

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет
имени К.И.Сатпаева


Институт автоматки и информационных технологий

Кафедра «Программная инженерия»

ДОПУЩЕН К ЗАЩИТЕ

Заведующая кафедрой ПИ

канд. физ-мат. наук, профессор

 А.Н. Молдагулова

« 20 » 05 2022 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту

На тему: «Разработка интернет-магазина для компании Gydos»

по специальности 5В070400 – Вычислительная техника и программное
обеспечение

Выполнила


Изтай З. Ж.

Рецензент

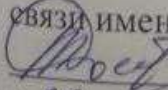
Научный руководитель

доктор PhD, директор института
информационных технологий НАО
Алматинский университет энергетики и
связи имени Г. Даукеева

Ассоциированный профессор

 Мукажанов Н.К.

« 20 » 05 2022 г.

 Досжанова А. А.

« 20 » 05 2022 г.

Алматы 2022

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет
имени К.И.Сатпаева

Институт автоматизации и информационных технологий


Кафедра "Программная инженерия"

5B070400 – Вычислительная техника и программное обеспечение

УТВЕРЖДАЮ

Заведующая кафедрой ПИ

канд. физ-мат. наук, профессор

 А.Н. Молдагулова

« 20 » 05 2022 г.

ЗАДАНИЕ

на выполнение дипломного проекта

Обучающемуся Изтай Зарине Жақайқызы

Тема: *Разработка интернет-магазина для компании Gydos*

Утверждена приказом проректора по академической работе № 429-П/0
от "24" 12 2021 г.

Срок сдачи законченного проекта: "24" 05 2022 г.

Исходные данные к дипломному проекту: *сбор теоретического материала,*
данные анализа по данной теме.

Краткое содержание дипломной работы:

- а) *Разработка классовой модели реляционной базы данных PostgreSQL*
- б) *Реализация многозвенной архитектуры веб-приложения в соответствии с концепцией MVC*
- в) *Проектирование и разработка пользовательского интерфейса*
- г) *Разработка, отладка программного комплекса*

Перечень подлежащих разработке в дипломном проекте вопросов (с точным указанием обязательных чертежей) *представлены 5 слайда презентации.*

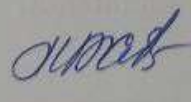

Рекомендуемая основная литература: *из 24 наименований*


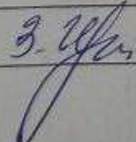
ГРАФИК
подготовки дипломного проекта

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю и консультантам	Примечание
1. Разработка классовой модели реляционной базы данных PostgreSQL	20.01.22	Выполнено
2. Реализация многозвенной архитектуры веб-приложения в соответствии с концепцией MVC	15.02.22	Выполнено
3. Проектирование и разработка пользовательского интерфейса	20.03.22	Выполнено
4. Разработка, отладка программного комплекса	18.04.22	Выполнено

Подписи

консультантов и нормоконтролера на законченный дипломный проект с указанием относящихся к ним разделов проекта

Наименования разделов	Консультанты, И.О.Ф. (уч. степень, звание)	Дата подписания	Подпись
Нормоконтролер	Жекамбаева М.Н. доктор Ph.D., ассоциированный профессор	20.05.22	
Программное обеспечение	Марғұлан К. магистр техн. наук, лектор	18.05.22	

Научный руководитель  Мукажанов Н.К.
 Задание принял к исполнению обучающийся  Изтай З.Ж.
 Дата «17» 11 2021 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет
имени К.И. Сатпаева

Институт автоматизации и информационных технологий

Кафедра «Программная инженерия»

Изтай Зарина Жақайқызы

Разработка интернет-магазина для компании Gydos

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к дипломному проекту

5B070400 – Вычислительная техника и программное обеспечение

Алматы 2022

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет
имени К.И.Сатпаева

Институт автоматизации и информационных технологий

Кафедра «Программная инженерия»

ДОПУЩЕН К ЗАЩИТЕ
Заведующая кафедрой ПИ
канд. физ-мат. наук, профессор
_____ А.Н. Молдагулова
«_____» _____ 2022 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к дипломному проекту

На тему: «Разработка интернет-магазина для компании Gydos»

по специальности 5В070400 – Вычислительная техника и программное
обеспечение

Выполнила

Изтай З. Ж.

Рецензент
доктор PhD, директор института
информационных технологий НАО
Алматинский университет энергетики и
связи имени Г. Даукеева
_____ Досжанова А. А.
«_____» _____ 2022 г.

Научный руководитель
Ассоциированный профессор
_____ Мукажанов Н.К.
«_____» _____ 2022 г.

Алматы 2022

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет
имени К.И.Сатпаева

Институт автоматизации и информационных технологий

Кафедра "Программная инженерия"

5B070400 – Вычислительная техника и программное обеспечение

УТВЕРЖДАЮ

Заведующая кафедрой ПИ
канд. физ-мат. наук, профессор
_____ А.Н. Молдагулова
«_____» _____ 2022 г.

ЗАДАНИЕ

на выполнение дипломного проекта

Обучающемуся Изтай Зарине Жақайқызы

Тема: *Разработка интернет-магазина для компании Gydos*

Утверждена приказом проректора по академической работе № _____
от "___" _____ 2021 г.

Срок сдачи законченного проекта: "___" _____ 2022 г.

Исходные данные к дипломному проекту: *сбор теоретического материала, данные анализа по данной теме.*

Краткое содержание дипломной работы:

- а) Разработка классовой модели реляционной базы данных PostgreSQL*
- б) Реализация многозвенной архитектуры веб-приложения в соответствии с концепцией MVC*
- в) Проектирование и разработка пользовательского интерфейса*
- г) Разработка, отладка программного комплекса*

Перечень подлежащих разработке в дипломном проекте вопросов (с точным указанием обязательных чертежей) *представлены ___ слайда презентации.*

Рекомендуемая основная литература: *из ___ наименований*

ГРАФИК

ПОДГОТОВКИ ДИПЛОМНОГО ПРОЕКТА

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю и консультантам	Примечание
1. Разработка классовой модели реляционной базы данных PostgreSQL	20.01.22	Выполнено
2. Реализация многозвенной архитектуры веб-приложения в соответствии с концепцией MVC	15.02.22	Выполнено
3. Проектирование и разработка пользовательского интерфейса	20.03.22	Выполнено
4. Разработка, отладка программного комплекса	18.04.22	Выполнено

Подписи

консультантов и нормоконтролера на законченный дипломный проект с указанием относящихся к ним разделов проекта

Наименования разделов	Консультанты, И.О.Ф. (уч. степень, звание)	Дата подписания	Подпись
Нормоконтролер	Жекамбаева М.Н. доктор Ph.D., ассоциированный профессор		
Программное обеспечение	Марғұлан Қ. магистр техн. наук, лектор		

Научный руководитель _____ Мукажанов Н.К.

Задание принял к исполнению обучающийся _____ Изтай З.Ж.

Дата

«__» _____ 2021 г.

АННОТАЦИЯ

Данный дипломный проект посвящен разработке программной системы для интернет-магазина для компании «Gydos, ТОО» в виде веб-приложения. Клиенты на сайте могут без регистрации и авторизации добавлять товары в корзину (после регистрации либо входа товары в корзине сохраняются), оплачивать заказ на сайте и узнавать о других услугах компании.

В письме о заказе, доставленном на почту клиента прикреплен QR код. При доставке либо самовывозе, сотрудник отсканировав его, подтвердит личность клиента и выдачу заказа его истинному покупателю.

Приложение направлено на ускорение и автоматизирование процесса принятия и обработки заявок на покупку товаров, гибкую систему учета товаров и на удобство работы продавца. Добавление данных о товарах происходит с помощью импорта из excel-файла.

Для решения был создан проект на базе фреймворка Django на языке программирования Python. Была интегрирована платежная система ePay 2.0 от Nalyk Bank, а так же регистрация и авторизация пользователей с помощью Google.

АҢДАТПА

Бұл дипломдық жоба веб-қосымша түрінде "GYDOS, ЖШС" компаниясы үшін интернет-дүкен әзірлеуге арналған. Сайттағы клиенттер тауарларды тіркеусіз және авторизациясыз себетке қоса алады (тіркелгеннен кейін немесе кіргеннен кейін тауарлар себетте сақталады), сайттағы тапсырыс үшін ақы төлей алады және компанияның басқа қызметтері туралы біле алады.

Клиенттің поштасына жеткізілген тапсырыс туралы хатта QR код бекітілген. Жеткізу немесе алып кету кезінде қызметкер оны сканерлеп, клиенттің жеке басын растайды және оның нақты сатып алушысына тапсырыс береді.

Қосымша тауарларды сатып алуға өтінімдерді қабылдау және өңдеу процесін жеделдетуге және автоматтандыруға, тауарларды есепке алудың икемді жүйесіне және сатушының жұмыс істеу ыңғайлылығына бағытталған. Өнім туралы деректерді қосу excel файлынан импорттау арқылы жүзеге асырылады.

Шешу үшін Python бағдарламалау тілінде Django жақтауының негізінде жоба құрылды. Nalyk Bank-тің ePay 2.0 төлем жүйесі, сондай-ақ Google арқылы пайдаланушыларды тіркеу және авторизациялау біріктірілді.

ANNATATION

This diploma project is dedicated to the development of an online store for the company "Gydos, LLP" in the form of a web application. Customers on the site can add products to the cart without registration and authorization (after registration or login, the goods in the basket will remain), pay for the order on the site and learn about other services of the company.

A QR code is attached to the order letter delivered to the customer's email. Upon delivery or pickup, the employee scans it, confirms the identity of the customer and issues the order to its true buyer.

The application is aimed at speeding up and automating the process of accepting and processing applications for the purchase of goods, a flexible system of accounting for goods and the convenience of the seller. Adding product data is done by importing from an excel file.

For the solution, a project was created based on the Django framework in the Python programming language. The ePay 2.0 payment system from Halyk Bank was integrated, as well as registration and authorization of users using Google.

СОДЕРЖАНИЕ

	Введение	9
1	Исследовательский раздел	10
1.1	Цель разработки	10
1.2	Основное понятие интернет-магазина	10
1.3	Аналоговые системы	12
1.4	Обзор по средам разработки	13
1.4.1	Конструкторы сайтов	13
1.4.2	Разработка программного обеспечения для интернет-магазина с помощью технологий веб-программирования	14
1.4.3	Обзор лучших бэкенд-фреймворков по состоянию на 2022 год.	15
1.5	Значимость и актуальность работы	19
2	Проектирование системы	22
2.1	Определение функциональных требований к системе	22
2.2	Проектирование диаграммы базы данных и интерфейса	23
3	Разработка интернет-магазина	25
3.1	Описание среды разработки	25
3.2	Разработка системы	25
	Заключение	36
	Список использованной литературы	37
	Приложение А. Техническое задание	39
	Приложение Б. Текст программы	40

ВВЕДЕНИЕ

За последние несколько лет электронная коммерция стала неотъемлемой частью глобальной розничной торговли. Как и многие другие отрасли, розничная торговля претерпела существенные изменения после появления Интернета, и благодаря продолжающейся цифровизации современной жизни потребители практически из каждой страны теперь получают выгоду от онлайн-покупок. Благодаря имеющемуся свободному доступу к Интернету число онлайн покупателей продолжает расти с каждым годом. В 2021 году более двух миллиардов человек приобрели товары или услуги онлайн, и в течение того же года объем продаж электронной розничной торговли превысил 4,2 триллиона долларов США по всему миру.

По статистике, пандемия COVID-19 сильно повлияла на глобальные доходы от электронной коммерции, добавив дополнительный рост продаж на 19% в 2020 году и дополнительный рост продаж на 22% к темпу роста продаж 2021 года, увеличение спрогнозированных темпов на 9% и 12% соответственно. Прогнозируется, что до 2024 года рейтинги мировых доходов от электронной коммерции возрастут минимум на 8%. Это показывает, что рост продаж в Интернете будет еще увеличиваться. Тема электронной коммерции будет актуальна всегда и развитию данной отрасли стоит уделять большое внимание.

Данный дипломный проект посвящен разработке программной системы интернет-магазина для компании «Gydos, ТОО» в виде веб-приложения. Компания занимается автоматизацией производственных процессов и продажей технического оборудования с 2016 года, занимает лидирующие позиции на рынке. В связи с расширением бизнеса и увеличением базы клиентов, компания столкнулась необходимостью автоматизации процессов приема заявок на покупку оборудования и гибкой системы учета товаров. Разработка интернет-магазина приведет к увеличению товарооборота и создаст удобства для клиентов компании.

В данной работе будут описаны аспекты успешной реализации интернет-магазина, способы его создания, сравнение способов создания, а также описание реализации интернет-магазина для компании «Gydos, ТОО».

1 Исследовательский раздел

1.1 Цель разработки

Цель дипломной работы – разработка интернет-магазина для компании Gydos.

Задачи дипломной работы:

- исследование рынка подобных продуктов;
- сбор необходимой информации для реализации проекта;
- выбор средств разработки;
- проектирование архитектуры базы данных;
- проектирование пользовательского интерфейса;
- написание кода;
- тестирование.

1.2 Основное понятие интернет-магазина

Интернет-магазин - веб-сайт или приложение для электронной коммерции, где покупатели могут видеть каталог товаров или услуг и приобретать их в электронном виде.

В 2021 году почти 20% мировых розничных продаж происходило онлайн. Начало создания интернет-магазина включает в себя формирование бизнес-плана, поиск платформы электронной коммерции либо способов его разработки и создание маркетинговой стратегии.

Хотя многие могут приравнять интернет-магазин к электронной коммерции, есть разница. Первый имеет цифровой каталог, который покупатели могут просматривать. Обычно для заказа есть корзина или список. Существует процесс оформления заказа, который включает в себя электронный способ оплаты, такой как кредитная карта или платежные системы, такие как PayPal. В то время как электронный способ оплаты является необходимой частью электронной коммерции, наличие цифрового каталога и процесса оформления заказа - нет.

Существует несколько важных компонентов сайта для онлайн продаж, обеспечивающие удобство клиенту и успех сайта:

1 Четкий логотип компании на сайте.

Запоминающийся и яркий логотип всегда привлекает пользователей. Логотип должен быть легко идентифицирован среди других логотипов. Это символ доверия компании или организации. Логотип представляет компанию, многие клиенты определяют свои любимые бренды по изображению логотипа. Он должен отражать идеи и продукты компании.

