	28
2.4 Описание работы алгоритма SHA-1 для хеширования паролей.....	32
Глава 3. Практическая реализация математической модели для ХaaS-портала	38
3.1 Выбор технологии разработки программного обеспечения	38
3.2 Описание основного принципа работы ХaaS-портала для телекоммуникационной компании.....	39

3.3 Тестирование математической модели для ХaaS-портала телекоммуникационной компании.....	47
3.4 Обоснование эффективности использования разработанной математической модели для ХaaS-портала телекоммуникационной компании.....	49
Заключение	52
Список используемой литературы	54

Введение

В настоящее время конкуренция в предоставлении услуг очень высокая, в связи с этим, чтобы компания была высокорентабельной, требуются особые отношения с клиентами.

На сегодня известно, что оптимизации производства и повышения качества продукции уже недостаточно, особенно в сфере услуг, так как отношения с клиентами играет такую же большую роль, как и качество продукции. Из этого следует, что для развития компании крайне важна продуктивная работа с клиентами, чтобы удержать и собрать новых клиентов.

Чтобы увеличить поток клиентов создают информационные WEB-порталы по продаже продуктов компании. Использование WEB-портала позволяет эффективно продвигать продукцию клиентам. WEB-портал — это модель, которая позволяет клиенту самостоятельно получить информацию и самостоятельно разобраться в продуктах компании и их свойствах. Внедрение WEB-портала целесообразно для автоматизации некоторых бизнес-процессов компании, а именно продвижения и продаж.

Актуальность выпускной квалификационной работы (ВКР) обусловлена необходимостью автоматизации регистрации вендора и выставление на продажу его программного обеспечения путем создания математической модели. ПО вендора состоит из XaaS-продуктов и будет выкладываться через созданный WEB-портал для продажи клиентам.

Предметом исследования является математическая модель для управления WEB-порталом, на котором происходит оформление заказов XaaS-продуктов.

Объект исследования: WEB-портал телекоммуникационной компании по продаже XaaS-продуктов.

Целью ВКР является создание математической модели для оптимального управления цифровым порталом по продаже XaaS-продуктов у телекоммуникационной компании.

Для достижения поставленной цели необходимо решить следующие задачи:

- Изучить предметную область;
- Разработать концептуальную модель WEB-портала;
- Разработать математическую модель для WEB-портала;
- Реализовать математическую модель на WEB-портале;
- Провести тестирование математической модели на WEB-портале по продаже ХaaS-продуктов;
- Обосновать эффективность использования математической модели на WEB-портале по продаже ХaaS-продуктов.

Выпускная квалификационная работа состоит из введения, трех глав, заключения, списка использованных источников.

В первой главе была выбрана технология концептуального моделирования. Дано определение WEB-портала, что такое ХaaS-продукт и необходимость в реализации портала для телекоммуникационной компании.

Во второй главе описываются основные характеристики WEB-портала, разрабатываемого для телекоммуникационной компании с помощью структурного и объектно-ориентированного подходов. Были выполнены проектирование логической модели WEB-портала при помощи UseCase диаграммы, а также разработка архитектуры данных.

В третьей главе представлены принципы работы WEB-портала и был сделан выбор технологии разработки программного обеспечения. Произведено тестирование функциональных требований, предоставлены скриншоты выполненной работы и обоснована эффективность использования WEB-портала.

Итогом выпускной квалификационной работы является разработка и управление цифровым порталом по продаже ХaaS-продуктов для телекоммуникационной компании.

Глава 1. Функциональное моделирование телекоммуникационной компании

1.1 Техничко-экономическая характеристика компании

Компания, о которой пойдет речь во всей выпускной работе, не будет оглашена, так как это коммерческая тайна. Это российская телекоммуникационная компания, одна из ведущих на рынке телеком-услуг.

Оператор предоставляет услуги широкополосного доступа (ШПД) в интернет и кабельного телевидения в 17 городах России — в сумме 4 млн домохозяйств, занимая 39% рынка кабельного ТВ и 32% рынка ШПД. Компания занимает четвертое место в России как по числу абонентов ШПД, так и по количеству абонентов платного ТВ.

У компании есть свой сайт. На рисунке 1.1 представлен скриншот.

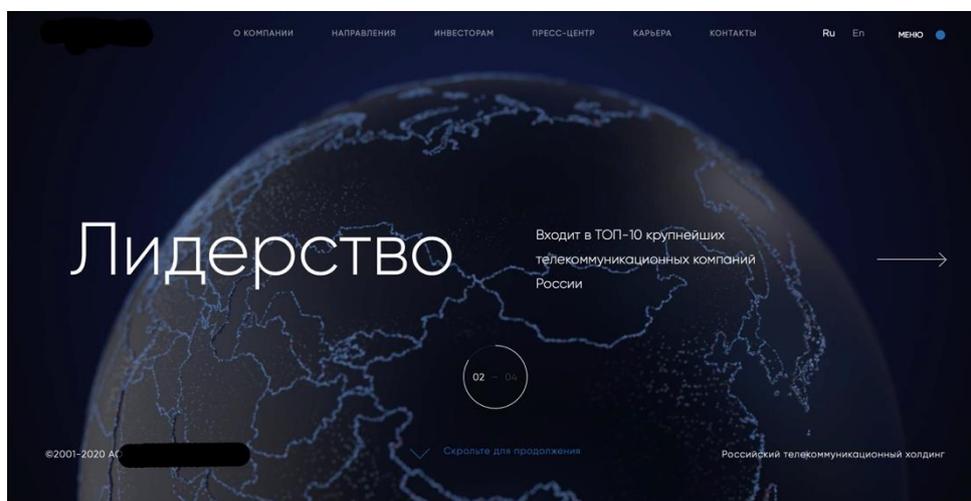


Рисунок 1.1 – Скриншот сайта компании

1.2 Определение ХaaS портала и его необходимость реализации

Для телекоммуникационной компании ХaaS-портал необходим, так как ИТ-услуги в современном мире переходят в облачные технологии и позволяют людям платить за сервис только тогда, когда они используют его. Для обозначения такой модели отношений существует понятие ХaaS или «всё-как-сервис» (Anything-as-a-service). Под это определение попадают все услуги, которые производятся через интернет и которые возможно предоставлять с применением облачных технологий. Так как существует разное понимание сервисов, поэтому “Х” в начале слова “ХaaS” обозначает неизвестную переменную, вместо которой подставляются разные понятия. К примеру, если сервис предоставляется как платформа для разработки, то название будет “РaaS”, если как инфраструктура – “IaaS”, если как софт – “SaaS”.

Общая сводка предоставления ХaaS сервисов, которая представлена в таблице 1.1, где колокация – размещение провайдером оборудования клиента на своей территории, а хостинг – услуга по предоставлению ресурсов для размещения информации на сервере, постоянно имеющем доступ к сети.

Таблица 1.1 – Разница между ХaaS-продуктами.

Колокация	Хостинг	IaaS	РaaS	SaaS
Данные	Данные	Данные	Данные	Данные
Приложения	Приложения	Приложения	Приложения	Приложения
Базы данных				
Операционная система				
Виртуализация	Виртуализация	Виртуализация	Виртуализация	Виртуализация
Физический сервер				
Сети и хранилища				

Дата-центр	Дата-центр	Дата-центр	Дата-центр	Дата-центр
------------	------------	------------	------------	------------

В таблице синим цветом показано то, что организует клиент, а оранжевым – то, что организует провайдер.

Для полного понимания, что является Хаас-продуктами разберем основные примеры. Главное преимущество Хаас-продуктов – это быстрота развертывания и экономия ресурсов заказчика.

В начале возьмём IaaS – инфраструктура-как-сервис. Для заказчика провайдер предоставляет хранение и доступ к данным. Сервис имеет нужную инфраструктуру, такое как серверное и облачное оборудование, инженеров, которые настраивают и обслуживают его, и помещение для размещения дата-центров. Все это выгодно для заказчика, так как при организации собственной инфраструктуры уйдет много времени и средств. Именно так клиент может снизить расходы, арендовав площадку провайдера и выбрав подходящий для себя вариант: установить собственный сервер в дата-центре, арендовать его или же использовать вычислительные мощности провайдера.

Теперь рассмотрим PaaS – платформа-как-сервис. Во время производства ПО необходима платформа, то есть среда разработки, базы данных, чтобы создать его в то время, как для организации всего этого необходимо время и деньги. Провайдер предоставляет эту возможность со своей стороны. Таким образом, клиент экономит, используя облачные среды разработки с поддержкой всех необходимых ему сервисов. Примерами являются Microsoft Azure, средство для развертывания приложений Docker, базы данных от Oracle. Преимуществом PaaS является быстрота запуска приложения, обработка коротких команд, а также сбор статистики о работе ПО клиента, необходимой для анализа и дальнейшего улучшения.

Следующий продукт SaaS – софт-как-сервис. Существуют все известные нам программы для отправки сообщений, редактирования сообщения и печати текста, но также существуют и программы для работы

непосредственно внутри компании, такие как CRM, ERP и прочие. Если раньше необходимо было покупать программы и устанавливать их, то сейчас есть возможность использовать их через браузер. В этом и заключается смысл SaaS-продуктов. Примерами являются Office365, сервисы Google, amoCRM и прочие. В сравнении с классическим видом приложений, устанавливаемых на компьютер, здесь нет необходимости покупать полную версию и платить сразу много за программу. Также большой плюс, что такие сервисы можно использовать с различных устройств.

Существуют и другие виды XaaS-продуктов, такие как DBaaS – база данных-как-сервис, SaaS – коммуникация-как-сервис и даже MaaS – кибератаки-как-сервис.

Для клиентов существуют много преимуществ XaaS-продуктов, именно поэтому сейчас это быстрорастущий рынок. Существует только один недостаток: при эксплуатации облачного сервиса в течении нескольких лет стоимость может оказаться дороже, чем покупка программы навсегда.

1.3 Концептуальное моделирование бизнес-процесса

1.3.1 Выбор технологии концептуального моделирования автоматизированного цифрового портала для управления заказами

Концептуальная модель разрабатывается для выбранной компании интегратором системы, беря за основу методологии реинжиниринга и управления бизнес-процессами компании. Построение и улучшение компании в методологии реинжиниринга происходит в несколько этапов:

- анализ модели «КАК ЕСТЬ»;

- разработка математической модели ТО-ВЕ («КАК ДОЛЖНО БЫТЬ»);
- разработка плана перехода из состояния («КАК ЕСТЬ» в состояние «КАК ДОЛЖНО БЫТЬ»);
- внедрение изменений и построение организации «КАК ДОЛЖНО БЫТЬ».

В современном мире для предоставления высокой эффективности цифрового портала рекомендуется использовать различные бизнес-кейсы для успешного выполнения процессного анализа. Также будут задействованы инструментальные средства с использованием структурного подхода, основанные на принципе функциональной декомпозиции – отбор единиц предметной области, их взаимосвязей и семантических отношений. Выберем технологию концептуального моделирования, проведя сравнительный анализ трех наиболее известных нотаций концептуального моделирования информационной системы: BPMN, ARIS, UML.

BPMN (BusinessProcessModelandNotation) – нотация и модель бизнес-процессов – это язык моделирования бизнес-процессов, который является промежуточным звеном между формализацией/визуализацией и воплощением бизнес-процесса.

UML (UnifiedModelingLanguage) – унифицированный язык моделирования. Это система обозначений, которую можно применять для объектно-ориентированного анализа и проектирования.