2 Акции и предложения.

Для привлечения внимания клиента на главной странице отображаются промо-акции и предложения. Навигация по странице должна быть легкой, а поиск товара - быстрым.

3 При размещении товаров, давать приоритет пользующимся наибольшим спросом.

При разработке сайта для электронной коммерции важно учитывать спрос товаров и наиболее популярные товары размещать выше остальных.

4 Наличие быстрой регистрации и корзины.

Для разработки интернет-магазина необходимо иметь опцию регистрации и корзины на сайте. Такой важный функционал должен быть интуитивно понятен при использовании и располагаться в легком доступе для пользователя. Наличие функции окна поиска позволяет клиенту искать конкретный продукт и напрямую переходить к описанию продукта. Значки входа в систему, поиска и корзины лучше разместить рядом.

5 Предоставление нескольких доступных вариантов оплаты.

Большая часть онлайн-покупок имеет возможность онлайн-платежей. Онлайн-платежи безопасны и облегчают покупки клиентам. Необходимо отобразить список карт, платежи с которых обрабатываются в подключенной платежной системе сайта в нижнем колонтитуле сайта или на странице оплаты.

6 Ссылки на социальные сети компании.

Большинство людей просматривают сайты социальных сетей, чтобы получить отзывы и узнать больше о компании и продуктах. Социальные сети также могут использоваться в качестве рекламы, с возможностью ознакомления с подробностями об услугах или продуктах. Рекламные мероприятия также можно проводить на сайтах популярных площадок и отправлять уведомления основным или постоянным клиентам.

7 Номера телефонов и параметры онлайн-чата:

Контактные данные компании также необходимо разместить на главной странице, что помогает клиенту связаться с вами во время неудобств, связанных с сайтом или доставкой. Лучший вариант – реализация онлайн-чата на сайте, что обеспечит легкую коммуникацию сотрудников с покупателями.

8 Обеспечение гарантии безопасности в вопросах о платежах.

Что касается покупок в Интернете, все платежи осуществляются через онлайн-транзакции. Поэтому необходимо предоставить клиенту гарантию того, что транзакции безопасны от посторонних глаз. Сертификаты аккредитации, такие как Network Solutions, McAfee дадут клиентам чувство безопасности во время платежей.

Существуют различные типы веб-сайтов для электронной коммерции:

1 Сайт индивидуального бренда - это сайт, который продает только продукты или услуги одного бизнеса или человека. Это самый распространенный тип веб-сайта.

2 Интернет-магазины.

Этот тип веб-сайта электронной коммерции распространен для крупных ритейлеров. Интернет-магазины - это более крупные магазины, которые обычно продают оптом продукты других брендов.

3 Партнерские сайты

Партнерский сайт - это место, где могут продаваться продукты другого бренда в обмен на комиссию.

4 Рынки

Другой тип веб-сайта электронной коммерции - это торговая площадка. Торговая площадка - это веб-сайт, на котором индивидуальные предприниматели могут выставлять свои товары на продажу. Наиболее популярные примеры включают Amazon и Etsy. Etsy и Amazon предоставляют поставщикам онлайн-рынок для перечисления и продажи своих продуктов.

1.3 Аналоговые системы

В результате проведенного анализа мирового рынка сервисов по предоставлению Интернет-торговли экспертом в области интернет-маркетинга, автором статей в журналах Forbes, NY times о популярных продуктах и онлайн-сервисах Кристин Уильямс, был опубликован рейтинг лучших онлайн-магазинов в мире по состоянию на 28 апреля 2022 года. Лидирующую позицию занял интернет-магазин американской компании Amazon.

1. Amazon.com
2. Walmart
3. Ebay.com
4. Alibaba.com
5. Target.com
6. Bestbuy.com
7. Flipkart.com
<8. Shop.comstrong>
9. Groupon.com
10. Ikea.com

Рисунок-1.1 – Список 10 лучших сайтов онлайн-покупок в мире 2022

В ходе анализа используемых технологий таким гигантом как Amazon, выяснилось, что у Amazon есть несколько команд разработки и нет единого технологического стека. Для серверной части большинством команд используется язык программирования Java. Для хранения данных в качестве системы управления базами данных используются MySQL и DynamoDB. Для интерфейса используются фреймворки AngularJS и GWT. Таким образом, стандартного стека не существует.

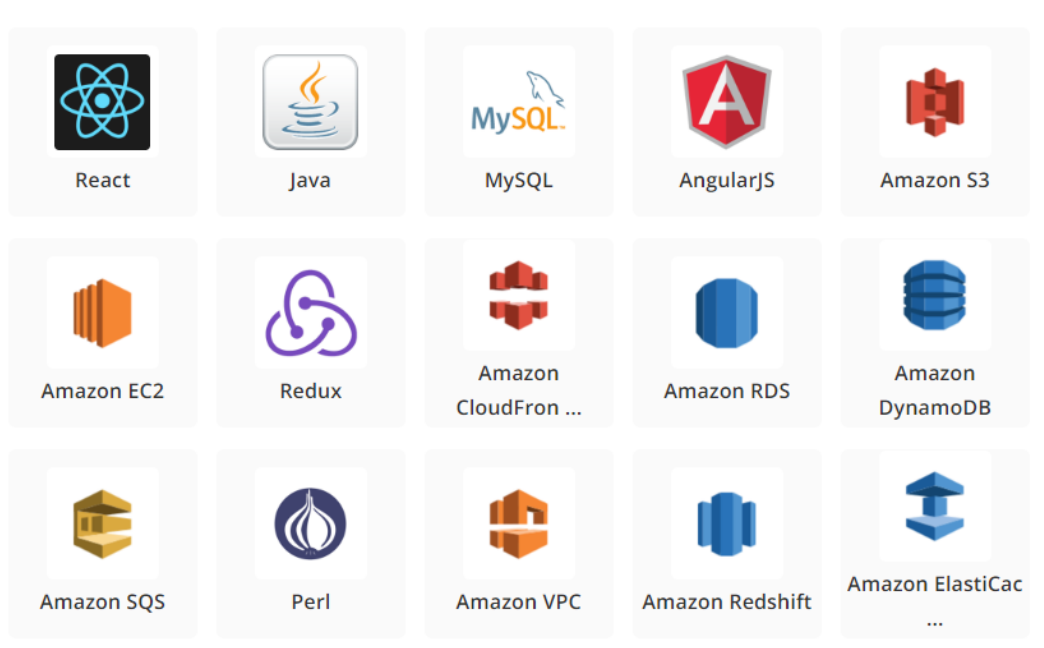


Рисунок-1.2 – Технологии, используемые в разработке интернет-магазина компании Amazon

В 2021 году согласно рейтингу TOP KZ Retail E-Commerce 2021, Kaspi.kz (Каспи Магазин) признан № 1 в электронной коммерции в Казахстане, тем самым в четвертый раз занимая первое место в рейтинге крупнейших казахстанских торговых интернет-площадок. Для разработки серверной части в качестве языка программирования используется Java, для хранения данных используется система управления базами данных Oracle.

1.4 Обзор по средам разработки

1.4.1 Конструкторы сайтов

Существует два наиболее популярных подхода для разработки собственного сайта - воспользоваться конструктором сайтов либо создать сайт, изучив веб-программирование.

Раньше разработка сайта всегда требовала много времени и усилий, огромного запаса креативности и обширных знаний в области кодирования. Такие сайты, как Weebly, Squarespace, Weblum и Wix, предлагают платформы, на которых пользователи могут создавать свои собственные сайты бесплатно или по низкой цене.

Конструктор сайтов – это быстрый, простой способ создать интернет-магазин с нуля. Проанализировав информацию, я выявила плюсы и минусы данного подхода.

Преимущества:

- подход не требует навыков в программировании, что подходит большому количеству людей;
- быстрое и дешевое решение;
- легкая поставка сайта;
- не требует навыков в дизайне.

Недостатки:

1 Не подходит для среднего и большого бизнеса. Бизнес-сайты должны иметь качественный дизайн, SEO-оптимизацию и профессиональные функции. Согласно исследованию, 48% людей считают, что дизайн сайта является одним из важнейших факторов в принятии решения о доверии сайту.

2 Шаблоны в конструкторах не разработаны на заказ. Бесплатные шаблоны выглядят универсальными, однако могут не работать на мобильных устройствах и не имеют дополнительных функций SEO-оптимизации.

3 Медленное время загрузки. Сайты, построенные с использованием конструкторов, загружаются слишком медленно по сравнению с сайтами разработанными специалистами.

4 Ограниченные возможности добавления нового функционала.

5 Большинство конструкторов сайтов основаны на Flash, что делает невозможным нахождение сайта в некоторых поисковых системах.

Конструктор сайтов, возможно является лучшим способом только для создания временного сайта либо сайта для личного хобби.

1.4.2 Разработка программного обеспечения для интернет-магазина с помощью технологий веб-программирования

Интерфейс и серверная часть являются наиболее важными условиями разработки программного обеспечения. Серверные технологии отвечают за серверную часть, в то время как интерфейсные инструменты относятся к клиентской части приложения. Обе стороны должны работать безупречно, чтобы программное решение было эффективным и гибким.

В то время как методы разработки программного обеспечения сместились в сторону пользователя и, как правило, удовлетворяют потребности и визуальные предпочтения заказчика, серверная часть – это то, что на самом деле

подпитывает программное обеспечение. Без превосходных серверных технологий веб- и мобильным приложениям будет не хватать масштабируемости, функциональности и быстродействия. Серверная часть - это так называемое “закулисье” программного решения, которое фокусируется на архитектуре приложения и действует как мост между интерфейсом и базой данных.

Программный фреймворк - это основа, на которой разработчики могут создавать приложения более быстрым и стандартизированным способом. Согласно Википедии, основная цель фреймворка - автоматизировать накладные расходы, связанные с разработкой программного обеспечения.

Основными преимуществами использования фреймворка для разработки являются:

- экономия времени;
- масштабируемость;
- надежность;
- безопасность;
- интеграция.

1.4.3 Обзор лучших бэкенд-фреймворков по состоянию на 2022 год

Django

Django - ведущий бэкенд-фреймворк с открытым исходным кодом, основанный на языке программирования Python. Он следует шаблону model view controller (MVC).

Django подходит для разработки сложных и многофункциональных веб-сайтов, управляемых базами данных, и это один из самых простых бэкенд-фреймворков. Django считается одним из лучших бэкенд-фреймворков для веб-разработки.

Этот бэкенд-фреймворк обеспечивает оптимальную подключаемость, меньшее количество кодирования, большее повторное использование и более быструю разработку. Он использует Python для всех операций и предоставляет дополнительный интерфейс администратора для создания, чтения, обновления и удаления операций. Django используется многими известными веб-сайтами, такими как Disqus, Mozilla и Washington Times.

Преимущества Django:

1 Fast framework - Django прост в использовании и в обучении. Что помогает разработчикам ускорить весь процесс разработки от начала до конца.

2 Многофункциональность – Django предоставляет широкий спектр функций, чтобы помочь пользователям позаботиться о некоторых общих требованиях к веб-разработке. Он помогает в таких задачах, как аутентификация пользователей, карты сайтов, администрирование контента и многое другое.

3 Оптимальная безопасность – Django - это безопасная платформа, которая помогает своим пользователям предотвратить несколько проблем безопасности, включая межсайтовые скрипты, clickjacking, SQL-инъекцию и подделку запросов. Он предоставляет систему аутентификации пользователей, позволяющую пользователям безопасно хранить и управлять паролями и учетными записями.

4 Высокая масштабируемость – Django предлагает своим пользователям высокий уровень масштабируемости.

5 Универсальный фреймворк – Django - это фреймворк, который можно использовать для разработки широкого спектра типов приложений. Некоторые из них включают приложения для социальных сетей, системы управления контентом и вычислительные платформы.

Особенности Django:

1 Open source – Django - это фреймворк с открытым исходным кодом, используемый для веб-приложений на основе Python. Он прост, удобен в использовании и надежен.

2 Система именования – Django имеет собственную систему именования для всех инструментов и функций. Кроме того, он также имеет простую в использовании панель администратора по сравнению с Yii и Lavarel.

Некоторые из ключевых функций Django включают простой синтаксис, архитектуру MVC, ORM (Object Relational Mapper) и библиотеки HTTP. Django также имеет свой собственный веб-сервер, фреймворк для модульного тестирования Python и компоненты, необходимые для решения некоторых случаев.

Следует использовать Django для веб-разработки в следующих случаях:

- для разработки веб-приложения или бэкенда API;
- для быстрой разработки веб-приложений;
- быстрое развертывание приложения и масштабирование его в соответствии с потребностями бизнеса;
- ORM для работы с базами данных вместо запросов к базе данных;
- разработать безопасное одностраничное приложение для извлечения данных или публикации данных.

Компании, использующие Django:

- Instagram;
- Disqus;
- Pinterest;
- Mozilla Firefox;
- Spotify;
- YouTube.

Laravel

Laravel - это веб-фреймворк на языке программирования PHP с открытым исходным кодом для разработки веб-приложений, которые следуют архитектуре model-view-controller (MVC). Laravel оценивается как один из лучших веб-

фреймворков, и некоторые эксперты считают эту технологию лучшим программным обеспечением для бэкенд-разработки.

Laravel также предоставляет своим пользователям несколько способов доступа к реляционным базам данных, а также утилиты обслуживания и развертывания приложений. Laravel имеет лицензию MIT и имеет исходный код, размещенный на GitHub.

Преимущества Laravel

1 Аутентификация: Реализация аутентификации довольно проста с Laravel, так как она предлагает легкую настройку. Laravel облегчает простую организацию логики авторизации и легкий контроль ресурсов доступа.

2 Простой API: Laravel предлагает простой API, который работает с библиотекой SwiftMailer. Laravel предлагает драйверы для Mandrill, SMTP, Mailgun, Amazon SES и SparkPost. Он также имеет драйверы для PHP mail и отправляет почту. Laravel позволяет быстро отправлять почту приложениям через облачный или локальный сервис. Он также обеспечивает поддержку отправки уведомлений по нескольким каналам доставки.

3 Кэш-бэкенды: Laravel предлагает поддержку Redis, Memcached и других широко используемых кэш-бэкендов. Он использует драйвер файлового кэша, который выполняет хранение кэшированных объектов в файловой системе. Более крупные приложения используют кэш в памяти, такой как APC или Memcached. Laravel также позволяет пользователям настраивать несколько конфигураций кэша.