ARIS (ArchitectureofIntegratedInformationSystems) – методология и тиражируемый программный продукт для моделирования бизнес-процессов организаций с помощью комплекса средств, которые формализуют поступающую информацию, анализируют и оптимизируют внутреннюю деятельность предприятия и представляют её в виде графических моделей.

Проведем анализ данных нотаций, который представлен ниже, в таблице 1.2.

Таблица 1.2 – Сравнительный анализ методологий.

Определение	BPMN	ARIS	UML
Легкость познания	Легко	Очень сложно	Сложно
Система хранения данных модели	Модели хранятся в файлах	Объектная база данных	Модели хранятся в файлах
Возможность декомпозиции	Неограниченная	Неограниченная	Неограниченная
Подход к проектированию	Функциональный	Процессный	Объектно-ориентированный

По данной таблице видно, что наиболее подходящей и легкой в использовании является нотация BPMN, которая удобна в создании моделей и имеет необходимый функциональный подход к проектированию. UML и ARIS имеют довольно сложную панель управления, а также значительно труднее для понимания и изучения.

Нотация BPMN содержит условные обозначения, которые позволяют отобразить бизнес-процессы в качестве диаграмм бизнес-процессов. BPMN подходит как для различных технических специалистов, так и на бизнес-аналитиков и менеджеров. Нотация BPMN создана для того, чтобы связывать дизайн бизнес-процесса и его реализацию. Для отображения комплексных семантических конструкций язык использует пять главных категорий интуитивно понятных элементов:

- элементы потока (события, процессы и шлюзы);
- данные (объекты данных и базы данных);

- соединяющие элементы (потoki управления, потоки сообщений и ассоциации);
- зоны ответственности (пулы и дорожки);
- артефакты (сноски).

Проанализировав и выбрав технологию концептуального моделирования, перейдем к анализу модели бизнес-процесса «КАК ЕСТЬ».

1.3.2 Анализ модели бизнес-процесса управления регистрацией вендоров телекоммуникационной компании «КАК ЕСТЬ»

В современном мире часто меняются тенденции, под которые телекоммуникационным компаниям постоянно надо подстраиваться. Допустим, если раньше популярность СМС была высокой, то сейчас она спала из-за появления Интернета и перехода общения в онлайн-мессенджеры. Это лишь один из примеров, когда спрос на интернет-сервисы увеличивается.

В связи с этим телекоммуникационной компанией было решено организовать цифровой портал для продажи ХааS-продуктов. Основная задача – привлекать вендоров с необходимыми для людей продуктами путем организации удобной площадки для реализации их продуктов.

Раньше, без портала, менеджер собственноручно регистрировал вендоров, которые обращались к нему путем телефонных звонков. Этот процесс был очень времязатратным, в следствие чего количество обработанных заявок было недостаточным и компания теряла клиентов, а также деньги. Для того, чтобы улучшить это слабое место бизнес-процесса, необходимо разработать диаграмму «КАК ЕСТЬ» и найти исходную точку для изменения.

Модель бизнес-процесса «КАК ЕСТЬ» описывает принципы и механизмы неавтоматизированного функционирования бизнес-процесса.

Подобная информация является основой для комплексного системного анализа процессов, поиска проблем и путей их преодоления.

На рисунке 1.3 представлена модель «КАК ЕСТЬ» с точки зрения менеджера по работе с вендорами, составленная при помощи BPMN.

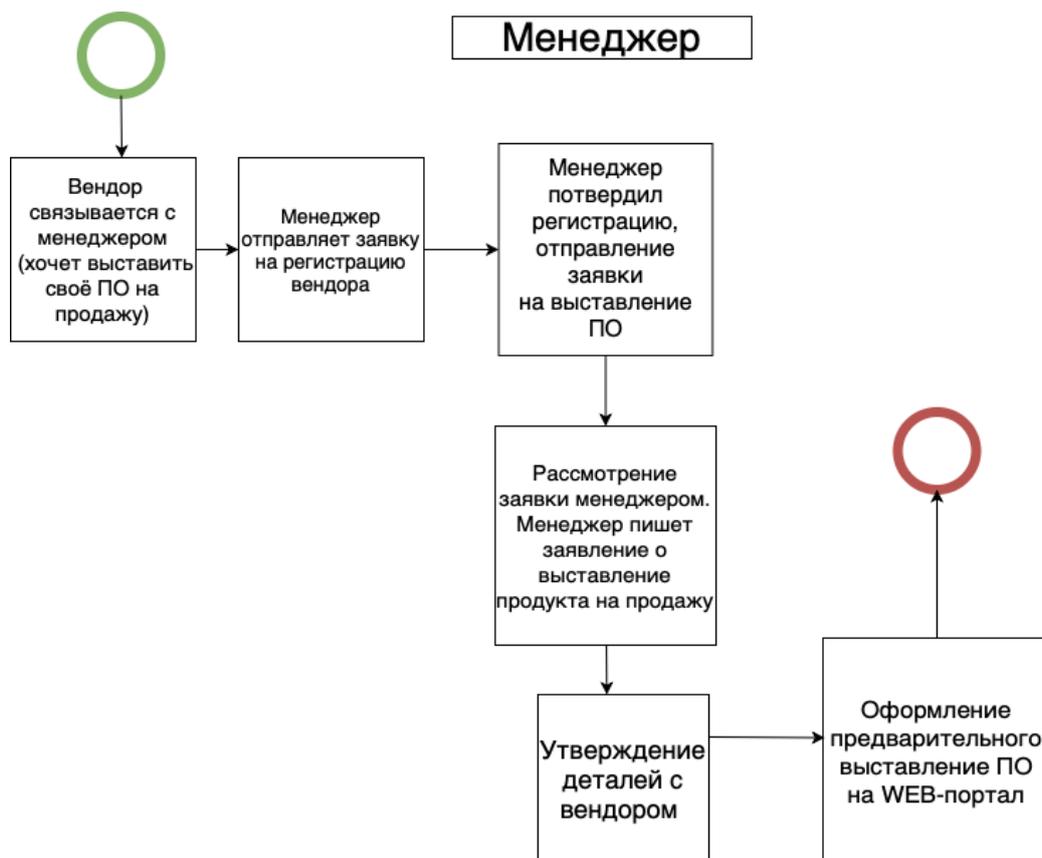


Рисунок 1.2 – Контекстная диаграмма «КАК ЕСТЬ».

Первым делом вендор связывается по телефону с менеджером по поводу выставления своего ПО. Далее менеджер отправляет форму «заявка на регистрацию». После того, как вендор её заполнил, он отправляет её менеджеру. Таким образом, теперь у него появляется возможность подать следующую заявку – на продажу своего ПО, при условии что менеджер одобрил его регистрацию. Форму для заявки «продажа ПО» так же отправляет менеджер. Вендор её снова заполняет, как и заявку на регистрацию, и пересылает менеджеру. После идет рассмотрение заявки

«продажа ПО» менеджером. Если заявка согласуется с требованиями, то менеджер её одобряет. В случае, если заявка не корректна, её отклоняют с подробным описанием причины. После одобрения заявки вендор пишет заявление на выставление продукта для его продажи. Если заявление оформлено правильно, клиент с менеджером утверждают детали, такие как: определение предварительной стоимости; согласование даты выставления продуктов. В конечном итоге, если все детали были утверждены с вендором, то менеджер переходит к оформлению предварительного выставлению ПО вендора на WEB-портал.

Для более подробного рассмотрения – декомпозируем все бизнес-процессы на подпроцессы. Декомпозиция регистрации ПО вендора приведена на рисунке 1.3.

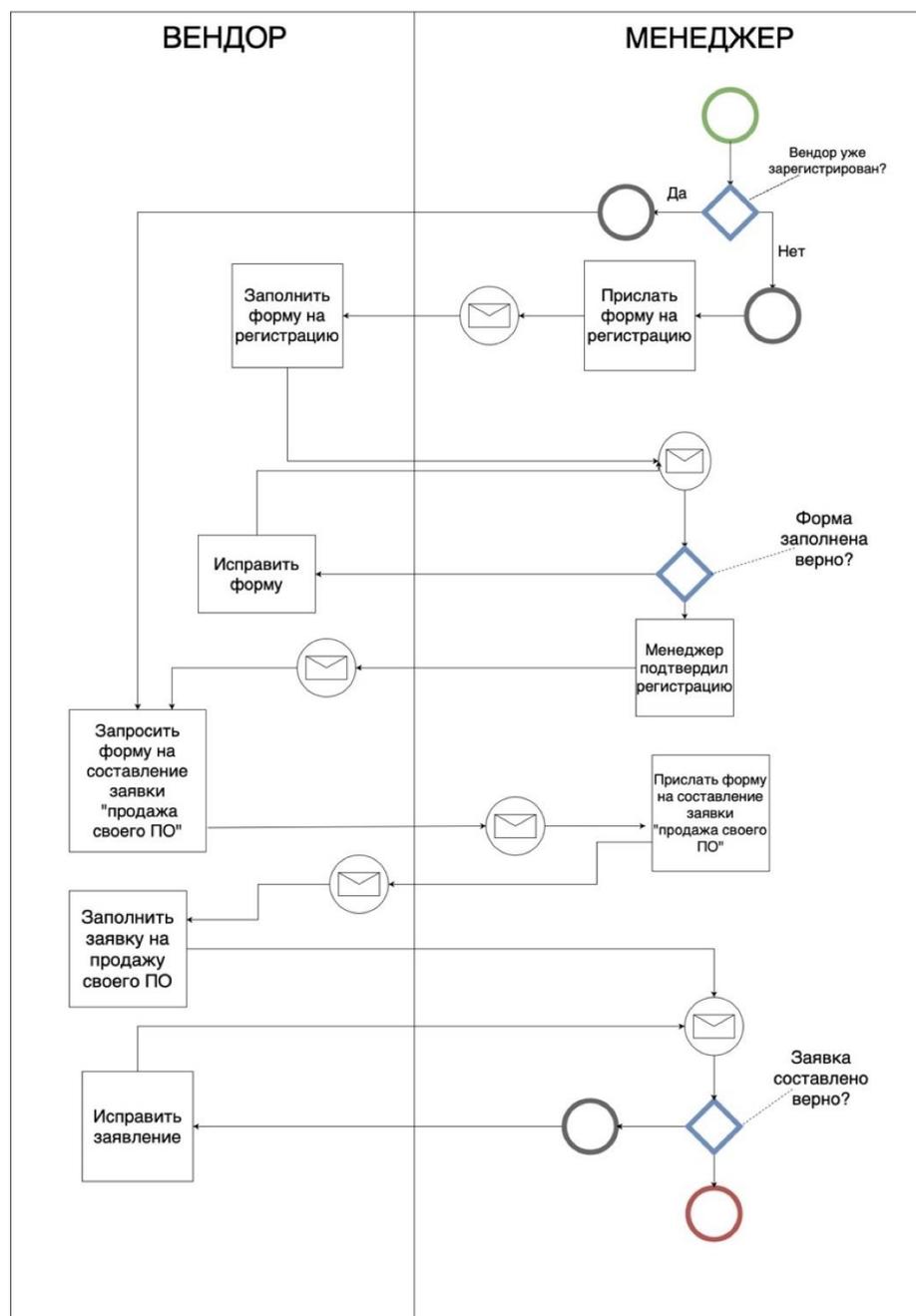


Рисунок 1.3 – Декомпозиция бизнес-процесса «Зарегистрировать вендора».