4 Журналы: проект Laravel имеет предварительно настроенную обработку исключений и ошибок. Laravel также имеет интеграцию библиотеки Monolog logging и предлагает поддержку нескольких обработчиков журналов.

5 Тестирование: Laravel удобен для тестирования и предлагает поддержку PHPUnit наряду с готовым к использованию phpunit.xml файл для приложений. Фреймворк поставляется с вспомогательными методами для выразительного тестирования приложений. Laravel также обеспечивает простое моделирование поведения пользователя для таких действий, как заполнение форм, переходы по ссылкам и запросы приложений.

Особенности Laravel

Механизм шаблонов: Фреймворк имеет легкие встроенные шаблоны, которые можно использовать для создания макетов и заполнения контента. Он также предоставляет виджеты с кодом JS и CSS. Шаблоны предназначены для разработки как простых, так и сложных секционных макетов.

Поддержка архитектуры MVC: Laravel предлагает поддержку шаблонов архитектуры MVC для эффективного разделения уровней презентации и бизнес-логики. Laravel MVC предлагает множество функций, повышает производительность и повышает масштабируемость и безопасность.

Eloquent Object Relational Mapping: пользователи могут использовать Eloquent Object Relational Mapping (ORM), включая простую реализацию активной записи PHP. ORM позволяет разработчикам приложений создавать

запросы к базе данных с использованием синтаксиса PHP без написания кода SQL. ORM сравнительно быстрее, чем другие PHP-фреймворки.

Безопасность: Laravel framework обеспечивает надежную безопасность веб-приложений с использованием хэшированных паролей. В результате пароли не хранятся в текстовом формате в базах данных. Алгоритм хеширования bcrypt также используется Laravel для генерации зашифрованных паролей. Эта платформа разработки использует подготовленные операторы SQL для уменьшения вероятности инъекционных атак.

Ruby on Rails

Ruby on Rails, также известный как Rails, представляет собой серверную платформу веб-приложений на основе Ruby с лицензией MIT. Rails - это фреймворк MVC, предлагающий структуры базы данных по умолчанию, веб-страницы и веб-службы. Ruby on Rails является одним из лучших фреймворков для веб-разработки и считается некоторыми разработчиками лучшим веб-фреймворком производительности.

Ruby on Rails продвигает использование веб-стандартов, таких как XML или JSON для передачи данных и CSS, JavaScript и HTML для взаимодействия. Rails уделяет приоритетное внимание использованию шаблонов разработки программного обеспечения, таких как active record pattern, convention over configuration (CoC) и don't repeat yourself (DRY).

Преимущества Ruby on Rails

1 Экономия времени - инструмент предлагает несколько готовых модулей и плагинов, которые позволяют разработчикам сократить время, необходимое для создания шаблонного кода. Это позволяет командам разработчиков минимизировать время сборки приложений по сравнению со случаями, когда они используют другие фреймворки.

2 Согласованность - Ruby on Rails предоставляет разработчикам возможность использовать стандартизированные системы хранения файлов и соглашения о программировании. Это помогает поддерживать структурную согласованность и читабельность проекта, экономя при этом значительное количество времени.

3 Экономичный - фреймворк с открытым исходным кодом и предлагается под лицензией MIT. Это означает, что пользователям не нужно ничего платить, чтобы получить фреймворк. С помощью Gems пользователи могут добавлять функции, не разрабатывая их с нуля. В результате они могут сэкономить много времени и усилий.

4 Качественная разработка - позволяет пользователям разрабатывать программное обеспечение без ошибок. Minitest инструмент в ядре Rails предоставляет пользователям множество функций тестирования, таких как mocking, синтаксис ожидания и тестовый бенчмаркинг. Ruby on Rails больше всего подходит для развертывания на основе поведения (BDD) и развертывания на основе тестирования (TDD). Оба этих подхода к разработке полезны для снижения затрат на разработку.

5 Масштабируемость - Ruby on Rails - это бэкэнд, который предлагает пользователям большую масштабируемость. Он более подходит, чем многие его альтернативы для веб-сайтов, которые ориентированы на обслуживание большого количества клиентов. Масштабируемость - одна из причин, по которой многие пользователи выбирают Ruby on Rails.

Особенности Ruby on Rails

1 Архитектура MVC – Ruby on Rails основана на MVC, одной из наиболее широко используемых архитектур веб-приложений во всем мире. Разработчики, которые уже имеют опыт работы с шаблонами MVC, могут быстро начать использовать Ruby on Rails. Эта архитектура разделяет коды по их функциям, то есть по уровню данных, уровню представления и поддержанию уровня ресурсов.

2 Active Record – Ruby on Rails опирается на библиотеку, известную как active record, что позволяет разработчикам легко выполнять проектирование запросов взаимодействия с базой данных. Запросы пишутся на языке программирования Ruby и преобразуются в SQL-запросы, которые получают выходные данные и возвращают объекты. Библиотека активных записей Ruby on Rails вполне способна, и потребность в разработке запроса на SQL снижается.

3 Convention over Configuration – файлы конфигурации не используются Ruby on Rails, поскольку он предоставляет динамические расширения среды выполнения, отражения и соглашения. Он следует подходу “Соглашение над конфигурацией” для автоматического присвоения значения без вмешательства пользователя. Несколько файлов конфигурации используются программными системами, такими как Java web application framework, причем каждый из них имеет несколько настроек. Соглашение повышает производительность, поскольку пользователям не требуется назначать время для настройки файла конфигурации.

4 Простое тестирование - Ruby on Rails предлагает RSpec, простую в использовании установку модульного тестирования.

1.5 Значимость и актуальность работы

Цель данного проекта - разработка интернет-магазина для компании GYDOS. Компания занимается автоматизацией производственных процессов и продажей технического оборудования с 2016 года, занимает лидирующие позиции на рынке.

Проект решает проблемы необходимости автоматизации процессов приема заявок на покупку оборудования и гибкой системы учета товаров, сложности увеличения товарооборота с которой сейчас сталкивается компания. После завершения проекта, готовое веб-приложение будет весомым вкладом в развитие бизнеса, оптимизацию работы сотрудников и в удобство клиентов.

За последние несколько лет электронная коммерция стала неотъемлемой частью глобальной розничной торговли. Как и многие другие отрасли, розничная

торговля претерпела существенные изменения после появления Интернета, и благодаря продолжающейся цифровизации современной жизни потребители практически из каждой страны теперь получают выгоду от онлайн-покупок. Благодаря имеющемуся свободному доступу к Интернету число онлайн покупателей продолжает расти с каждым годом. В 2021 году более двух миллиардов человек приобрели товары или услуги онлайн, и в течение того же года объем продаж электронной розничной торговли превысил 4,2 триллиона долларов США по всему миру.

По статистике (см. Рисунок-1.3), пандемия COVID-19 сильно повлияла на глобальные доходы от электронной коммерции, добавив дополнительный рост продаж на 19% в 2020 году и дополнительный рост продаж на 22% к темпу роста продаж 2021 года, увеличение спрогнозированных темпов на 9% и 12% соответственно.

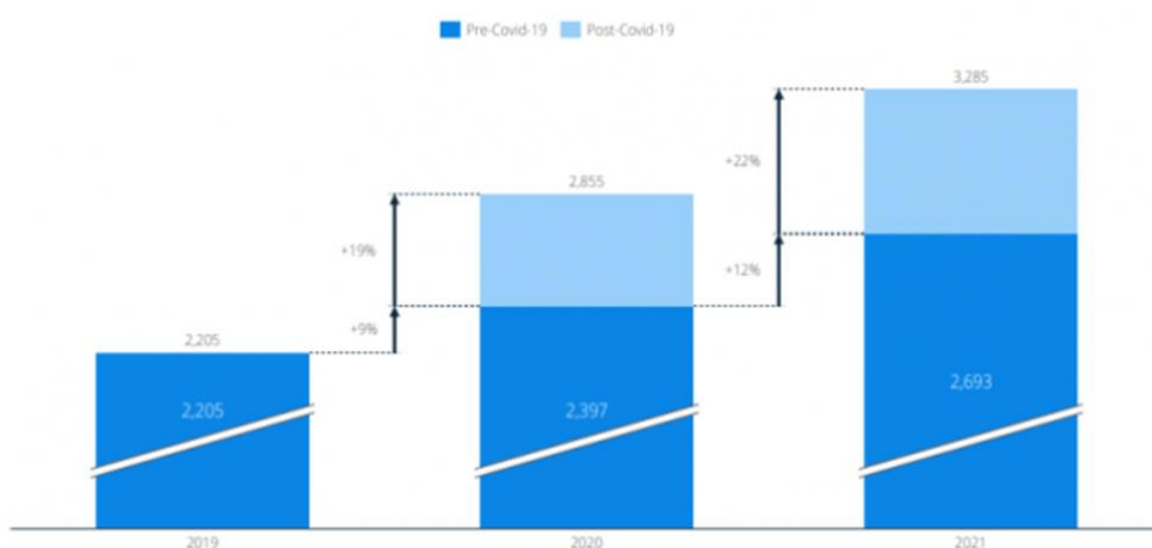


Рисунок-1.3 – Рост продаж в электронной коммерции за 2019, 2020 и 2021 гг.

Прогнозируется, что до 2024 года рейтинги мировых доходов от электронной коммерции возрастут минимум на 8%. Это показывает, что рост продаж в Интернете будет еще увеличиваться. Тема электронной коммерции будет актуальна всегда и развитию данной отрасли стоит уделять большое внимание (см. Рисунок 1.4).

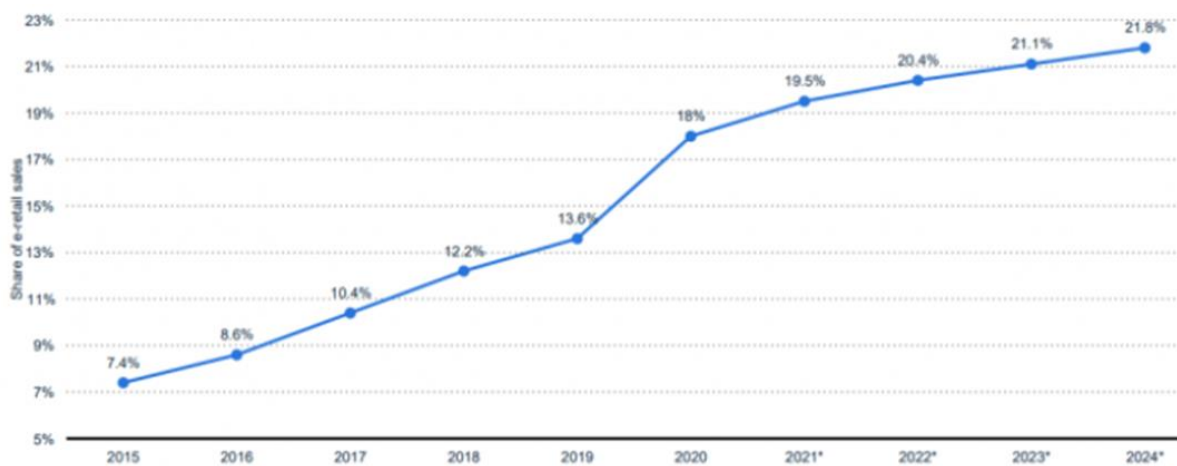


Рисунок-1.4 – Доля электронной коммерции в общем объеме мировых розничных продаж с 2015 года с прогнозом до 2024

2 Проектирование системы

Для достижения поставленной цели необходимо решить следующие задачи:

- определение функциональных требований к системе;
- проектирование системы и базы данных;
- разработка системы;
- тестирование системы.

2.1 Определение функциональных требований к системе

Функциональные требования в разработке программного обеспечения помогают определить предполагаемое поведение системы, описывают функциональность, которую система будет предоставлять её пользователям.

Информационная система интернет-магазина предоставляет следующий функционал:

1 регистрацию и авторизацию пользователей. Регистрация производится в двух вариантах: вход с помощью Google и регистрация с почтой в качестве идентификатора, паролем и верификацией аккаунта через почту. Войти можно будет так же с помощью Google, либо введя почту в качестве логина и пароль;

2 возможность восстановления пароля.

Для авторизованных и неавторизованных пользователей:

1 просмотр главной страницы с товарами, новостями от компании;

2 просмотр страницы с маркетинговой информацией о компании;

3 страница с контактными данными, адресом компании с возможностью оставить отзыв/письмо в форме;

4 просмотр страницы товара;

5 добавление товара в корзину;

6 удаление товара(-ов) из корзины;

7 возможность оплатить корзину;

8 рассылка уведомления о новом заказе сотруднику;

9 генерация QR кода для клиента, отсканировав который, сотрудник моментально получит информацию о клиенте, его заказах и т.д.;

10 возможность отмены заказа и возврата денег клиенту.

2.2 Проектирование диаграммы базы данных и интерфейса

Для отображения связей между таблицами в базе данных используется Диаграмма отношений сущностей Entity-Relationship model (ER-модель).

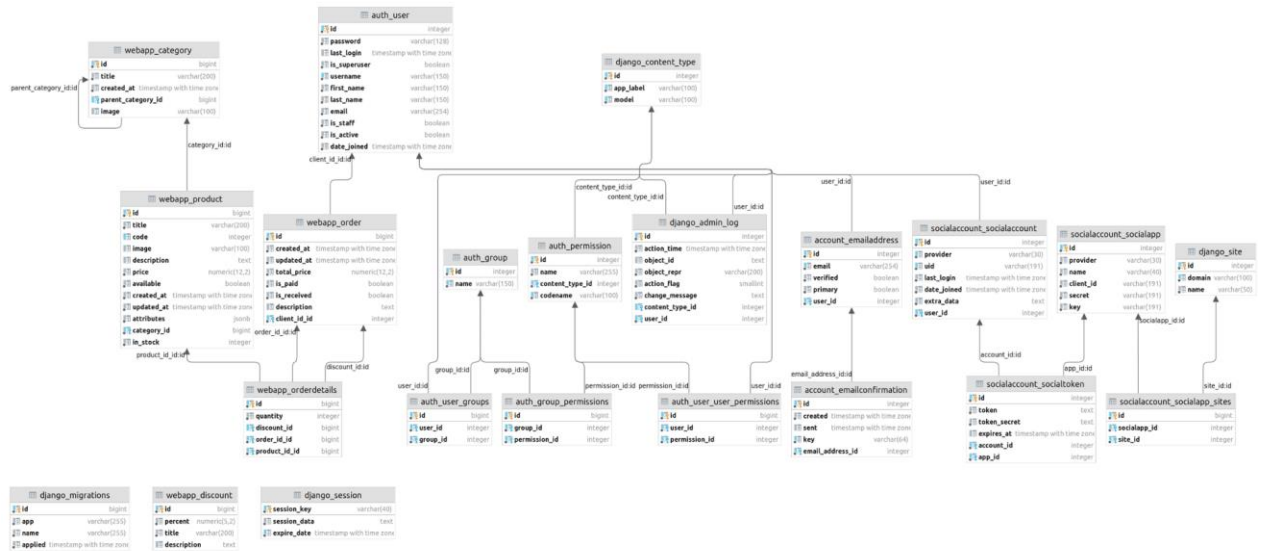


Рисунок-2.1 – Диаграмма отношений сущностей в базе данных

Существует несколько основных сущностей:

1 Продукт:

- уникальный номер записи;
- название продукта;
- фотография продукта;
- текстовое описание продукта;
- цена;
- дата создания;
- дата изменения записи;
- количество;
- код товара;
- уникальные характеристики (поле JSON);
- актуален ли продукт (булево поле).