Анализ показал, что на эти бизнес-процессы тратится очень много лишних ресурсов, такие как время и кадры. Сегодня существует тенденция к уменьшению взаимодействия клиента и менеджера, которая позволяет компания значительно экономить ресурсы. Рассмотрев диаграмму «КАК ЕСТЬ», можно сделать вывод, что именно данные бизнес-процессы в значительной степени ухудшают производительность компании и понижают

КРІ (числовые показатели деятельности компании, помогающие предприятию в достижении целей или результативности и эффективности процесса). Автоматизация оптимизирует затраты ресурсов и поможет обрабатывать большее количество заявок, вследствие чего произойдет увеличение прибыли.

После проведенного анализа деятельности менеджера по работе с вендорами, используя структурное моделирование, перейдем к формированию требований к проектируемой математической модели для веб портала.

1.3.3 Обоснование необходимости автоматизированного варианта решения и формирование требований к новой технологии

FURPS+ это классификация требований к программным системам, расширенная версия классификации требований FURPS, которая расшифровывается следующим образом:

- functionality – функциональные требования;
- usability – требования к удобству использования;
- reliability – требования к надежности;
- performance – требования к производительности;
- supportability – требования к поддержке.

Дополнительно включает в себя ограничения, разделенные на следующие группы требований:

- design – ограничения проектирования;
- implementation – ограничения разработки;
- interface- ограничения на интерфейсы;
- physical – физические ограничения.

Процессы «Заполнить заявление на выставление ПО» и «Заполнить бланк сведений о вендоре и внести в БД», входящие в состав бизнес-процесса «Зарегистрировать вендора», обладают следующими недостатками:

1. Сложный процесс формирования документов вендора и менеджером по работе с вендорами.
2. Достаточно долгая обработка информации вендоров перед выставлением ПО на WEB-портал.

Данные недостатки можно устранить путем автоматизации заполнения заявок при помощи добавления математической модели в разрабатываемый WEB-портал для телекоммуникационной компании.

Таблица 1.3 - Требования к разрабатываемой математической модели.

Требования	Статус	Полезность	Риск	Стабильность
Регистрация вендора	Одобренные	Критичное	Средний	Низкая
Добавление новой заявки	Одобренные	Критичное	Средний	Низкая
Отслеживание заявки по статусу	Одобренные	Критичное	Средний	Низкая
Формирование заявления на выставление продукта	Одобренные	Критичное	Средний	Низкая
Интерфейс WEB-портала должен быть логичен прост и понятен в использовании	Одобренные	Критичное	Низкий	Низкая
Единый стиль и цветовая гамма	Одобренные	Критичное	Низкий	Низкая
Обучение и квалификация сотрудников	Одобренные	Критичное	Низкий	Низкая
Проверка введенных данных на корректность	Одобренные	Критичное	Низкий	Низкая
Разграничение прав доступа сотрудников и клиентов	Одобренные	Критичное	Средний	Средняя
Доступ к системе 24 часа в сутки	Одобренные	Критичное	Низкий	Низкая
Время отклика системы (не	Предложенные	Критичное	Средний	Средняя

должно превышать более 1 минуты)				
Расширение дополнительного функционала	Предложенные	Критичное	Средний	Средняя
Поддержка работоспособности – обновление данных	Предложенные	Критичное	Средний	Средняя
Средства разработки документации в MSWord	Одобрённые	Критичное	Низкий	Низкая
Используемая СУБД- MySQL	Одобрённые	Критичное	Средний	Средняя
Реализация бизнес-логики на JavaSpring	Одобрённые	Критичное	Средний	Средняя

1.4 Постановка задачи на разработку математической модели для ХааS-портала

Разрабатываемая математическая модель для WEB-портала будет внедрена в основной проект телекоммуникационной компании для того, чтобы её прибыль возросла путем увеличения продаж продуктов вендоров. Цель создания автоматизированного WEB-портала с математической моделью является обеспечение удобного использования со стороны вендоров, а также для упрощения работы менеджера, для дальнейшего успешного процесса покупок ХааS-продуктов для пользователей сайта. Для достижения поставленной цели создания математической модели WEB-портала были сформированы основные требования:

- введение базы данных, которая содержит всю информацию о вендоре и его программном обеспечении;
- интуитивно-понятный интерфейс для вендоров и сотрудников компании (менеджеров), который поможет быстро разобраться с функционалом портала;
- автоматизация процесса регистрации вендоров и их ПО;

- централизованное хранение всех документов в базе данных, просмотр в любой момент времени.

Сотрудники телекоммуникационной компании не будут тратить много времени и сил на выполнение лишних задач. Именно поэтому необходим автоматизированный WEB-портал, который позволит сократить время и обеспечит более качественное и надежное выполнение процессов.

На рисунке 1.4 приведена реализованная модель «КАК ДОЛЖНО БЫТЬ», в которой реализуются перечисленные требования.

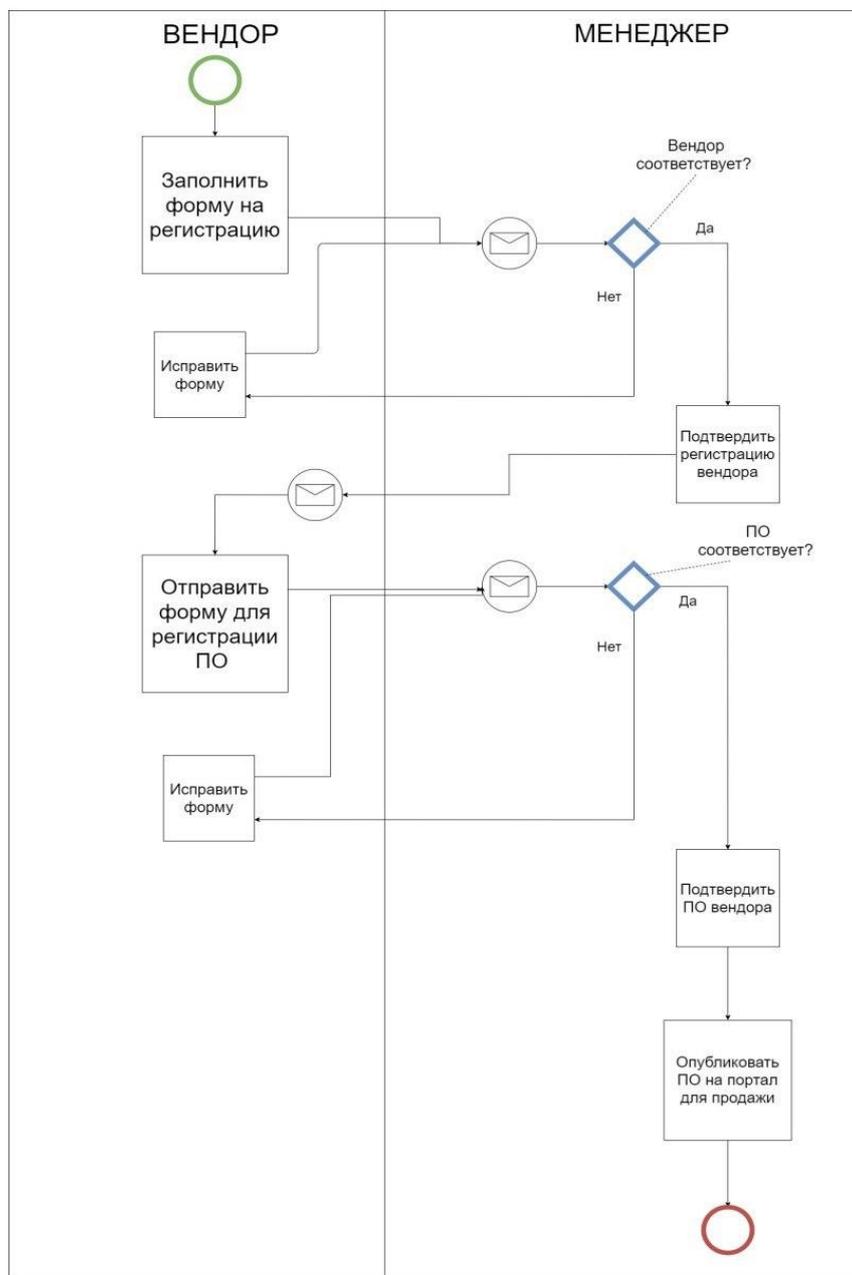


Рисунок 1.4 – Модель «КАК ДОЛЖНО БЫТЬ».

Здесь показан оптимизированный процесс, и в дальнейшем данная модель готова к финальному проектированию на WEB-портал.

Выводы по первой главе.

В первой главе была выбрана технология концептуального моделирования. Было дано определение WEB-портала, ХaaS-продукта, а также определена необходимость в реализации портала для телекоммуникационной компании. Предоставлена общая сводка ХaaS-сервисов и их детальное описание. Было описана модель «КАКЕСТЬ», которая использовалась в прошлом. Из-за неактуальности и медленного сервиса необходимо было улучшить это слабое место бизнес-процесса, для чего была построена модель «КАК ДОЛЖНО БЫТЬ» и найдена исходная точка для изменения. После было сформировано требование к новой технологии, поставлена цель разработки и выделены основные задачи последующей работы.

Глава 2. Проектирование математической модели

2.1 Проектирование математической модели для WEB-портала с помощью UseCase диаграммы

UseCase – сценарная техника описания взаимодействия. UseCase фиксирует соглашение между участниками системы о ее поведении и описывает поведение системы при ее ответах на запрос одного из участников, называемого основным действующим лицом, в различных условиях[1].

Техника UseCase упрощает разработку WEB-портала и тестирование его функций, уменьшает количество ошибок при разработке и ускоряет процесс создания.

Описание UseCase в полной мере отражает требования пользователей и системы к функциональности портала. Оно служит элементом планирования работы над порталом и базой для создания тестовых сценариев, обеспечивает полноту функциональных требований и правильную разработку портала.

Для отображения функциональной логики информационной системы WEB-портала используется диаграмма UseCase. Она представлена на рисунке номер 2.1.

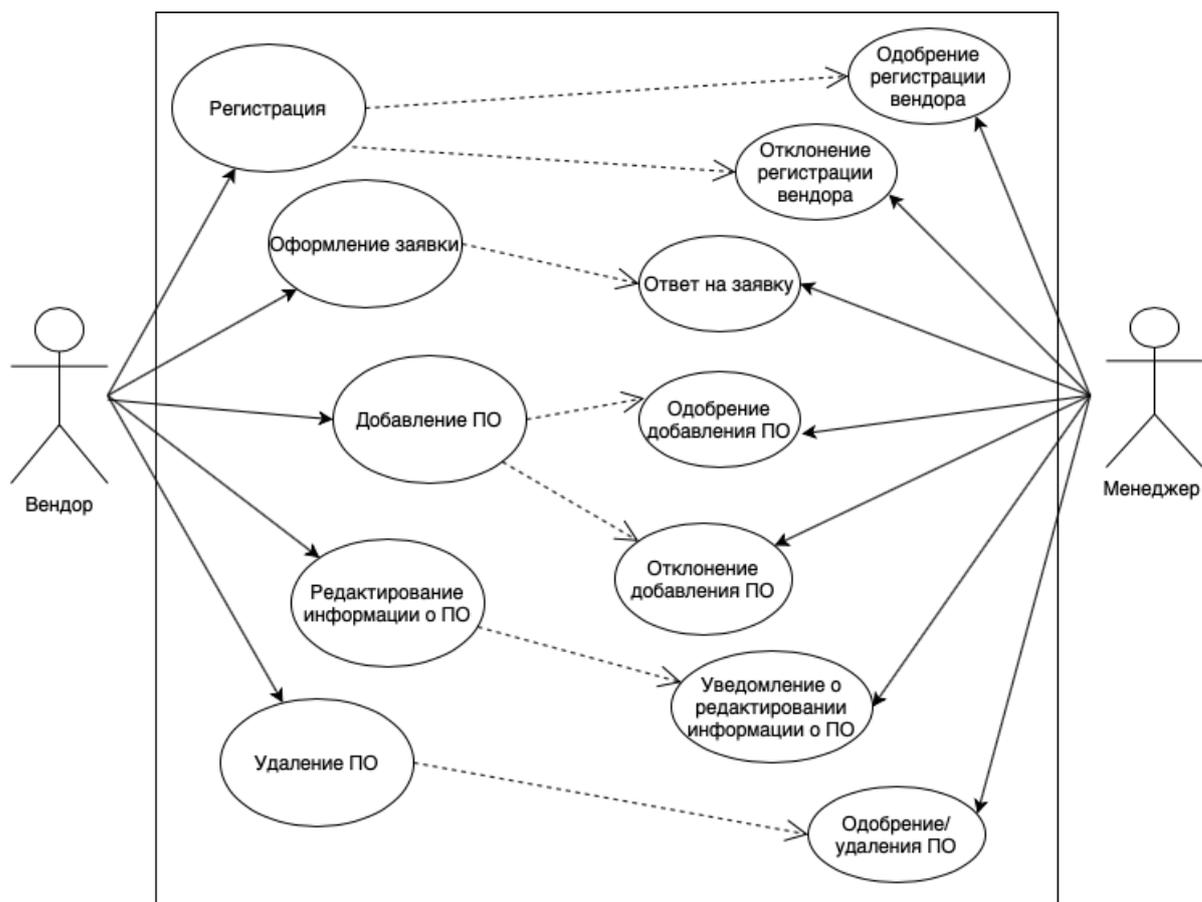


Рисунок 2.1 – Использование диаграммы UseCase.

На данной диаграмме есть 2 роли, которые будут представлены в математической модели на WEB-портале:

- Вендор – лицо, использующее портал для продажи ПО. Он предоставляет системе регистрационные данные, формируя заявку. Заполняет информацию о ПО и его конфигурации, загружает необходимые файлы и при необходимости удаляет ПО.
- Менеджер компании – должностное лицо, проверяющее заявки на регистрацию вендора, добавление или не добавление ПО на портал, редактирование информации о ПО, одобряет или удаляет ПО с портала.

2.2 Описание основных UseCase WEB-портала

Для понимания работы WEB-портала используется описание основных UseCase, только между вендором и менеджером.

Таблица 2.1 – UseCase «Регистрация вендора».

Действующие лица	Вендор, Менеджер, Система
Цель	Совершить регистрацию
Предусловие	Вендор не зарегистрирован
<p>Успешный сценарий:</p> <p>Вендор заполняет регистрационную форму. Менеджер одобряет регистрацию. Система создает учетную запись вендора.</p>	
Результат	Вендор зарегистрирован

Таблица 2.2 – UseCase «Редактирование учетной записи вендора».

Действующие лица	Вендор, Система
Цель	Изменить информацию учетной записи вендора
Предусловие	У вендора в учетной записи представлена информация, которую он хочет изменить.
<p>Успешный сценарий:</p> <p>Вендор заходит на страницу редактирования учетной записи. Вендор изменяет желаемую информацию. Система меняет информацию учетной записи вендора.</p>	
Результат	Информация учетной записи вендора изменена

Таблица 2.3 – UseCase «Оформление заявки вендора».

Действующие лица	Вендор, Система
Цель	Заполнить и подать заявку на продажу ПО
Предусловие	Вендор хочет подать заявку для дальнейшей продажи своего ПО
<p>Успешный сценарий:</p> <p>Вендор заходит на страницу подачи заявки. Вендор заполняет требуемую информацию и отправляет её на обработку. Система отправляет заявку менеджеру.</p>	
Результат	Заявка оформлена

Таблица 2.4 – UseCase «Заявка вендора одобрена».

Действующие лица	Менеджер, Система
Цель	Подтвердить заявку о продаже ПО и зарегистрировать вендора
Предусловие	Менеджеру пришла заявка от вендора
<p>Успешный сценарий:</p> <p>Менеджер обрабатывает заявку вендора. Менеджер одобряет заявку. Система регистрирует вендора, как продавца ПО Система отображает вендору, что заявка одобрена.</p>	
Результат	Заявка одобрена, вендор зарегистрирован

Таблица 2.5 – UseCase «Добавление ПО вендора».

Действующие лица	Вендор, Менеджер, Система
Цель	Добавить ПО вендора на WEB-портал
Предусловие	ПО вендора не загружено
<p>Успешный сценарий:</p> <p>Вендор заполняет форму загрузки и прикрепляет необходимые файлы ПО. Менеджер одобряет загрузку. Система загружает ПО на портал.</p>	
Результат	ПО загружено на WEB-портал

Таблица 2.6 – UseCase «Редактирование информации о ПО».

Действующие лица	Вендор, Менеджер, Система
Цель	Изменить информацию о ПО вендора
Предусловие	Информация о ПО вендора не отредактирована
<p>Успешный сценарий:</p> <p>Вендор заходит на страницу редактирования информации о ПО. Вендор редактирует информацию о ПО. Менеджер одобряет изменения. Система изменяет информацию о ПО.</p>	
Результат	Информация о ПО вендора изменена

Таблица 2.7 – UseCase «Удаление ПО с портала».

Действующие лица	Вендор, Система.
Цель	Удалить ПО вендора с портала
Предусловие	ПО вендора находится на портале
Успешный сценарий: Вендор заходит на страницу редактирования информации о ПО. Вендор нажимает кнопку «Удалить». Система удаляет ПО с портала.	
Результат	ПО вендора удалено с портала

2.3 Проектирование архитектуры данных

Для создания WEB-портала необходимо спроектировать архитектуру данных. Этот процесс состоит из двух шагов:

- Логическое описание сущностей и связей между ними.
- Построение ER-диаграммы реляционной базы данных, которая будет использоваться WEB-порталом.

Рассмотрим полную картину сущностей, в котором присутствуют пользователи для понимания завершеного бизнес-процесса продажи ХaaS-продуктов:

- Сущность «Пользователь». Отражает клиента портала. Содержит следующие данные: имя, фамилия, почта, пароль, название компании, где он работает, на рынке какой страны он собирается покупать ПО (необязательные данные), дата и время регистрации.

- Сущность «Вендор». Отражает вендора, который использует портал для продажи ПО. Содержит следующие данные: имя, фамилия, имя для отображения (необязательные данные), почта, пароль, логин сервиса, название компании, которую он представляет, позиция в компании (необязательные данные), номер телефона, дата и время регистрации, менеджер, одоббивший регистрацию вендора.

- Сущность «Менеджер». Отражает менеджера, который проводит модерацию портала. Содержит следующие данные: имя, фамилия, имя для отображения (необязательные данные), почта, пароль, логин, id работника.

- Сущность «ПО». Отражает загружаемое на портал ПО. Содержит следующие данные: название, вендор, все возможные конфигурации данного ПО, все возможные тарифы, дата и время добавления на портал, возможность автоматического продления подписки, менеджер, одоббивший добавление ПО.

- Сущность «Категория». Отражает категории, к которым относится данное ПО. Содержит следующие данные: название категории.

- Сущность «Период контракта». Отражает все возможные периоды, на которые пользователь может оплатить подписку на ПО. Содержит следующие данные: ПО, период в месяцах.

- Сущность «Заказ». Отражает основную информацию о заказе без конфигурации. Содержит следующие данные: пользователь, дата и время создания, статус, общая цена, платежный адрес.

- Сущность «Заказ продукта». Отражает каждый продукт, вошедший в заказ. Содержит следующие данные: заказ, ПО, конфигурация, включено ли автопродление, количество, тариф, подходящий под конфигурацию и период, период.

- Сущность «Платежный аккаунт». Отражает адрес для выставления счета. Содержит следующие данные: пользователь, адрес, город, код страны, почтовый индекс, язык, код валюты, компания (необязательная информация), номер телефона (необязательная информация).

Рассмотрим логические связи между сущностями:

- «Пользователь» - «Заказ»: пользователь делает заказ.
- «Пользователь» - «Платежный аккаунт»: пользователь имеет платежный аккаунт.
- «Менеджер» - «Вендор»: менеджер одобряет регистрацию вендора.
- «Менеджер» - «ПО»: менеджер одобряет добавление ПО на портал.
- «Вендор» - «ПО»: вендор добавляет ПО на портал.
- «ПО» - «Категория»: ПО имеет категорию, категория содержит ПО.
- «ПО» - «Период контракта»: ПО имеет периоды контракта.
- «ПО» - «Заказ продукта»: ПО входит в заказ.
- «Заказ» - «Заказ продукта»: заказ содержит заказ продукта.
- «Заказ» - «Платежный аккаунт»: заказ проходит по платежному аккаунту.

Когда необходимые сущности и связи между ними описаны, построим ER-диаграмму реляционной базы данных, представленной на рисунке номер 2.2.

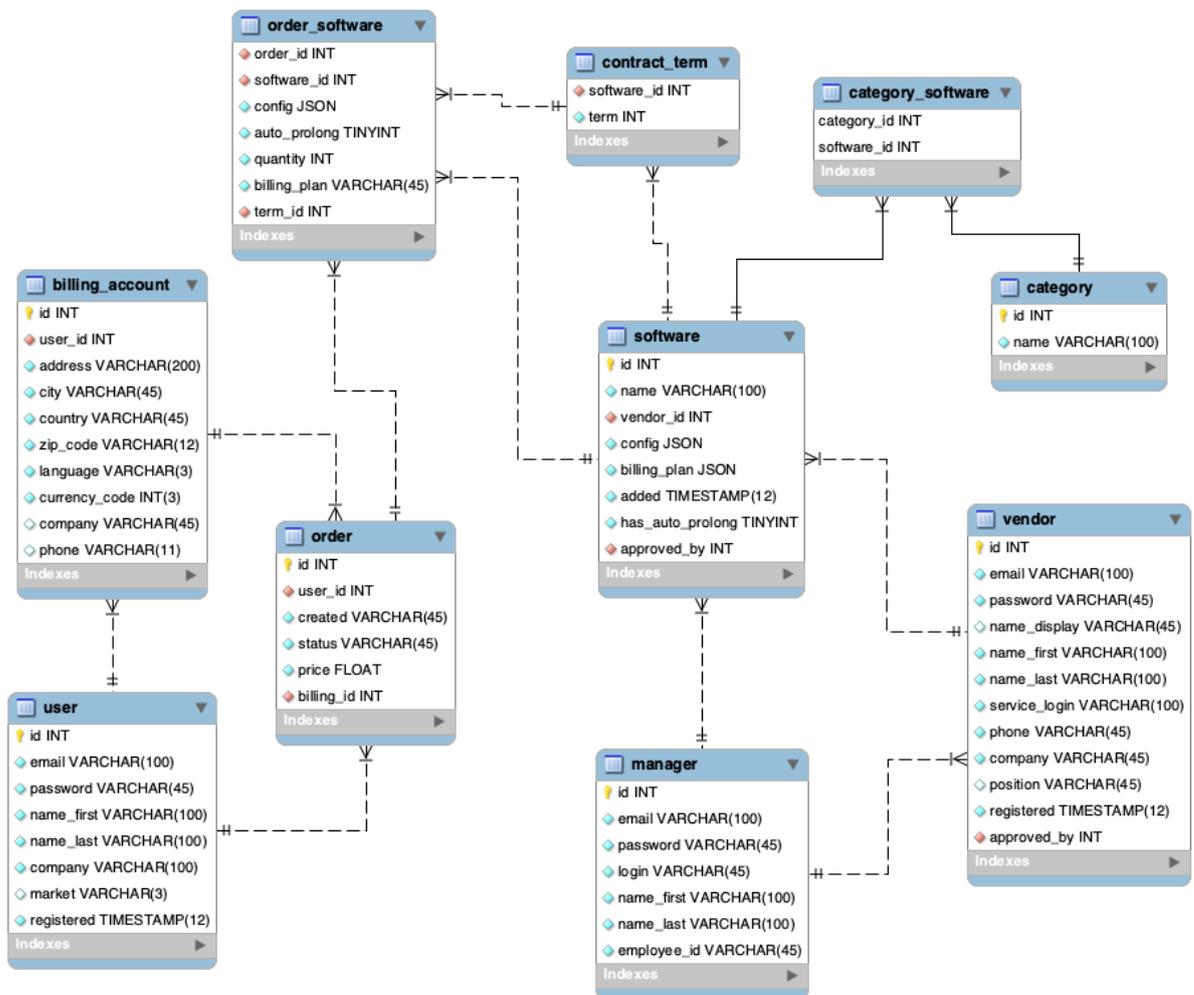


Рисунок 2.2 – Диаграмма рабочих баз данных.