2 Заказ:

- номер заказа;
- идентификатор клиента;
- дата создания заказа;
- оплачен ли заказ;
- получен ли заказ;
- общая сумма заказа.

3 Таблица с деталями заказов:

- номер заказа;

- номер продукта;
- количество;
- размер скидки;
- цена за штуку.

4 Категории товаров:

- уникальный номер;
- дата создания категории;
- название;
- номер родительской категории (если есть);
- изображение.

Важно, чтобы в Sitemap диаграмме, не была глубокая вложенность. Так как в ином случае пользователю становится сложнее ориентироваться на сайте.

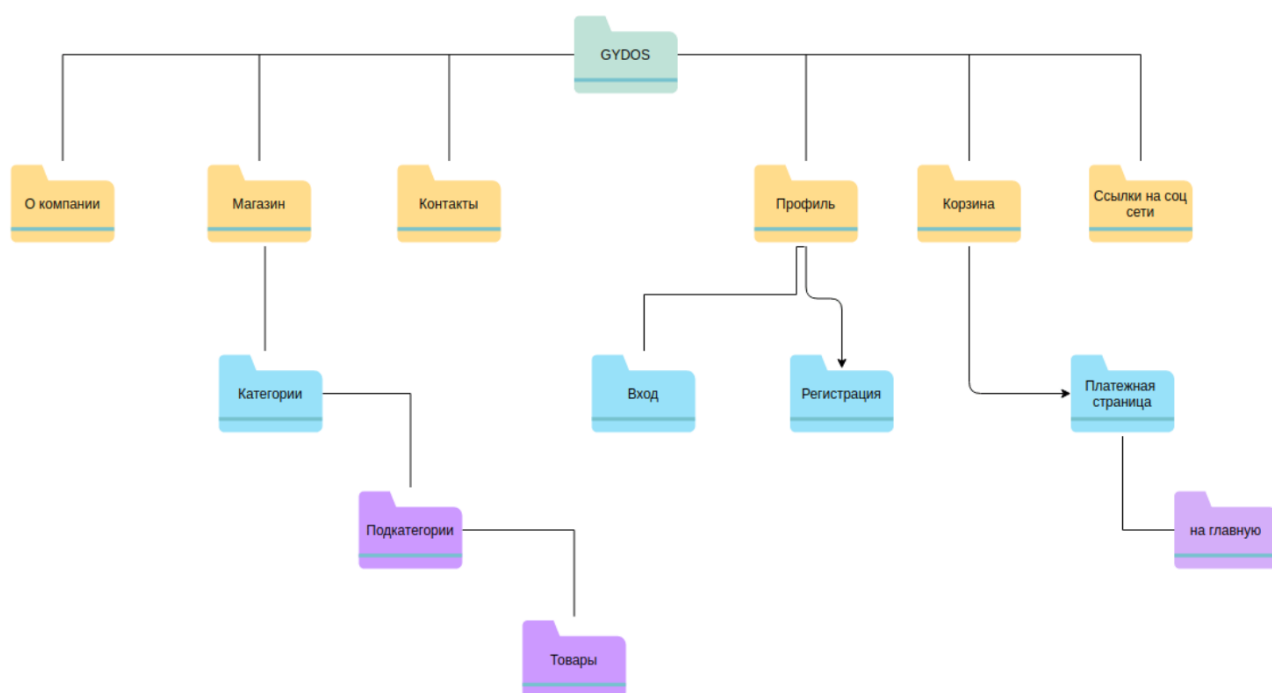


Рисунок-2.2 – Sitemap приложения

3 Разработка интернет-магазина

3.1 Описание среды разработки

Проанализировав несколько бэкенд фреймворков, а также характеристики и потребности разрабатываемого в дипломном проекте приложения, я сделала вывод о том, что фреймворк Django подходит лучше всех.

Причины выбора Django:

Django соответствует следующим требованиям проекта:

- 1 быстрая разработка в рамках дипломного проекта;
- 2 возможность масштабируемости приложения в будущем;
- 3 обеспечение высокой безопасности, так как в веб-приложении будут совершаться платежи;
- 4 удобство разработки за счет многофункциональности;
- 5 имеет встроенный интерфейс администратора, который упрощает работу с ним.

По умолчанию фреймворк Django использует легковесную базу данных sqlite3, но ее функционал не отвечает всем требованиям масштабирования проекта. В связи с этим, проанализировав достоинства и недостатки таких СУБД как Oracle, MySQL, PostgreSQL, мой выбор пал на последний вариант.

3.2 Разработка системы

При первоначальной загрузке сайта, пользователь попадает на главную страницу. В шапке сайта располагаются кнопки логотипа, который ведет на главную страницу, меню, откуда пользователь может попасть в магазин, на страницу просмотра сведений о компании, ее опыте работы на рынке, достижениях, услугах и т.д.

Пользователь, нажав на кнопку «Контакты» может перейти на страницу с контактными данными компании, а также там же заполнив форму отправить письмо компании. Кнопки входа/выхода, регистрации, перехода в корзину и поиска удобно расположены рядом, в правом углу.

На главной странице могут располагаться различные маркетинговые решения компании (см. Рисунок-3.1).

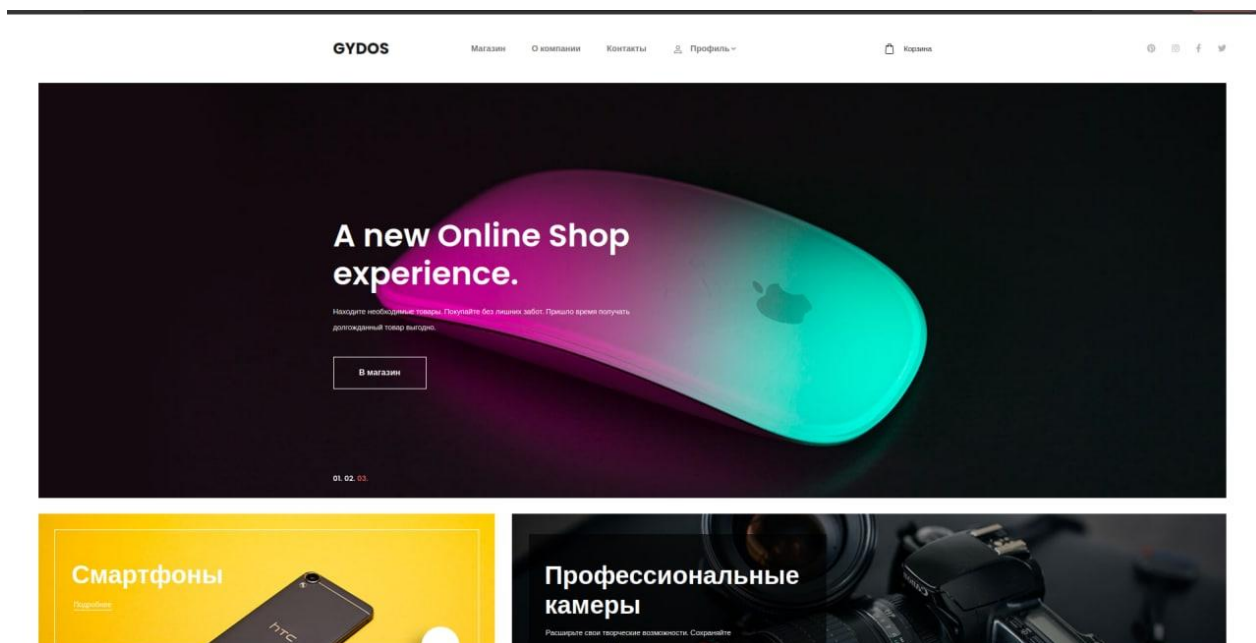


Рисунок-3.1 – Главная страница сайта

При прокручивании вниз, на главной странице можно посмотреть доступные категории товаров из интернет-магазина (см. Рисунок-3.2).

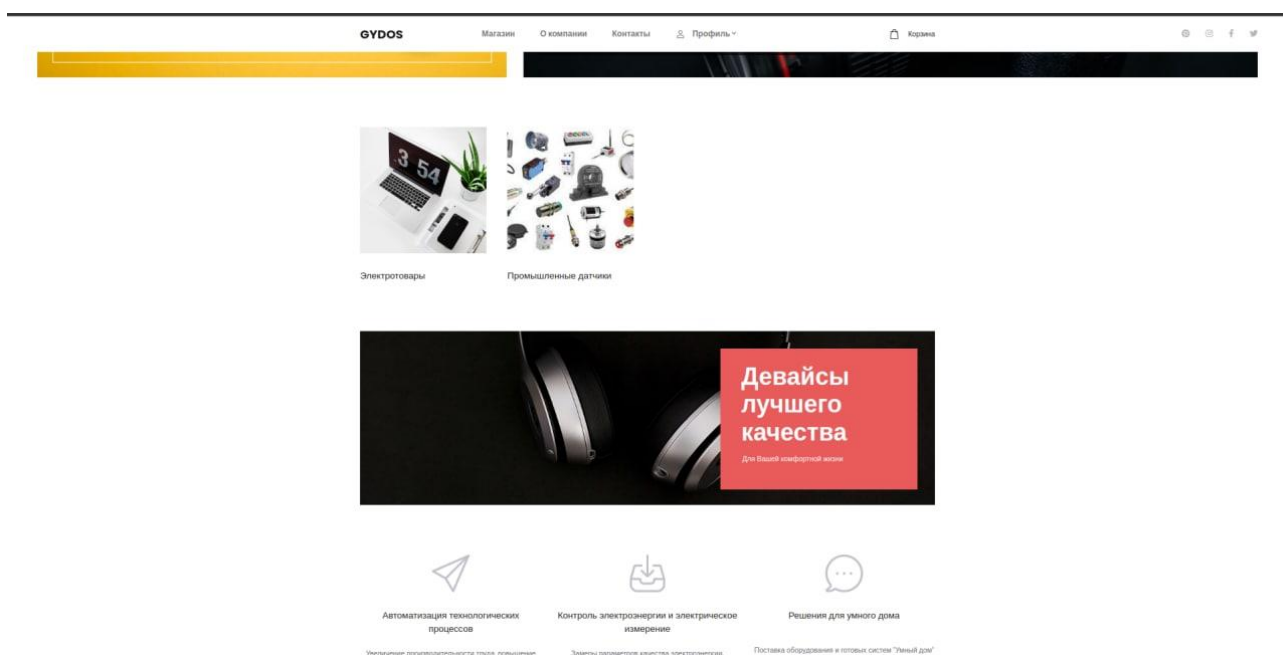


Рисунок-3.2 – Нижняя часть главной страницы

При нажатии на кнопку «Магазин» в навигации сайта, пользователь перенаправляется на страницу просмотра всех товаров (см. Рисунок-3.3).

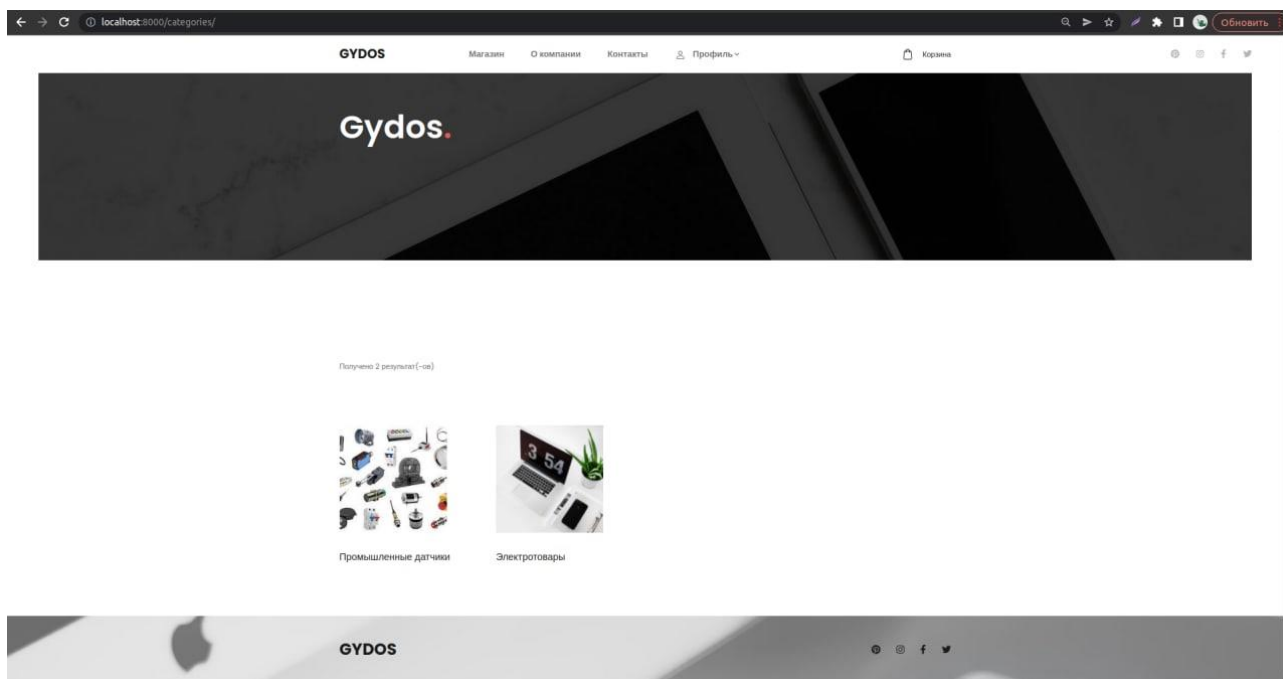


Рисунок-3.3 – Страница интернет-магазина

Реализована страница «О компании», с помощью которой клиент может подробнее ознакомиться с работой и историей компании (см. Рисунок-3.4).