Диаграмма представляет рабочую реляционную базу данных, которая будет использована при создании WEB-портала, данная диаграмма показывает общую картину: связи между сущностями, в которых есть участник в роли пользователя. Эта информация необходима для отображения участия пользователя на WEB-портале. Сущность «Пользователь» и его логические связи («Пользователь» - «Заказ»: пользователь делает заказ. «Пользователь» - «Платежный аккаунт»: пользователь имеет платежный аккаунт) не входят в разрабатываемую математическую модель, но они необходимы для упоминания, чтобы видеть пользу разрабатываемой математической модели на WEB-портале.

2.4 Описание работы алгоритма SHA-1 для хеширования паролей

Одно из ключевых требований вендоров к проектируемому порталу – это безопасность. Без обеспечения надлежащей безопасности будет невозможно привлечь количество вендоров, необходимое для целесообразности создания данного портала. Кроме того, возникновение хотя бы одного случая взлома аккаунта вендора может серьезно повлиять на репутацию компании и привести к значительному падению выручки.

Важной частью обеспечения безопасности и защитой от взлома является хеширование паролей. Проведя анализ существующих алгоритмов, было принято решение использовать SecureHashAlgorithm 1 в разрабатываемом портале.

SecureHashAlgorithm 1 (SHA-1) – алгоритм криптографического хеширования. Главной особенностью алгоритма является то, что он достаточно популярен в наше время, его использует во многих криптографических приложениях и протоколах. Также он рекомендован в качестве основного для государственных учреждений в США. Он является схожим с алгоритмом MD4, так как в его основе лежат схожие принципы, который внёс Рональдом Ривестом при проектировании MessageDigest 4.

Данный алгоритм будет задействован для того, чтобы захешировать пароли вендоров, зарегистрировавшихся на WEB-портале.

Работа алгоритма SHA-1 заключается в том, что он реализует хеш-функцию, основанную на идеи сжатия данных. Аргументами функции сжатия являются блок сообщения длиной 512 бит и результат предыдущего блока сообщения. Предыдущий блок представляет собой значение всех хеш-блоков до этого момента:

$$M_i = h_i = f(M_i, h_{i-1}) \quad (1)$$

где M_i – это искомый хеш-блок сообщения;

h_i – это хеш-значение всего сообщения;

$f(M_i, h_{i-1})$ – это функция сжатия.

Исходя из формулы выше, можно сказать, что хеш-значение каждого последующего сообщения является результатом совокупности предыдущих блоков.

Пример работы алгоритма SHA-1 будет представлен на рисунке 2.3.

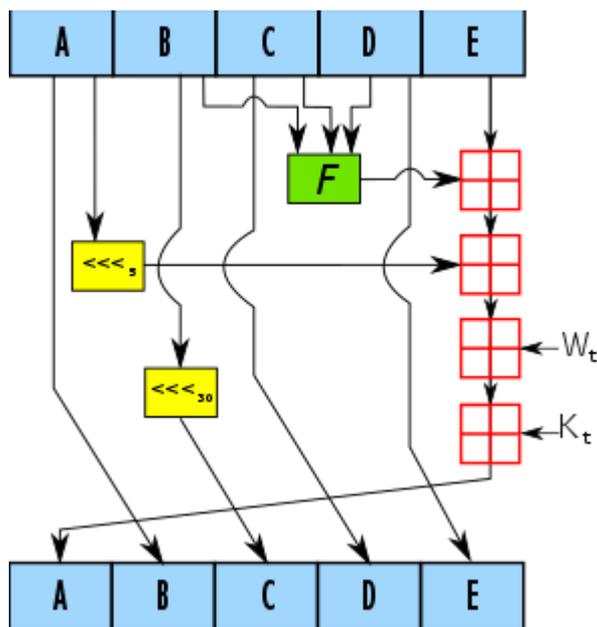


Рисунок 2.3 – Пример работы алгоритма хеширования SHA-1.

На данном рисунке изображено разбиение входного сообщения на блоки, который, в свою очередь, шифруются путём зависимости от предыдущих.

Рассмотрим подробную работу алгоритма SHA-1. Исходное сообщение разбивается на блоки по 512 бит в каждом. В заключительный блок сообщения, если он не кратен 512 бит, дописывается дополнительные биты до длины, кратной 512 бит. Сначала добавляется 1 бит, а потом нули, чтобы длина блока стала равной $(512 - 64 = 448)$ бит. В этих 64 битах записывается длина исходного сообщения в big-endian формате. Если же заключительный блок имеет длину превышающую 448 бит, но менее 512 бит, то заполнение

недостающего размера осуществляется следующим образом: сначала добавляется 1 бит, затем нули вплоть до конца 512-битного блока; после этого создается ещё один 512-битный блок, который наполняется до 448 бит нулями, после чего в оставшиеся 64 битах записывается длина исходного сообщения в битах. Дополнение последнего блока осуществляется всегда, даже если сообщение уже имеет нужную длину.

Для начала работы SHA-1 алгоритма определяются четыре нелинейных операции:

$$F_t(m, l, k) = (m \wedge l) \vee (\neg m \wedge k), K_t = 0x5A827999, \text{ где } 0 \leq t \leq 19 \quad (2)$$

$$F_t(m, l, k) = m \oplus l \oplus k, K_t = 0x6ED9EBA1, \text{ где } 20 \leq t \leq 39 \quad (3)$$

$$F_t(m, l, k) = (m \wedge l)(m \wedge k)(l \wedge k), K_t = 0x8F1BBCDC, \text{ где } 40 \leq t \leq 59 \quad (4)$$

$$F_t(m, l, k) = m \oplus l \oplus k, K_t = 0xCA62C1D6, \text{ где } 60 \leq t \leq 79 \quad (5)$$

где t – это количество битовых слов;

m, l и k – это константы;

K_t – это переменные используемые при сдвиге битов.

Главный цикл итеративно обрабатывает каждый 512-битный блок. Итерация состоит из четырёх этапов по двадцать операций в каждом. Блок сообщения преобразуется из 1632-битовых слов в 80 32-битовых по следующему правилу:

$$W_t = M_t, \text{ где } 0 \leq t \leq 15 \quad (6)$$

$$W_t = (W_{t-3} \oplus W_{t-8} \oplus W_{t-14} \oplus W_{t-16}) \ll 1, \text{ где } 16 \leq t \leq 79 \quad (7)$$

где " \ll " – это циклический сдвиг влево.

Далее на рисунке 2.4 будет представлен результат работы SHA-1.

```
SHA-1("The quick brown fox jumps over the lazy dog")
= 2fd4e1c6 7a2d28fc ed849ee1 bb76e739 1b93eb12
```

Рисунок 2.4 – результат работы алгоритма хеширования SHA-1 на основе входного текста.

Далее будет представлена блок-схема обобщённого варианта алгоритма регистрации вендора и хеширования пароля (рисунок 2.5).

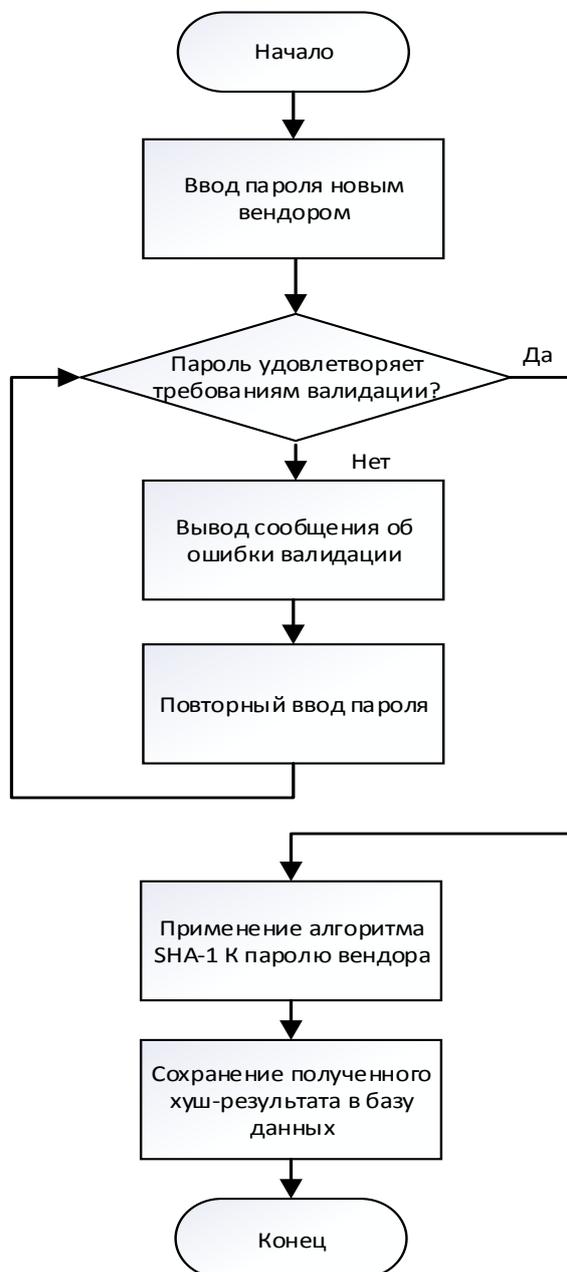


Рисунок 2.5 – Блок-схема обобщённого варианта регистрации вендора в WEB-приложении с применением алгоритма хеширования пароля SHA-1

Из рисунка выше видно, что в WEB-приложении будет поддерживаться валидация введённого вендором пароля. В случае неверного ввода будет

выводится сообщение об ошибке, а если всё пройдет успешно, то пароль захешируется с применением алгоритма SHA-1 и сохранится в базу данных.

Сравним алгоритм SHA-1 с алгоритмом MD5:

Сходства:

- они оба состоят из четырех этапов обработки;
- каждое последующее действие прибавляется к ранее полученному результату;
- размер обрабатываемого кластера 512 бит;
- сложение по модулю равно 2^{32} , совместимы с 32-битной архитектурой.

Различия:

- SHA-1 на четвертом этапе используется функцию, также как и на втором этапе;
- каждое действие MD5 использует уникальную прибавляемую константу. В SHA-1 константы используются те же самые для каждой из четырех операций;
- SHA-1 использует пять переменных;
- SHA-1 исправляет ошибки с использованием циклического кода;
- В MD5 четыре сдвига отличаются от значений, используемых на предыдущих этапах. В SHA каждый этап использует постоянное значение сдвига.
- В MD5 четыре разных элементарных логических функции, в SHA-1 их три.
- В MD5 длина дайджеста составляет 128 бит, в SHA-1 оно 160 бит.
- SHA-1 содержит раундов 80 вместо 64 и выполняется на 160-битном буфере, а не на 128-битном буфером MD5. Таким образом, SHA-1 должен

выполняться приблизительно на 25% медленнее, чем MD5 на той же аппаратуре.

В результате получаем, что алгоритм отлично подойдёт для хеширования паролей регистрирующихся вендоров. Также можно будет проверять правильность набранного пароля вендором путём преобразования хешированной версии пароля в оригинальную и их сравнения.

Вывод по второй главе.

На этом этапе работы была выполнена большая часть ВКР, заключающая в себя проектирование WEB-портала при помощи UseCase диаграммы. Было подробно расписано о понятии UseCase и основных UseCaseWEB-портала. Также была представлена математическая модель хеширования паролей регистрирующихся вендоров на WEB-портале. Для создания WEB-портала необходимо спроектировать архитектуру данных, что и было так же сделано в этой главе.

Глава 3. Практическая реализация математической модели для ХааS-портала

3.1 Выбор технологии разработки программного обеспечения

Важнейшим этапом при разработке ХааS-портала является выбор технологии реализации. Для правильной реализации ХааS-портала можно выделить несколько языков программирования: Python, Ruby, Java.

Python – высокоуровневый язык программирования, разработанный для быстроты и удобства использования, ориентированный на повышенной производительности и читаемости кода.

Ruby–динамический язык программирования, который обладает независимой от операционной системы реализацией многопоточности и предназначен для разработки WEB-приложений.

Java–объектно-ориентированный платформо-независимый язык программирования, который способен работать с базами данных и создавать динамические WEB-сайты.

Для того, чтобы сделать правильный выбор языка программирования проведём сравнительный анализ этих трёх выбранных языков программирования по критериям, представленным в таблице 3.1.

Таблица 3.1 – Сравнительный анализ языков программирования

Критерий	Языки программирования		
	Python	Ruby	Java
Обучение в университете	+	-	+
Простота	-	-	+
Результативность	+	-	+
Быстродействие	-	+	+
Обучаемость	+	+	+
Обширность возможности	-	-	+

работы с базами данных			
Потребление памяти	-	+	-
Итого	3	3	7

По итогам сравнительного анализа языков программирования был выбран язык программирования Java, так как данный язык имеет широкие возможности для работы с базами данных, а также поддерживается большим количеством серверов.

3.2 Описание основного принципа работы ХааS-портала для телекоммуникационной компании

Благодаря переходу из модели «КАК ЕСТЬ» в модель «КАК ДОЛЖНО БЫТЬ» были произведены некоторые модернизации. Данный портал позволяет вендорам и клиентам регистрироваться и авторизоваться в системе, редактировать профиль, выбирать товары, просматривать список услуг, совершать покупку. Система портала так же позволяет работникам компании управлять и корректировать необходимый функционал, отслеживать список покупок и т.д.

Рассмотрим форму для заполнения вендора:

Details		Integration Data	
Address	<input type="text" value="1 Microsoft Way"/>	Annual Revenue of company	<input type="text"/>
Business Units support	<input type="text" value="false"/>	CSR E-mail	<input type="text" value="Test@gmail.com"/>
CSR Phone number	<input type="text" value="05693 / 9950-0"/>	City	<input type="text" value="Redmond"/>
Cost Centers support	<input type="text" value="false"/>	Country	<input type="text" value="United States"/>
Culture	<input type="text"/>	Customer Account Number	<input type="text" value="79050"/>
Customer Category	<input type="text"/>	Customer Identification Number (CIN)	<input type="text"/>
Customer Reference	<input type="text"/>	Customer Since	<input type="text" value="22/05/2017"/>
Customer Tags	<input type="text"/>	DMP mastered	<input type="text"/>
Detailed Sector	<input type="text" value="007"/>	E-mail	<input type="text" value="10100612customer@mail.ru"/>
External Customer Subscriptions	<input a044b16a-1861-4308-8086-a3a3b506fac2\".1,\"1...\""="" type="text" value="{\"/>	Germany MS Customer Domain	<input type="text"/>
Google domain	<input "="" type="text" value="goog-test.test14052020.google-reseller.bechte-cl...\"/>	Google primary admin ID	<input type="text" value="114870486264179326094"/>
Industry Sector	<input type="text" value="003"/>	LDAP Configuration	<input type="text"/>
MS Customer Domain	<input type="text" value="10100612test05082020.onmicrosoft.com"/>	Main Market	<input type="text" value="+ DE"/>
Mont profile Id	<input type="text"/>	No. of Employees in organisation	<input type="text" value="100-1000"/>
Reseller ID	<input type="text" value="10100612"/>	SFA Customer ID	<input type="text"/>
ZSquare Account Email	<input type="text"/>	Zip code	<input type="text" value="98052"/>

Рисунок 3.1 – Формы на заполнение для регистрации вендора.

Billing Account ID	<input type="text" value="16841738"/>
CSR Name	<input type="text" value="CSR csr"/>
Company Name	<input type="text" value="10100612CompanyName"/>
Credit Rating	<input type="text"/>
Customer Age	<input type="text" value="37"/>
Customer Qualification	<input type="text"/>
Customer Subscriptions	<input 6be07941-1dfc-454d-b8b3-0dd48a470a3b\".1,\"f...\""="" type="text" value="{\"/>
Description	<input type="text"/>
ESET Account Email	<input type="text" value="10100612customer@mail.ru"/>
Google customer ID	<input type="text" value="C01aefikh"/>
Google transfer token	<input type="text"/>
Language	<input type="text"/>
Mont MS Domain Prefix	<input type="text"/>
Phone number	<input type="text" value="88005553535"/>
State	<input type="text"/>
inn	<input type="text"/>

Рисунок 3.2 – Продолжение формы на заполнение для регистрации вендора.

Данная форма нужна для указания информации о вендоре, для последующей регистрации вендора в системе. После заполнения формы самим вендором происходит процесс рассмотрения этой заявки со стороны системы (на правильность введения данных) и менеджера, который и принимает эту форму или переотправляет вендору для доработки.

The screenshot displays a web application interface for managing vendor scenarios. At the top left, the user is logged in as 'Dr. Web'. The interface is divided into two main sections: 'General Parameters' and 'Endpoints'. The 'Endpoints' section is currently active. On the left side, there is a search bar and a list of items: 'Prolong' (highlighted in blue), 'Purchase', 'Resume', and 'Suspend'. The main content area shows a 'Scenario Name' dropdown menu set to 'Prolong'. Below this, there are two toggle switches: 'Active' (which is turned on) and 'Public' (which is turned off). Further down, there are three tabs: 'Tasks', 'Tasks for Error Handling', and 'Advanced Settings'. The 'Tasks' tab is selected, showing a list of tasks with buttons for 'Add', 'Up', 'Down', 'Edit', 'Remove', and a 'Choose Template' button. The task list includes 'Initialize Context' and 'Prolong Subscription'.

Рисунок 3.3 – Форма для регистрации ПО вендора.

На данном рисунке мы можем увидеть, как происходит регистрация ПО вендора, который хочет в дальнейшем его продать. Вендор формирует продукт: его название, информацию о нём, возможные скидки при различных условиях и статус своего продукта. О статусе продукта, можно увидеть на рисунке 3.4.

• DDoS Guard

General Parameters Endpoints

Search

Get Personal Account URL

Increase License Quantity

Prolong

Purchase

Resume

Schedule Update Subscription

Suspend

Update Subscription

Рисунок 3.4 – Статус продукта вендора.

Менеджер видит заполненные статусы и подтверждает их в соответствии с поставленными требованиями к ПО вендоров на WEB-портал.

Теперь рассмотрим конкретные функции системы со стороны клиента. При запуске главной страницы появляется меню с возможностью покупок товаров. Изображено на рисунке 3.5.

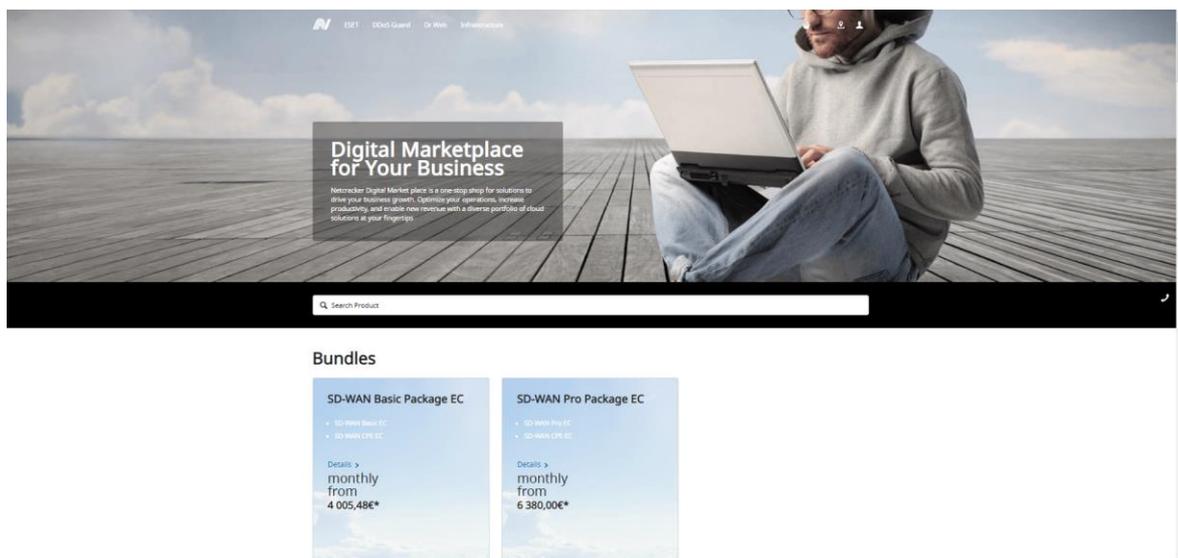


Рисунок 3.5 – Главная страница со стороны клиента.

Перед покупкой клиенту необходимо пройти аутентификацию и авторизацию на портале для создания личного кабинета. При нажатии на иконку в правом верхнем углу, выглядящую, как человек, отображается форма (на рисунке 3.6) для регистрации и аутентификации/авторизации пользователя.