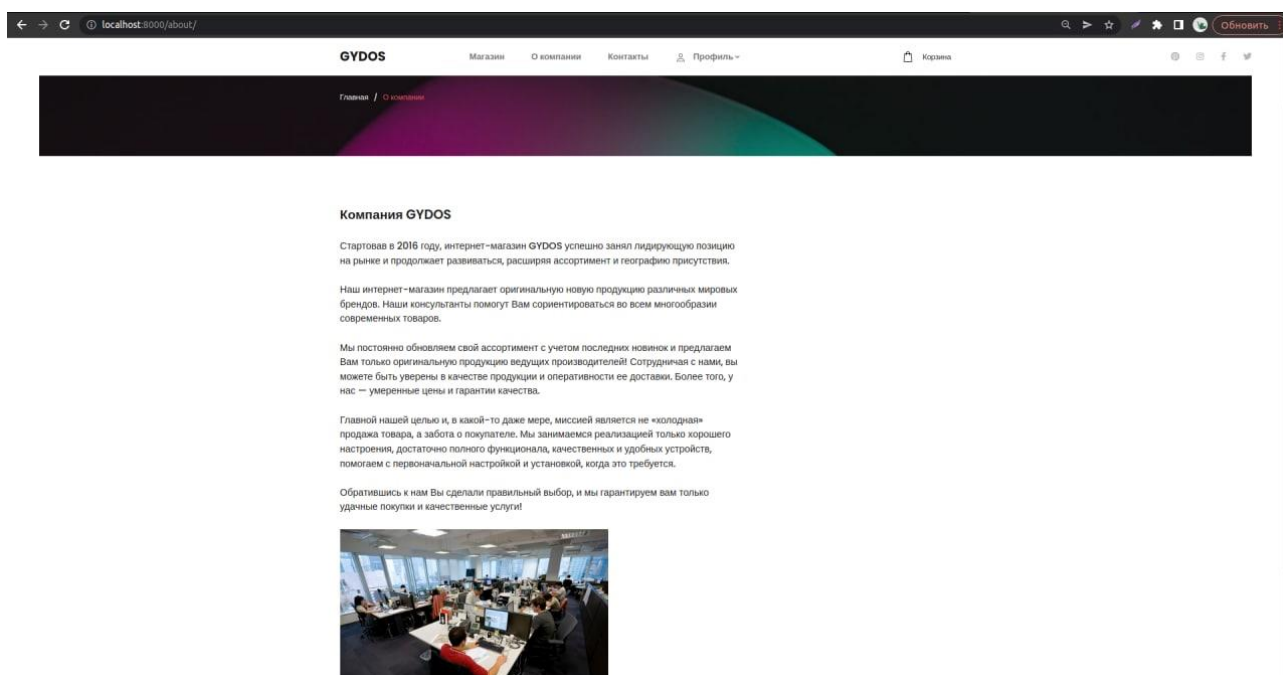


Рисунок-3.4 – Страница с подробной информацией о работе компании

Страница «Контакты» предоставляет клиенту контактные данные компании, а также удобную навигационную карту (см. Рисунок-3.6). Тут пользователь может оставить письмо с отзывом либо с пожеланиями (см. Рисунок-3.5).

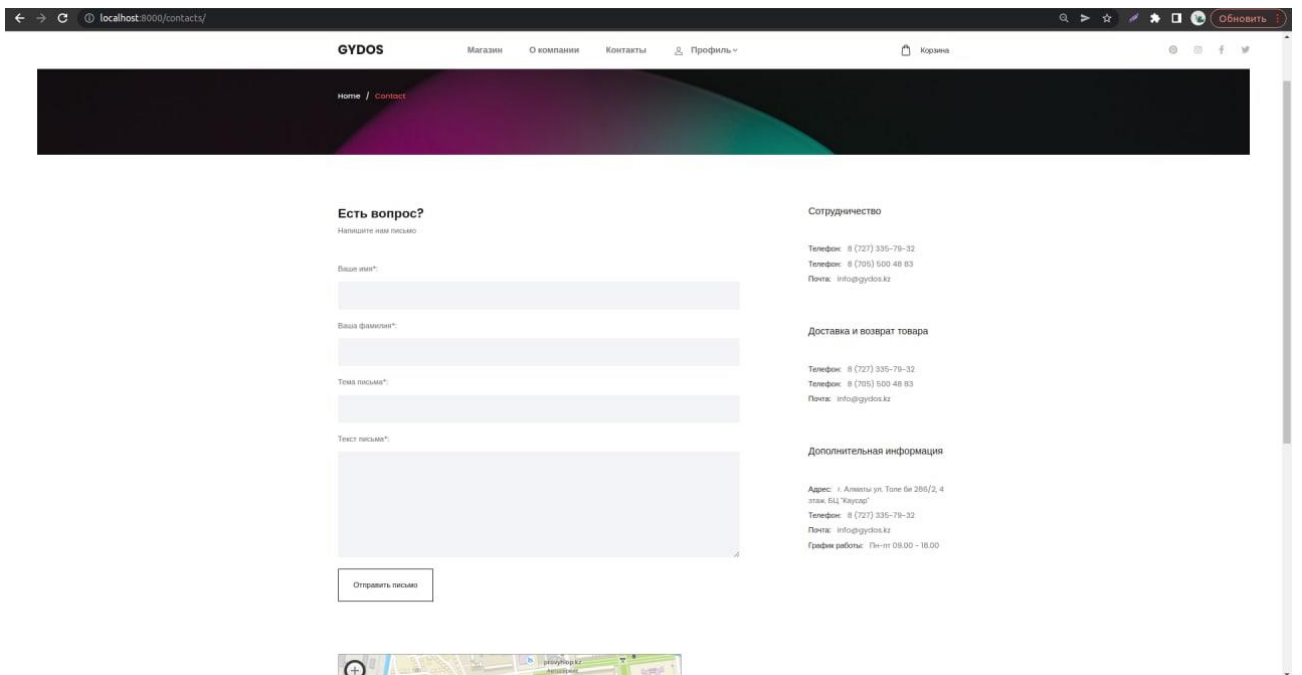


Рисунок-3.5 – Страница с контактными данными

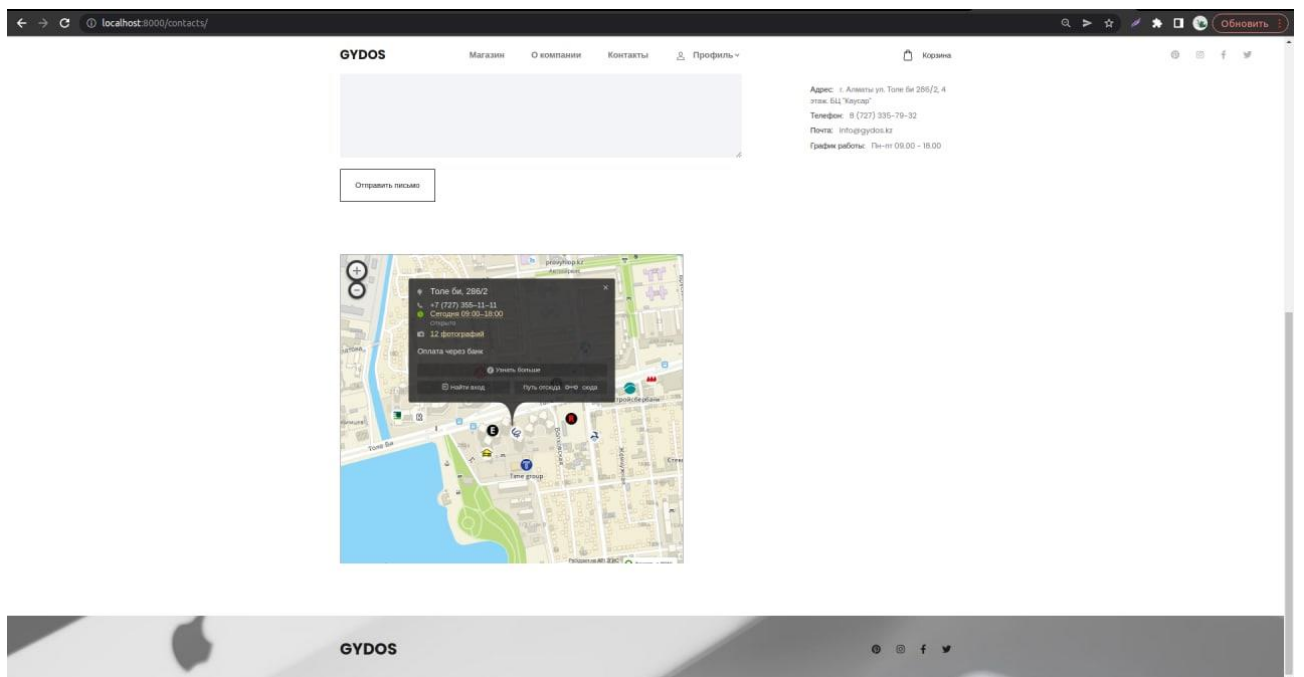


Рисунок-3.6 – Страница с навигационной картой

Далее, на рисунке ниже представлена форма входа в аккаунт. Вход и регистрация могут быть осуществимы пользователем двумя методами:

- с помощью почты и пароля, с подтверждением почты;
- с помощью аккаунта Google.

Есть возможность восстановления пароля и переадресации на регистрацию, в случае если пользователь еще не авторизован.

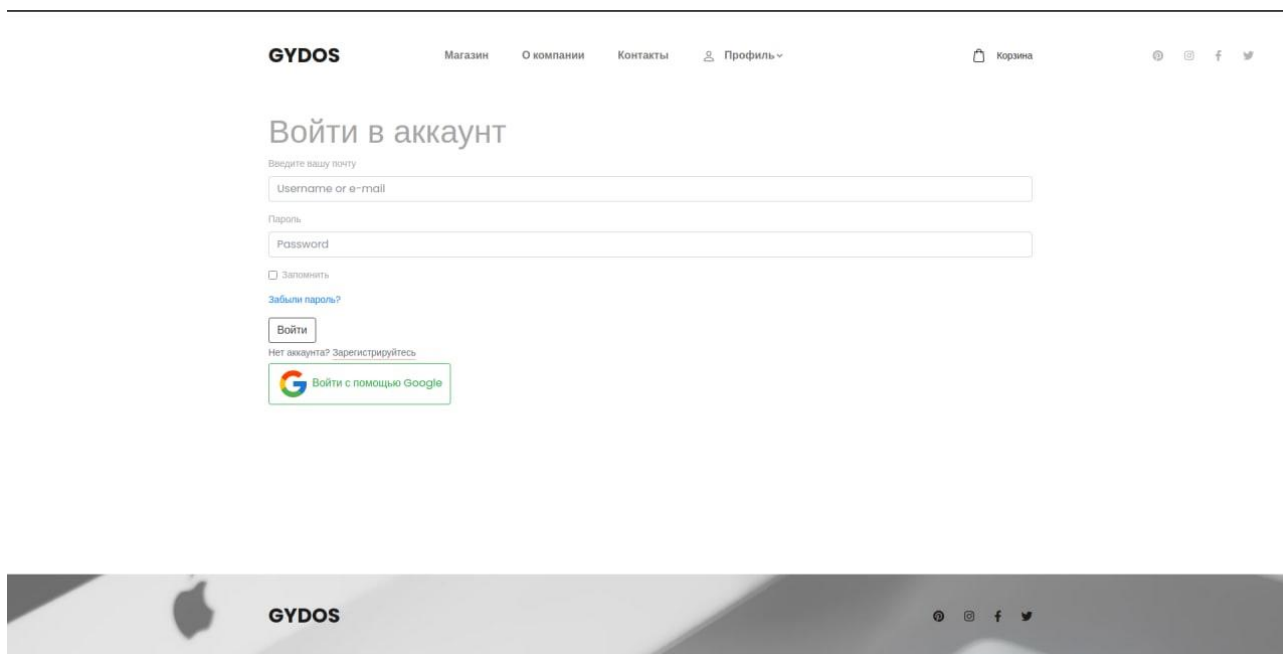


Рисунок-3.7 – Страница с формой входа в аккаунт

При выборе одной из категорий товаров, отображаются подкатегории, входящие в данную категорию (см. Рисунок-3.2.8).