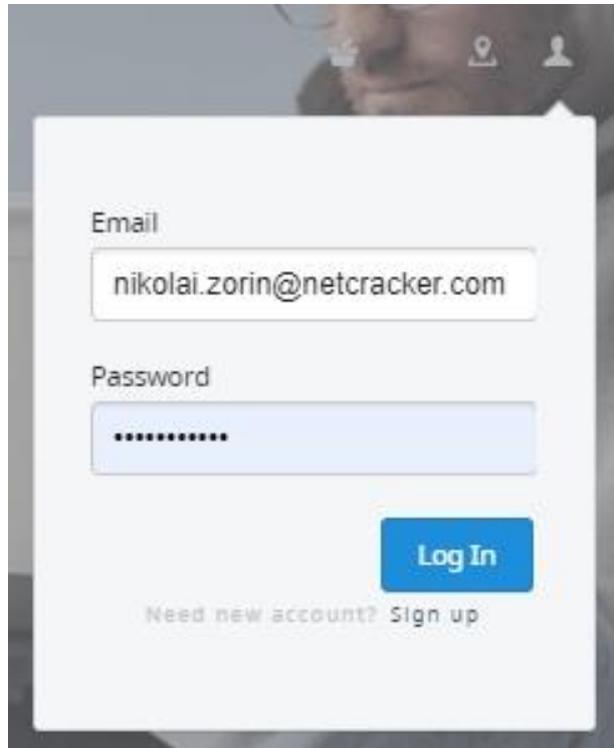


Рисунок 3.6 – Форма аутентификации и авторизации пользователя.

После того, как пользователь введет свои данные и войдет в систему, ему станут доступны такие функции системы как:

- просмотр продуктов;
- формирование корзины;
- оставить отзыв с возможностью обратной связи;
- просмотр и редактирование профиля.

На рисунке 3.7 показаны продукты, которые доступны для покупки клиенту. Допустим, зайдём в один из них для оформления его. При нажатии на продукт вы попадаете в более детальную форму, где можно указать дополнительную информацию.

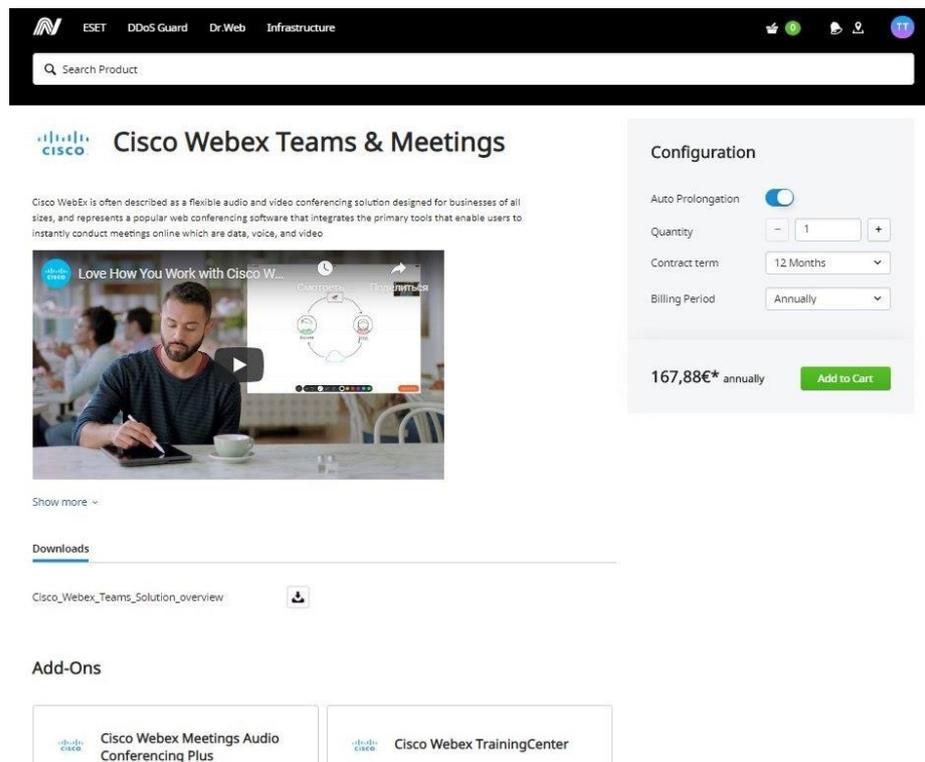


Рисунок 3.7 – Просмотр продуктов.

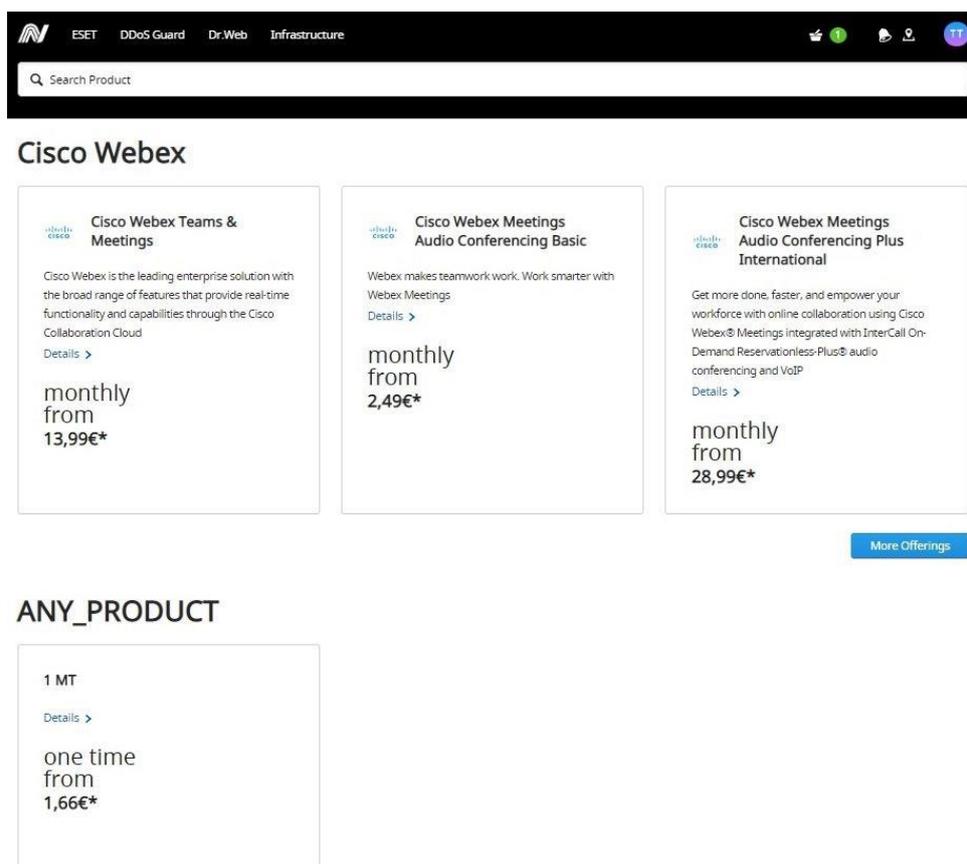


Рисунок 3.8 – Просмотр информации о выбранном продукте.

Нажав на кнопку «AddtoCart», пользователь получает уведомление, что этот товар отправился в корзину покупок с возможностью перейти в неё или же продолжить выбирать другие товары.

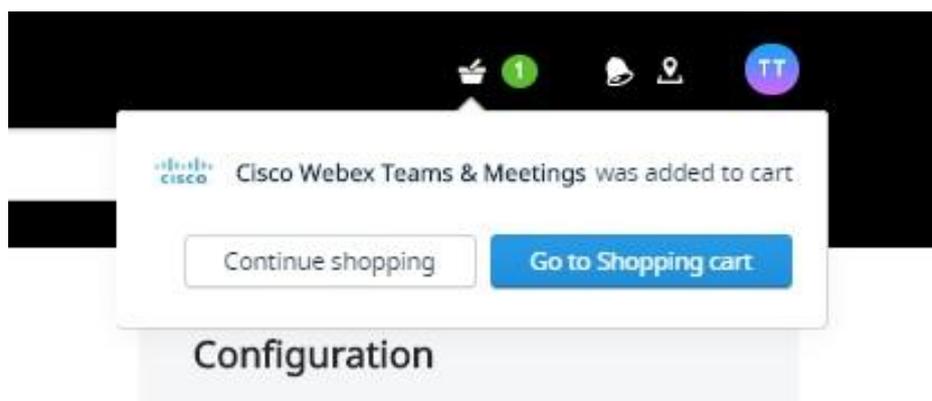


Рисунок 3.9 – Уведомление о добавлении продукта в корзину.

После попадания в корзину товаров путем нажатия на кнопку «GotoShoppingcart» пользователю нужно пройти 3 шага для завершения покупки.

Для начала это просмотр всей корзины с возможным увеличением количества товара.

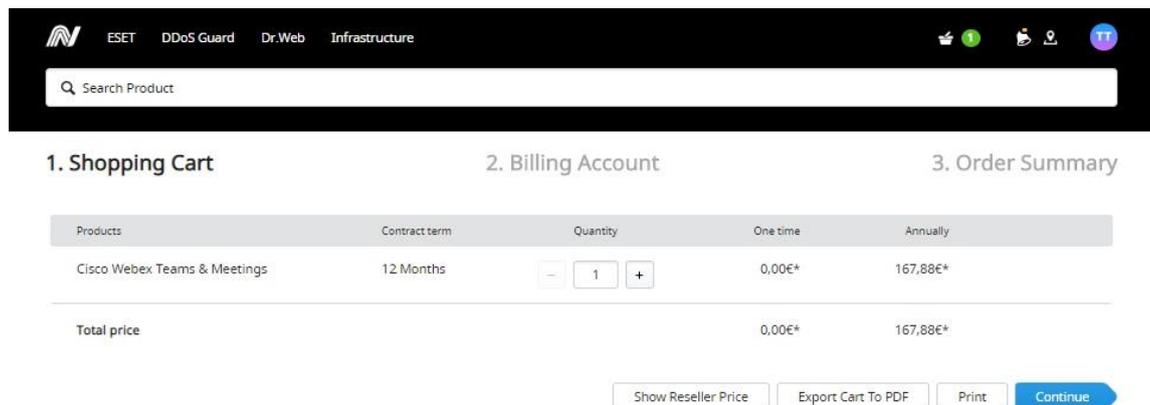


Рисунок 3.10 – Первый шаг завершения покупки.

На втором шаге подтверждается биллинг аккаунт покупателя. Биллинг аккаунт – это совокупность инструментов, работающих автоматизировано и используемых для выставления счетов абонентам.



Рисунок 3.11 – Второй шаг завершения покупки.

На последнем шаге подтверждаются условия и положения выбранного товара. Каждый вендор вправе загружать сюда другую дополнительную информацию, необходимую для ознакомления клиентом.

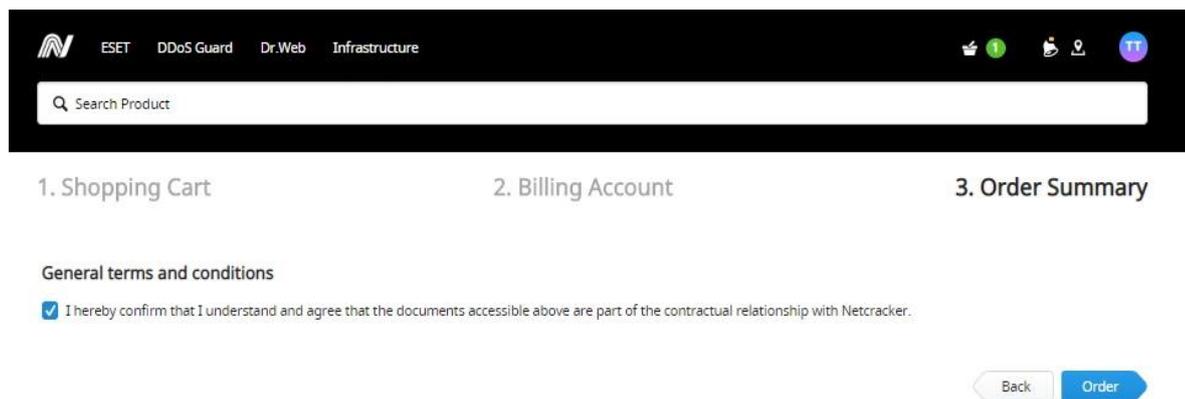


Рисунок 3.12 – Третий шаг завершения покупки.

В ходе работы был спроектирован цифровой портал, в котором были решены поставленные в начале работы основные функциональные задачи, а также учтено и реализовано удобство пользования. Во время описания основного принципа работы портала с ХaaS-продуктами для телекоммуникационной компании был представлен алгоритм выставления вендором продуктов и реализация покупок клиентов, желающих приобрести продукт. По итогам описания работы портала для телекоммуникационной компании можно сделать вывод, что данный разработанный ХaaS-портал работоспособен и удовлетворяет заданным требованиям.

3.3 Тестирование математической модели для ХaaS-портала телекоммуникационной компании

Тестирование программного продукта – это процесс проверки программного продукта на предмет ошибок и соответствия требованиям. На проверку программного продукта на предмет ошибок будет использоваться метод «Чёрного ящика (Blackboxtesting)». Тестирование черного ящика – это проверка продукта согласно предъявляемым требованиям без изучения

программного кода и без доступа к базе данных. Для проведения теста, необходимо составить тест кейсы, которые представлены в таблице 3.2.

Таблица 3.2 – Тест-кейсы ХааS-портала.

№	Название	Инструкции	Результат
1. Проверка наличия всех необходимых элементов и тестирования способности системы			
1.1.	Укомплектованность главного меню	Проверить наличие необходимых элементов: продукты, сервисы, профиль, обратная связь	Все необходимые элементы в наличии и готовы к работе
1.2	Укомплектованность меню для входа и регистрации	Проверить наличие необходимых элементов	Все необходимые элементы в наличии и готовы к работе
1.3.	Укомплектованность меню пользователя	Проверить наличие необходимых элементов: корзина покупок, страница личного кабинета	Все необходимые элементы в наличии и готовы к работе
1.4	Формы для ввода данных	Проверить наличие форм для ввода данных: информация пользователя, обратная связь	Формы в наличии и готовы к заполнению
1.5	Наличие модуля вход/выход в систему	Проверить наличие модуля: вход/выход из системы	Модуль в наличии и готов к работе
1.6	Наличие каждого модуля каталога продуктов	Проверить наличие всех модулей каталога продуктов	Модуль в наличии и готов к работе
2. Тестирование функциональных возможностей системы			
2.1	Соответствие элементов панели управления	Проверить соответствие элементов панели управления их функциональному модулю	Все элементы соответствуют своему функциональному модулю
2.2	Соответствие вывода информации о выбранном продукте	Проверить соответствие вывода информации о выбранном продукте	Все данные выводятся правильно
2.3	Соответствие данных о выполненных покупках	Проверить соответствие данных о выполненных покупках	Все заполненные данные выводятся и записываются корректно

3. Тестирование бизнес-логики (импорт данных и т.п.)			
3.1	Работа системы без авторизации	Проверить возможность подачи заявки без авторизации	Без регистрации модуль подачи заявки не доступен
3.2	Работа модуля редактирования	Отредактировать информацию о продукте	Редактирование информации доступно после авторизации
3.3	Вывод данных для вендора	Проверить вывод данных о покупке для вендора	Вывод производится корректно
3.4	Модуль редактирования своего продукта	Проверить модуль редактирования продукта вендора	Редактирование доступно
3.5	Модуль изменения личных данных вендора	Проверить возможность изменение данных клиента	Изменения возможны
4. Тестирование дополнительных функций			
4.1	Выход из системы	Проверить выход из профиля	Выход из профиля происходит корректно
4.2	Сохранение сеанса вендора	Проверить сохранение сеанса вендора	Сеанс сохраняется

По итогам тестирования система доказала свою работоспособность и удовлетворила все выдвинутые технические требования.

Далее перейдем к проверке пригодности системы для использования на предприятии.

3.4 Обоснование эффективности использования разработанной математической модели для ХaaS-портала телекоммуникационной компании

Основной целью при разработке математической модели для ХaaS-портала телекоммуникационной компании была автоматизация оптимизация процессов регистрации вендоров и выставление на продажу их продуктов (ПО).

Данная цель достигается посредством создания необходимых условий для работы вендоров, быстрой пропускной способностью сайта и работы менеджера со стороны компании.

При помощи разработанной математической модели для вендора стало проще и быстрее регистрироваться, а процесс начала продаж начинается заметно скорее. Если раньше процесс от регистрации до выставления продукта на продажу занимал в среднем 4 недели, то сейчас срок сократился до одной недели.



Диаграмма 3.1 – Среднее количество недель от регистрации до продажи ПО вендоров.

Одно из основных достоинств для пользователя является ускоренное получение информации и необходимых продуктов, путем оптимизации времени и ресурсов.

Фактор оценки и работоспособности Хаас-портала для предприятия является время, которое необходимо для обработки одной заявки. В связи с имплементации математической модели на портал среднее время обработки является сопоставимым с требуемым.

Можно сделать вывод, что разработка и дальнейшее использование системы полностью оправдывает себя и оптимизирует работу как и менеджера, так и вендора. Разработанная математическая модель для Хаас-

портала имеет возможности для расширения функциональности при необходимости.

Вывод по третьей главе.

Была выбрана технология разработки математической модели для ХааS-портала для телекоммуникационной компании. Также выполненная работа отвечает предъявленным требованиям заказчика.

Была описана работа как разрабатываемой модели для портала, так и самого портала, представлены экранные формы портала, раскрыты возможности модели и самого портала.

После тестирования ХааS-портала для телекоммуникационной компании, была проведена проверка системы на её пригодность для использования, в результате данной проверки было доказано, что портал может применяться, использоваться и влиять на увеличение продаж вендоров. По итогу ХааS-портал был признан исправным и работоспособным.

Заключение

Итогом выпускной квалификационной работы является разработанная математическая модель для WEB-портала телекоммуникационной компании. Данный портал предназначен для автоматизированных продаж ХaaS-продуктов со стороны вендоров и для покупок этих продуктов со стороны клиента, который придёт на WEB-портал. Также разработанная математическая модель позволяет производить быструю регистрацию вендоров и их ПО.

Во время проектирования математической модели для ХaaS-портала был произведен анализ предметной области, на основании которого было принято решение об оптимизации портала благодаря разработке математической модели «КАК ДОЛЖНО БЫТЬ». Были сформулированы основные требования к этой модели на ХaaS-портале и были определены основные функции, которые должна выполнять разрабатываемая система: регистрировать вендоров, регистрировать продукты вендоров, редактировать их и удалять.

Для WEB-портала телекоммуникационной компании были разработаны следующие компоненты: база данных WEB-портала, логическая модель данных и в завершении проектирования была построена математическая модель. После проектирования базы данных необходимо было выбрать средство для непосредственной разработки математической модели для WEB-портала. По результатам анализа и сравнения средств разработки был выбран язык программирования Java. После выбора средств разработки было произведено описание основных принципов работы WEB-портала для телекоммуникационной компании.

Разработанная система позволит вендорам размещать своё программное обеспечение в открытом доступе для его дальнейшей продажи. Смоделирована математическая модель, соответствующая требованиям,

которая позволяет не только вендору, но и клиенту получать выгоду, так как появление продуктов на портале стало намного быстрее, чем описывалось ранее.

Реализованный WEB-портал для телекоммуникационной компании имеет широкий функционал для управления со стороны вендоров, управления со стороны менеджеров, покупки ПО клиентами, тем самым обеспечивая сокращение временных затрат на взаимодействие с людьми. Это позволяет увеличить количество продаж вендоров, что приводит к увеличению прибыли телекоммуникационной компании.

Дальнейшее развитие WEB-портала для компании предусматривает увеличение функциональных возможностей, например, поддержку мобильной версии портала. Также предусмотрена модернизация, как и WEB-портала, так и математической модели для телекоммуникационной компании по мере востребованности расширения управления клиентскими отношениями.

Список используемой литературы

1. Бояршинов М. Г. Математическое моделирование в ВУЗе. /Информатика и образование. М.: Логос, 2020. –13 с.
2. Вендров А.М. Один из подходов к выбору средств проектирования баз данных и приложений. М.:СУБД, 2017. №3. -67 с.
3. ВигерсК. Разработка требований к программному обеспечению. М.: Логос, 2018. – 236 с.
4. Гребешков А.Ю. Вычислительная техника, сети и телекоммуникации: Учебное пособие для вузов.М.: ГЛТ, 2016. - 190 с.
5. Грес П.В. Математика для бакалавров. Универсальный курс для студентов гуманитарных направлений: Учебное пособие. М.: Логос, 2015. –288 с.
6. ГОСТР 2.105-2019.Национальный стандарт Российской Федерации. Единая система конструкторской документации. Общие требования к текстовым документам. 2020. – 131 с.
7. Дронов В.А. Разработка современных Web-сайтов. М.: БХВ-Петербург, 2020. –414 с.
8. Дакетт Д. HTML и CSS. Разработка и создание Web-сайтов. М.: Oreilly, 2017. – 55 с.
9. Емельянова С.В. Информационные технологии и вычислительные системы: Обработка информации и анализ данных. Программная инженерия. Математическое моделирование. Прикладные аспекты информатики. М.: Ленанд, 2015. - 104 с.
10. Браун И. Веб-разработка с применением Node и Express. Полноценное использование стека Java. М.: Oreilly, 2017. – 79с.
11. Коберн А.Л. Современные методы описания функциональных требований к системам. М.: Лори, 2019.– 105 с.

- 12.Материал из Википедии — свободной энциклопедии о системах управления сайтом. Электронный ресурс, URL:<http://ru.wikipedia.org/wiki/CMS>.
- 13.Печников В.Н. Создание Web-страниц и Web-сайтов. Самоучитель. М.: Триумф, 2016. –470 с.
- 14.Фримен Э., Робсон Э. Изучаем программирование на Java. М.Ж Oreilly, в перев. ПИТЕР, 2015. – 154 с.
- 15.Химич Р.А. Блог: 5 обязательных технологических партнёров любой телекоммуникационной компании. Электронный ресурс, URL: <http://mediasat.info/2015/09/24/5-must-have-tech-partnerships/>.
- 16.Яргер Р., Риз Дж., Кинг Т. MySQL и mSQL. Базы данных для небольших предприятий и Интернета. СПб: Символ-Плюс, 2015. – 340 с.
- 17.Content management system. Электронный ресурс, URL:<http://www.brutto.ru/informacija/uznat-bolshe/content-management-system>.
- 18.Barker R. CASE*Method. Entity-Relationship Modelling. М.: Oracle Corporation UK Limited, Addison-Wesley Publishing Co., 2020. – 235 p.
- 19.Jesmond A., Chudley O. Smashing UX Design. М.: Smashing, 2020.
- 20.Hogan L. A Project Guide to UX Design: For User Experience Designers in the Field or in the Making. М.: Kew Riders, 2020. – 182 p.
- 21.Dr Arthur Tatnall. Web Portals: The New Gateways to Internet Information and Services М: Idea Group Publishing, 2017. – 235 p.
- 22.Steve Souders. High Performance Web Sites: Essential Knowledge For Front-End Engineers М: O'Reilly Media, 2019. – 75 p.