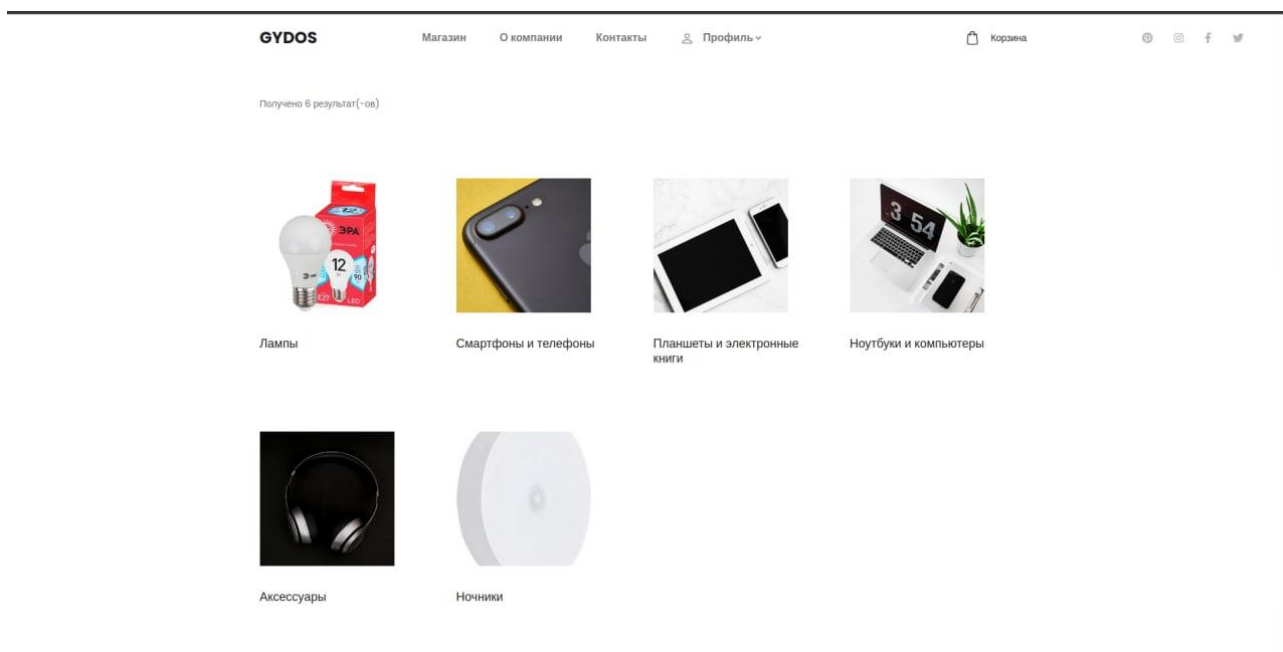


Рисунок-3.8 – Товары из категории «электротовары»

Страница с товарами из выбранной подкатегории выглядит следующим образом (см. Рисунок-3.9):

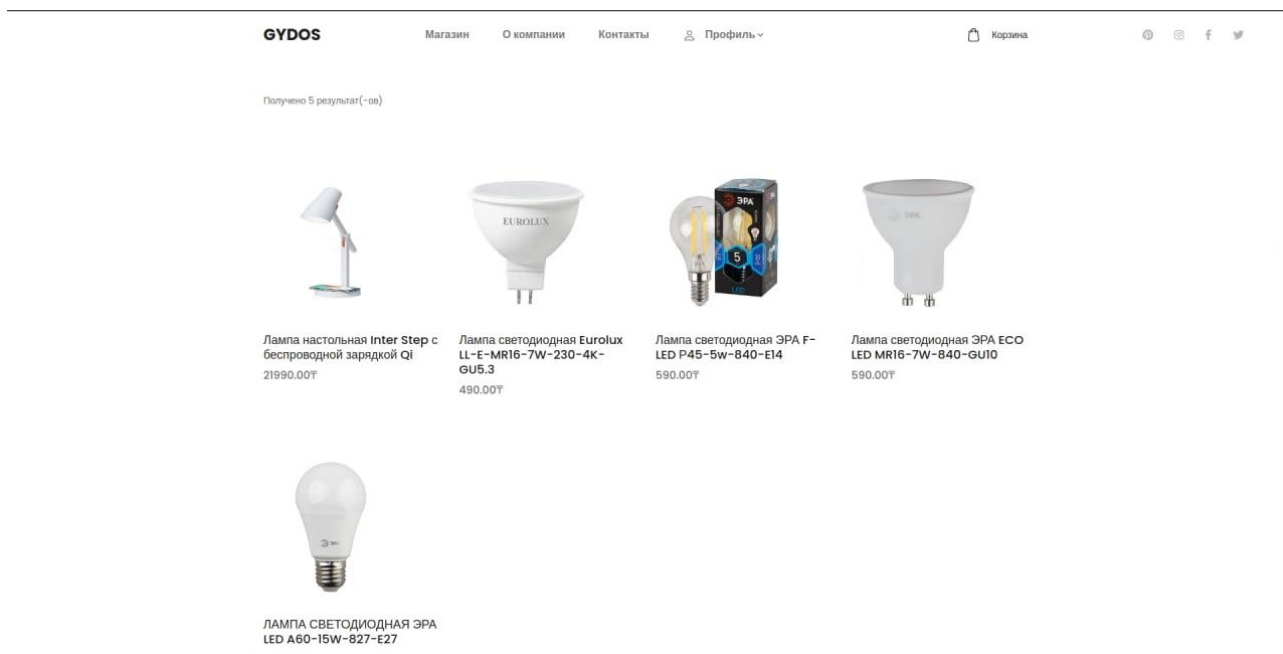


Рисунок-3.9 – Товары из категории «лампы»

Перейдя на страницу детального просмотра товара, пользователь может ознакомиться с подробной информацией и добавить товар в корзину (см. Рисунок-3.10).

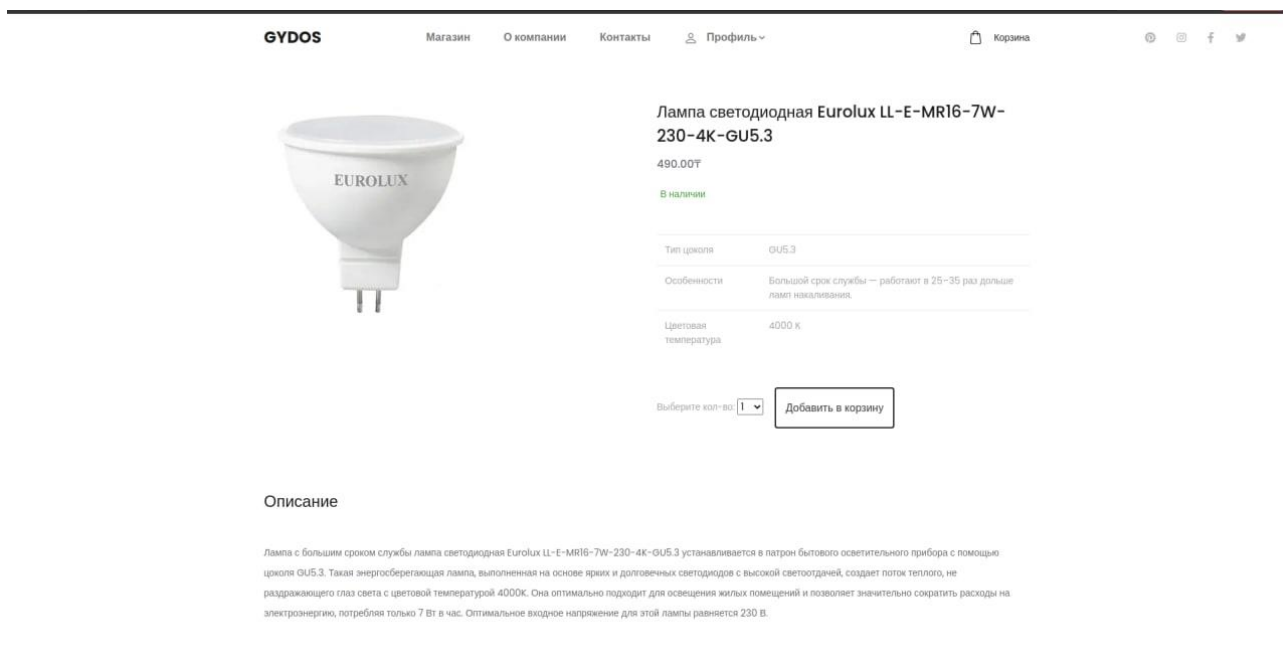


Рисунок-3.10 – Страница детального просмотра товара

На этой же странице покупателю предоставляются рекомендации к выбору (см. Рисунок-3.11).

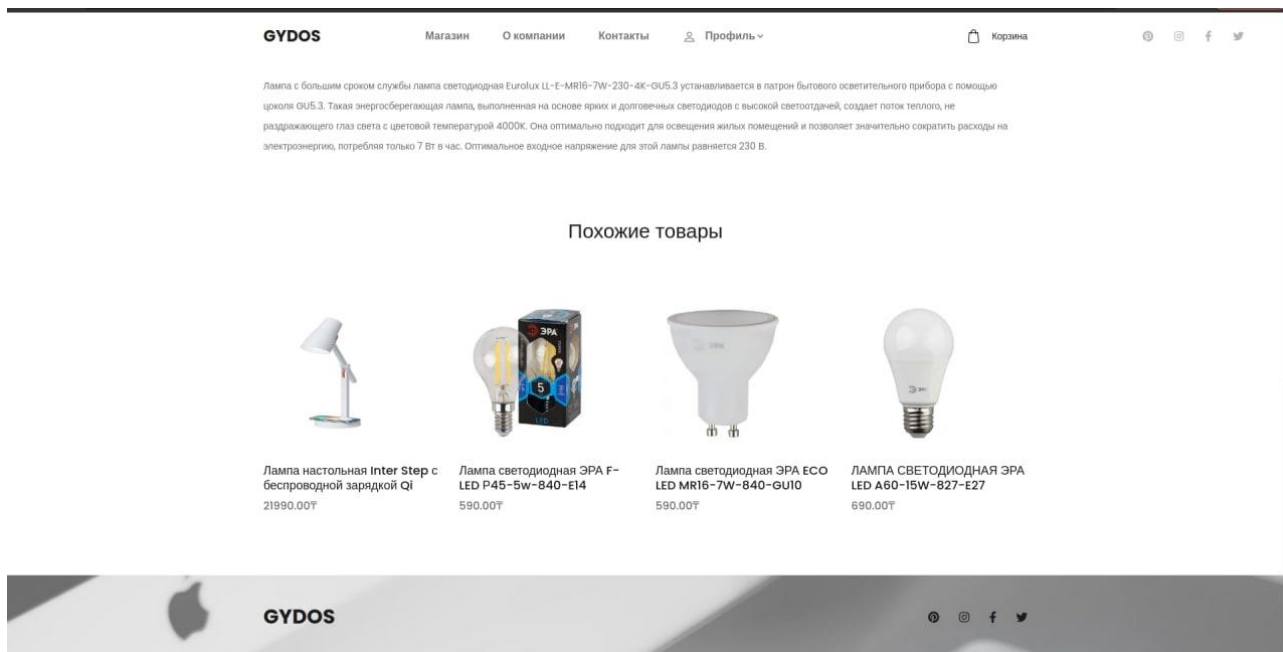


Рисунок-3.11 – Рекомендации к выбору для покупателя

В правом верхнем углу реализована страница просмотра корзины с возможностью удаления товара из корзины, возврата обратно в магазин в случае незавершенных покупок и перехода к формированию заказа (см. Рисунок-3.12).

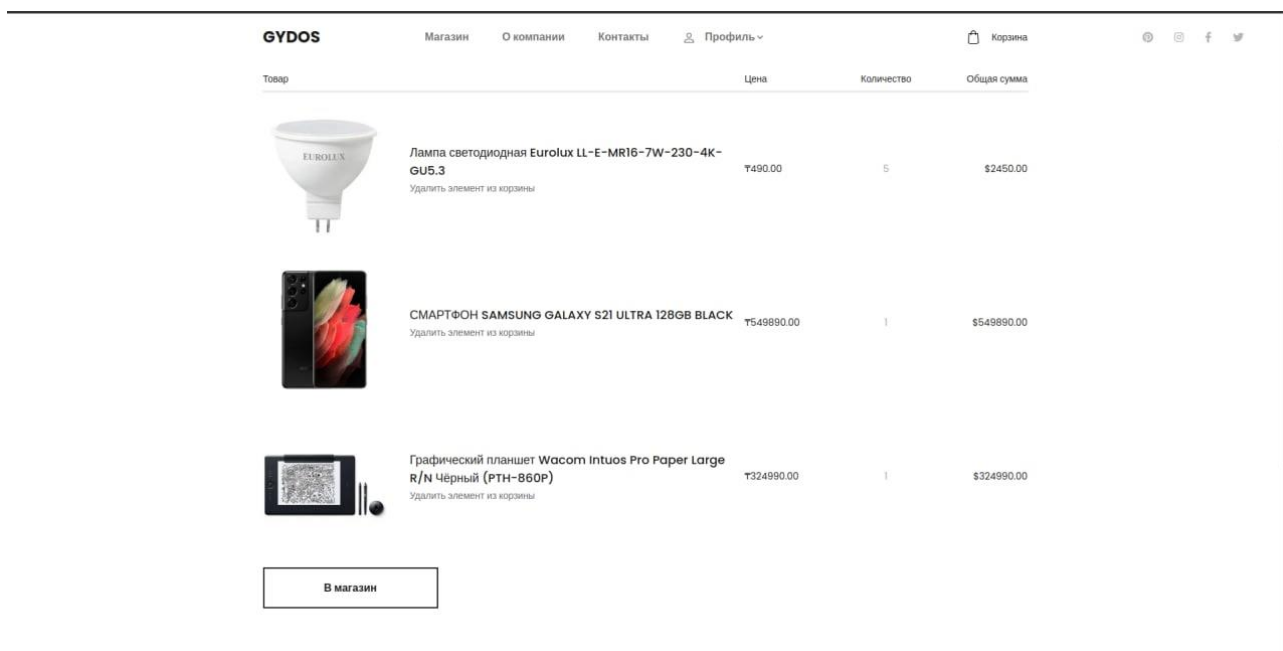


Рисунок-3.12 – Корзина с товарами

Ниже на рисунке показана страница оформления заказа (см. Рисунок-3.13).

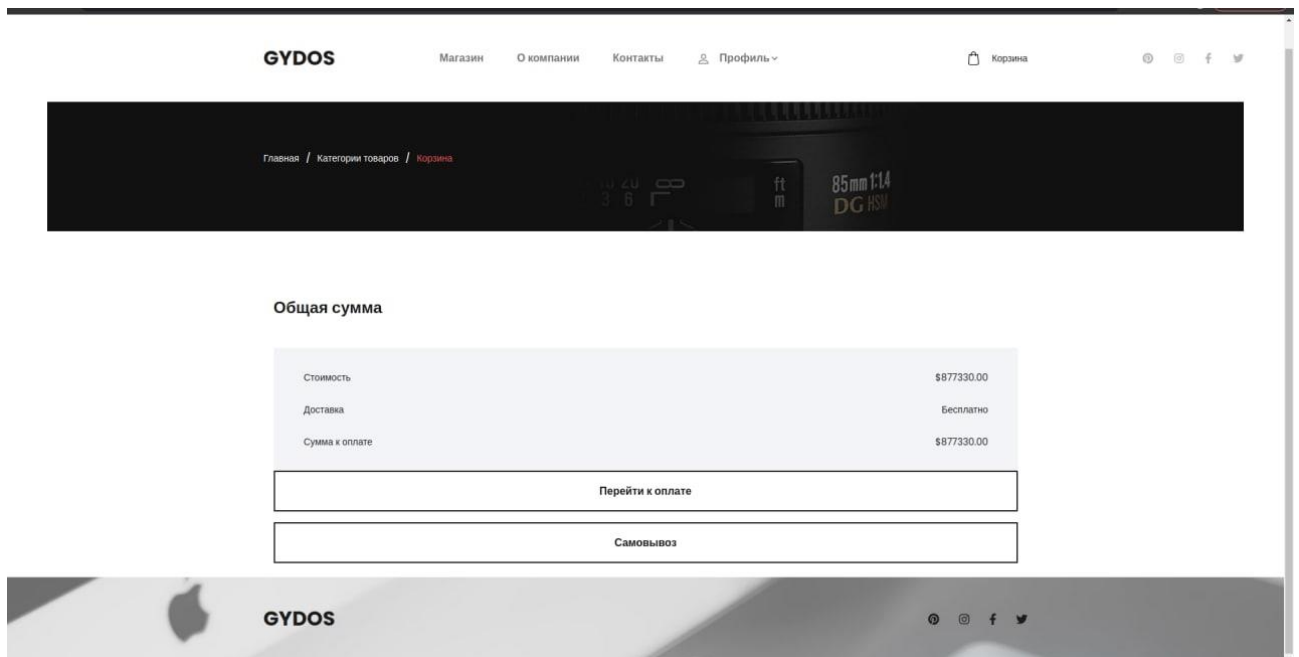


Рисунок-3.13 – Страница формирования заказа

Для продавцов есть возможность загрузки информации о всех товарах в файл.

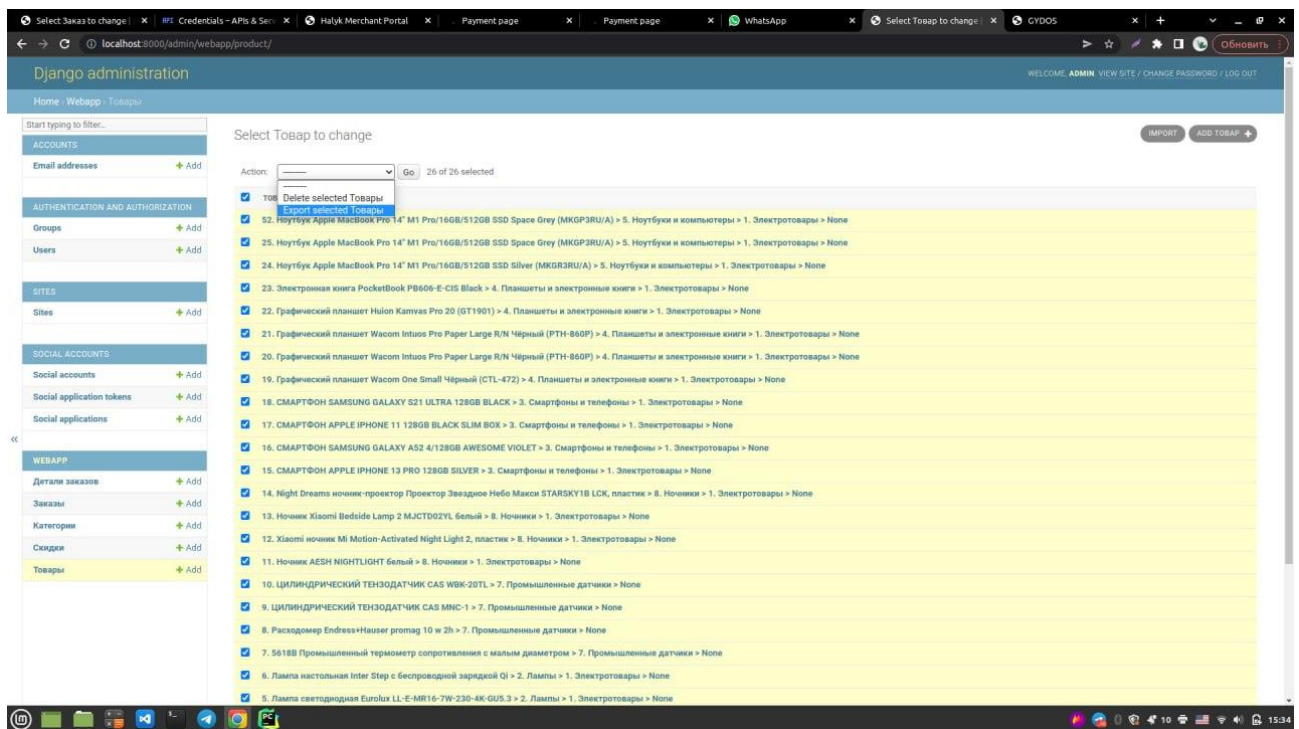


Рисунок-3.14 – Экспорт данных товаров в файл

Product-2022-05-18 (1).xlsx - LibreOffice Calc

Файл Правка Вид Вставка Формат Стили Лист Данные Сервис Окно Справка

Calibri 11 Ж К Ч А

НЗЗ fx Σ =

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Категорid	title	code	image	descript	price	available	created	updated	attribute	in_stock					
Ноутбуки	52 Ноутбук Apple MacBook Pro 14" M1	products	т	222000	1	2022-05	2022-05	{\u0414	0						
Ноутбуки	25 Ноутбук Apple MacBook Pro 14" M1	products	т	1115990	0	2022-05	2022-05	{\u0414	7000						
Ноутбуки	24 Ноутбук Apple MacBook Pro 14" M1	products	т	1180990	0	2022-05	2022-05	{\u0414	7777000						
Планшет	23 Электронная книга PocketBook PB	products	й	44990	0	2022-05	2022-05	{\u0414	7770						
Планшет	22 Графический планшет Huion Kamu	products	Если ваш	319990	1	2022-05	2022-05	{\u0414	8000						
Планшет	21 Графический планшет Wacom Intu	products	ер.	324990	0	2022-05	2022-05	{\u0414	7000						
Планшет	20 Графический планшет Wacom Intu	products	ер.	324990	0	2022-05	2022-05	{\u0414	7000						
Планшет	19 Графический планшет Wacom One	products	WacomSp	44990	1	2022-05	2022-05	{\u0414	8888						
Смартфон	18 СМАРТФОН SAMSUNG GALAXY S21	products	Датчик о	549890	1	2022-05	2022-05	{\u0414	8000						
Смартфон	17 СМАРТФОН APPLE IPHONE 11 128GB	products	Двойная	329990	0	2022-05	2022-05	{\u0414	700						
Смартфон	16 СМАРТФОН SAMSUNG GALAXY A52	products	Датчик о	179890	0	2022-05	2022-05	{\u0414	8099						
Смартфон	15 СМАРТФОН APPLE IPHONE 13 PRO 1	products	Бесконта	634990	0	2022-05	2022-05	{\u0414	8000						
Ночники	14 Night Dreams ночник-проектор Pri	products	колонка,	13938	0	2022-05	2022-05	{\u0414	7777						
Ночники	13 Ночник Xiaomi Bedside Lamp 2 MJ	products	Уникальн	18600	0	2022-05	2022-05	{\u0414	7000						
Ночники	12 Xiaomi ночник Mj Motion-Activated	products	я	5400	0	2022-05	2022-05	{\u0414	800						
Ночники	11 Ночник AESH NIGHTLIGHT Белый	products	безопасн	2100	0	2022-05	2022-05	{\u0414	500						
Промышл	10 ЦИЛИНДРИЧЕСКИЙ ТЕНЗОДАТЧИК	products	Самосто	194042	0	2022-05	2022-05	{\u0414	10000						
Промышл	9 ЦИЛИНДРИЧЕСКИЙ ТЕНЗОДАТЧИК	products	Стальные	127445	0	2022-05	2022-05	{\u0414	700						
Промышл	8 Расходомер Endress+Hauser proma	products	нием.	2000000	0	2022-05	2022-05	{\u0414	500						
Промышл	7 5618В Промышленный термометр	products	1560.	10000	0	2022-05	2022-05	{\u0414	1000						
Лампы	6 Лампа настольная Inter Step с бес	products	Images/pe	21990	0	2022-05	2022-05	{\u0414	800						
Лампы	5 Лампа светодиодная Eurolux LL-E	products	я 230 В.	490	0	2022-05	2022-05	{\u0414	100						
Лампы	4 Лампа светодиодная ЭРА F-LED P4	products	я	590	0	2022-05	2022-05	{\u0414	800						
Лампы	3 Лампа светодиодн	0	products	Современ	590	0	2022-05	2022-05	{\u0414	100					
Электрон	2 МОДЕМ TP-LINK TD	63347		16000	1	2022-03	2022-05	{\u0414	3						
Лампы	1 ЛАМПА СВЕТОДИО	5425	products	Бытовая	690	1	2022-03	2022-05	{\u0414	1					

Рисунок-3.15 – Файл, загруженный на устройство после экспорта

Также, для продавцов есть возможность загрузки информации о товарах в базу данных с помощью импорта из файла.

Django administration

Home Webapp Товары Import

Start typing to filter...

ACCOUNTS

Email addresses + Add

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

SITES

Sites + Add

SOCIAL ACCOUNTS

Social accounts + Add

Social application tokens + Add

Social applications + Add

WEBAPP

Детали заказов + Add

Заказы + Add

Категории + Add

Сюжеты + Add

Товары + Add

Import

This importer will import the following fields: Категория, id, title, code, image, description, price, available, created_at, updated_at, attributes, in_stock

File to import: Product2...18 (2).xlsx

Format:

Рисунок-3.16 – Выбор расширения файла импорта и загрузка файла

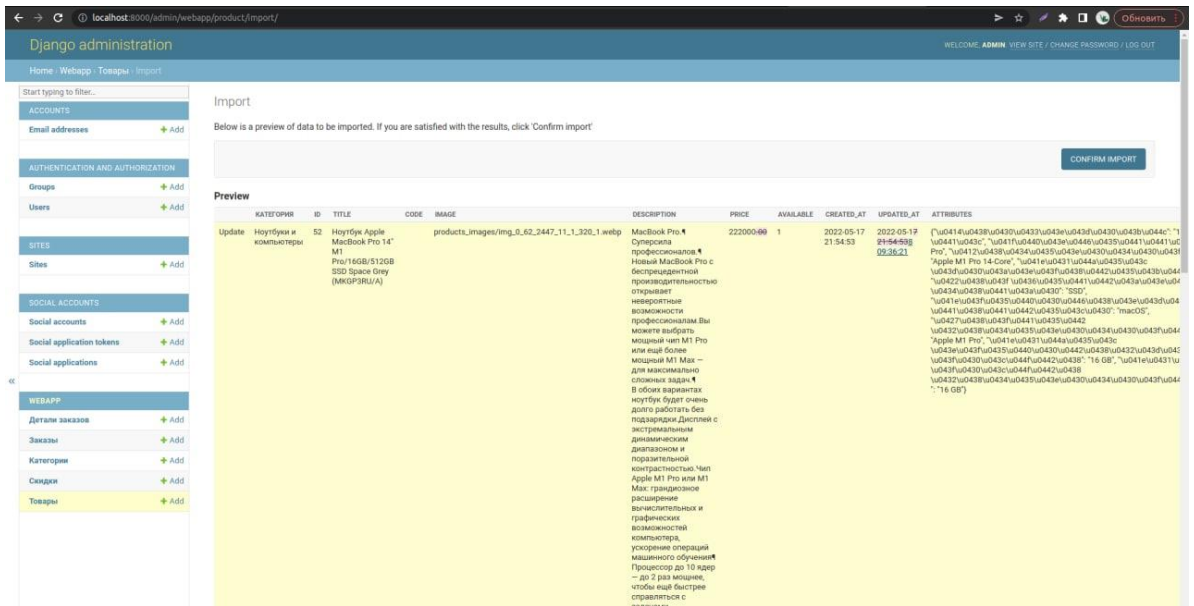


Рисунок-3.17 – Предварительный просмотр записи о товаре в базе данных

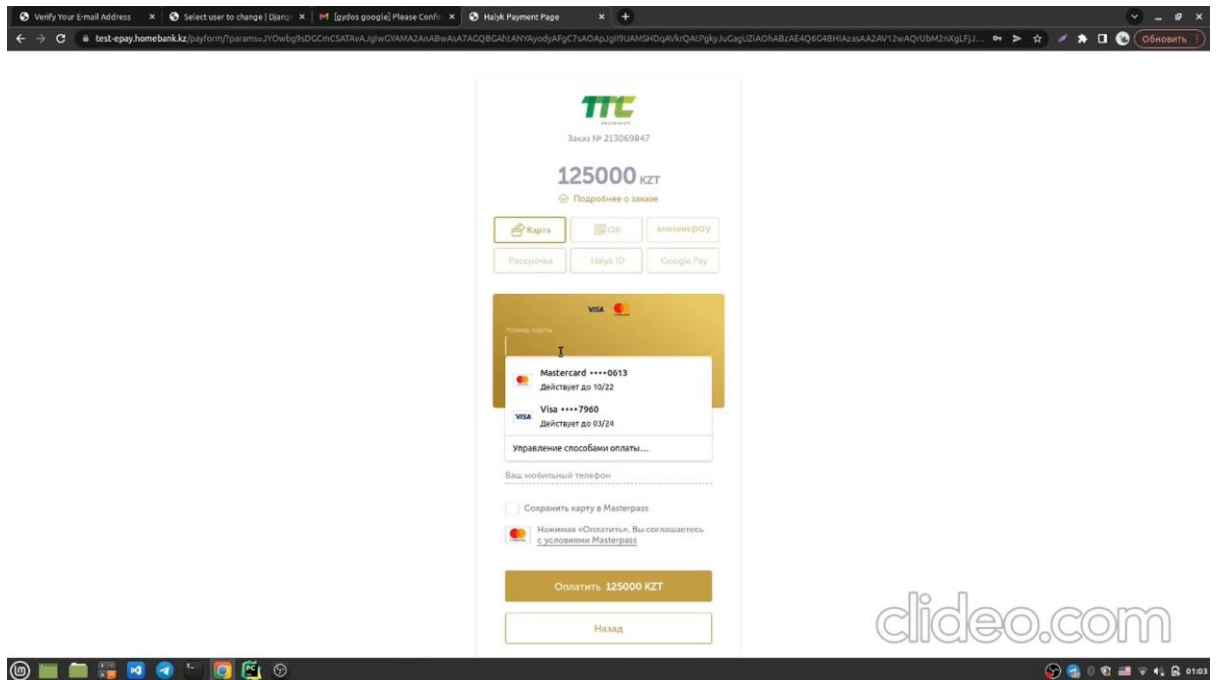
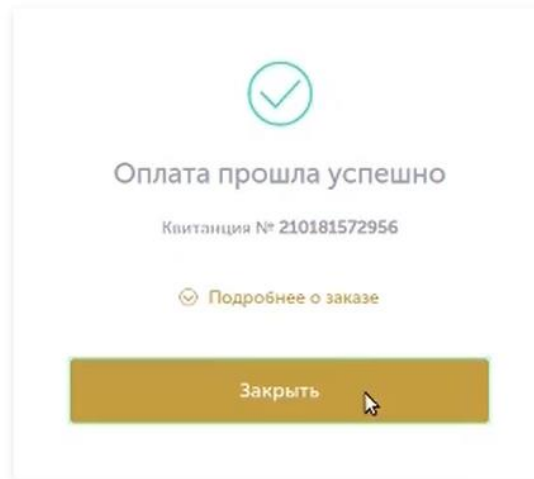


Рисунок-3.18 – После добавления товара в корзину и нажатия на «Оплатить» происходит перенаправление на страницу платежной системы



Информация о Visa 3-D Secure и
Mastercard SecureCode владельцам
платежных карт: **Halyk Bank**

Рисунок-3.19 – Уведомление о статусе платежа

После оплаты, на почту пользователя присылается чек с информацией о платеже.

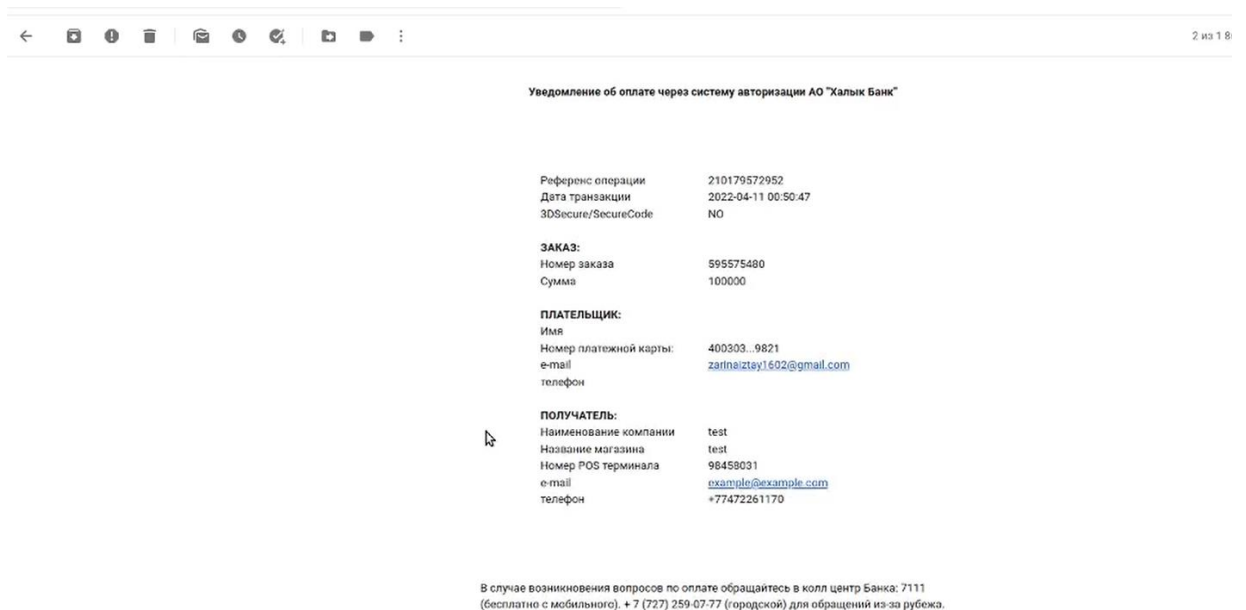


Рисунок-3.20 – Чек об оплате, присланный на почту пользователя

ЗАКЛЮЧЕНИЕ

Результатом данной дипломной работы является автоматизированная информационная система онлайн-магазина по запросу компании Gydos. Программный продукт может использоваться не только для принятия и обработки заявок от покупателей, но и как система учета товаров для бизнеса.

В процессе проектирования и реализации были неоднократные взаимодействия с заказчиком в лице представителя от компании.

В процессе исследования предметной области был проведён анализ сравнительных характеристик информационных систем, примененной технологии для разработки. Данное приложение ориентировано на общую аудиторию и является свободным и распространяемым ресурсом. Основными требованиями по созданию приложения были функциональные возможности, такие как:

- просмотр каталога товаров;
- добавление товаров с уникальными полями в базу;
- добавление товаров в корзину;
- оплата стоимости заказа с помощью платежной системы;
- получение чека и уведомления на почту.

В процессе разработки приложение обрело собственный стиль, дополнилось функционалом, которые запросы были пересмотрены. Во время разработки были открыты новые возможности фреймворка Django, приняты во внимание многие современные методологии и концепции программирования.

Произошло знакомство с реляционной базой данных PostgreSQL, которая значительно улучшила качество разработки и скорость работы самого приложения.

Проект был создан с помощью фреймворка Django на языке Python и базы данных PostgreSQL, с интеграцией платежной системы от Halyk Bank. Все вышеперечисленные цели и требования были достигнуты.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- 1 Документация Django // Электронная версия на сайте <https://docs.djangoproject.com/en/4.0/>
- 2 Запрос данных PostgreSQL // Электронная версия на сайте <https://www.8host.com/blog/zapros-dannyx-postgresql-pri-pomoshhi-komandy-select/>
- 3 Model field reference // Электронная версия на сайте <https://docs.djangoproject.com/en/4.0/ref/models/fields/#jsonfield>
- 4 Пробуем JSON в Django и PostgreSQL (и сравниваем с MongoDB) // Электронная версия на сайте <http://www.lexev.org/2015/trying-json-combo-django-and-postgresql/>
- 5 Antonio Melé Django 3 By Example. – Birmingham: Packt Publishing Ltd., 2020. – 569 с.
- 6 Логирование в Django // Электронная версия на сайте <https://djbook.ru/rel1.4/topics/logging.html>
- 7 Django Allauth Учебник Социальные логины // Электронная версия на сайте <https://pythobyte.com/django-allauth-tutorial-social-logins-bqj8gk7cd-1a73d811/>
- 8 The future of ecommerce: how ecommerce will change in 2021 and beyond // Электронная версия на сайте <https://www.oberlo.com/blog/future-of-ecommerce>
- 9 The Future of eCommerce: 8 Trends to Watch Out for in 2021 // Электронная версия на сайте <https://blog.shift4shop.com/future-of-ecommerce-2021>
- 10 Сайт на конструкторе: плюсы, минусы и когда стоит их использовать // Электронная версия на сайте <https://olprime.ru/blog/konstruktor-saytov-vse-za-i-protiv/>
- 11 Django Руководство часть 11: Разворачивание сайта на сервере // Электронная версия на сайте <https://developer.mozilla.org/ru/docs/Learn/Server-side/Django/Deployment>
- 12 Heroku // Электронная версия на сайте <https://blog.skillfactory.ru/glossary/heroku/>
- 13 JWT Аутентификация в Django // Электронная версия на сайте <https://code.tutsplus.com/ru/tutorials/how-to-authenticate-with-jwt-in-django--cms-30460>
- 14 E-commerce worldwide // Электронная версия на сайте <https://www.statista.com/topics/871/online-shopping/>
- 15 Top 10 Online Best Shopping Sites in World 2022 [Updated] // Электронная версия на сайте <https://www.bloggersideas.com/top-10-online-shopping-sites-in-world-best-shopping-sites-in-world/>
- 16 8 Types of eCommerce Websites & Business Models // Электронная версия на сайте <https://www.hostgator.com/blog/ecommerce-website-types-business-models/>

- 17 When to Use Django? Comparison with other Development Stacks // Электронная версия на сайте <https://www.geeksforgeeks.org/when-to-use-django-comparison-with-other-development-stacks/>
- 18 Top 10 backend frameworks – Which is the best? // Электронная версия на сайте <https://blog.back4app.com/backend-frameworks/>
- 19 Django Developers Survey 2021 // Электронная версия на сайте <https://lp.jetbrains.com/django-developer-survey-2021-486/>
- 20 Форсье Джефф, Биссекс Пол Django. Разработка веб-приложений на Python. – М.: Символ-Плюс, 2021. – 150 с.
- 21 Дронов В.А. Django 2.1. Практика создания веб-сайтов на Python. - СПб.: ВHV, 2019. – 672 с.
- 22 Адриан Головатый, Джейкоб Каплан-Мосс Django. - М.: Символ-Плюс, 2019, - 558 с.
- 23 Прохоренок Н.А., Дронов В.А. Python 3. Самое необходимое. - СПб.: ВHV, 2019. – 608 с.
- 24 Кириченко А.В. Динамические сайты на HTML, CSS, JavaScript и Bootstrap. Практика, практика и только практика. - СПб.: Наука и техника, 2021. – 272 с.

Приложение А

Техническое задание

А.1.5 Техническое задание на разработку интернет-магазина для компании Gydos

Настоящее техническое задание распространяется на разработку интернет-магазина для компании Gydos. Предполагается, что этот продукт будут использовать для электронной коммерции и онлайн продаж товаров компании, а также для маркетинговых целей компании. Клиенты компании смогут заказывать товары, сотрудники отслеживать поступающие заказы, экспортировать и импортировать данные из базы данных в удобном им формате.

А.1.5.1 Основание для разработки

Система разрабатывается в качестве дипломного проекта.

А.1.5.2 Назначение

Программное обеспечение, разработанное в виде веб-приложения предназначено для совершения покупок в интернет-магазине компании Gydos, а также оптимизации и автоматизации работы сотрудников компании.

А.1.5.3 Требования к функциональным характеристикам

Информационная система интернет-магазина предоставляет следующий функционал:

- регистрация и авторизация пользователей. Регистрация производится в двух вариантах: вход с помощью Google и регистрация с почтой в качестве идентификатора, паролем и верификацией аккаунта через почту. Войти можно будет так же с помощью Google, либо введя почту в качестве логина и пароль;
- возможность восстановить пароль для авторизованных пользователей;
- просмотр главной страницы с товарами, новостями от компании;
- просмотр страницы с маркетинговой информацией о компании;

Продолжение приложения А

- страница с контактными данными, адресом компании с возможностью оставить отзыв/письмо в форме;
- просмотр страницы товара;
- добавление товара в корзину;
- удаление товара(-ов) из корзины;
- возможность оплатить корзину;
- рассылка уведомления о новом заказе сотруднику;
- генерация QR кода для клиента, отсканировав который, сотрудник моментально получит информацию о клиенте, его заказах и т.д.;
- возможность отмены заказа и возврата денег клиенту.

А.1.5.4 Требования к надежности

Предусмотреть проверку вводимой информации. Предусмотреть некорректные действия со стороны пользователя при работе с системой. Обеспечить целостность хранимой информации.

Приложение Б

Логика и текст ПО интернет-магазина Gydos views.py

```
from django.contrib.auth import login
from django.contrib.auth.models import User
from django.shortcuts import redirect
from django.urls import reverse
from django.views.generic import CreateView
from .form import MyUserCreationForm

#Регистрация пользователя
class RegisterView(CreateView):
    model = User
    template_name = 'user_create.html'
    form_class = MyUserCreationForm

    def form_valid(self, form):
        user = form.save()
        login(self.request, user)
        return redirect(self.get_success_url())

    def get_success_url(self):
        next_url = self.request.GET.get('next')
        if not next_url:
            next_url = self.request.POST.get('next')
        if not next_url:
            next_url = reverse('webapp:index')
        return next_url

import uuid
from django.shortcuts import render, redirect, get_object_or_404
from django.views.decorators.csrf import csrf_exempt
from django.views.decorators.http import require_POST
from webapp.models import Product
from .cart import Cart
from .forms import CartAddProductForm
import logging
import json
```

Продолжение приложения Б

#добавление в корзину

```
@require_POST
def cart_add(request, product_id):
    cart = Cart(request)
    product = get_object_or_404(Product, id=product_id)
    form = CartAddProductForm(request.POST)
    if form.is_valid():
        cd = form.cleaned_data
        cart.add(product=product,
                quantity=cd['quantity'],
                update_quantity=cd['update'])
    return redirect('cart:cart_detail')
```

#удаление товара из корзины

```
def product_remove(request, product_id):
    cart = Cart(request)
    product = get_object_or_404(Product, id=product_id)
    cart.remove(product)
    return redirect('cart:cart_detail')
```

#очистка корзины

```
def cart_clear(request):
    cart = Cart(request)
    cart.clear()
    return redirect('cart:cart_detail')
```

#посмотреть товары в корзине

```
def cart_detail(request):
    cart = Cart(request)
    for item in cart:
        item['update_quantity_form'] =
            CartAddProductForm(initial={'quantity': item['quantity'],
                                        'update': True})
    code = str(uuid.uuid4()).int & (1 << 64) - 1)[-9:]
    return render(request, 'cart.html', {'cart': cart, 'code': code})
```

```
logger = logging.getLogger('main')
```

Продолжение приложения Б

```
#представление, принимающее сообщение от платежной системы
@csrf_exempt
def postlink_catch_view(request):
    data = json.loads(request.body.decode())
    logger.info(data)

    # if data['code'] == 'ok':
    #     return redirect('orders:order_create', data=data)

from django.shortcuts import render, get_object_or_404
from django.views import View
from django.views.generic import TemplateView, ListView
from django.shortcuts import render
from webapp.models import Category, Product
from cart.forms import CartAddProductForm
from webapp.form import PostMessageForm

class IndexView(ListView): #для первоначального отображения всех
категорий
    template_name = 'index.html'
    context_object_name = 'categories'

    def get_queryset(self):
        return Category.objects.filter(parent_category=None).order_by('-
created_at')

# Есть категории, а есть надкатегории.
# надкатегория > категории > страница товара
# категории > страница товара

class ParentCategoriesView(ListView): #для первоначального
отображения всех категорий
    template_name = 'categories.html'
    context_object_name = 'categories'

    def get_queryset(self):
        return Category.objects.filter(parent_category=None).order_by('-
created_at')
```

Продолжение приложения Б

```
class CategoryView(View): #при нажатии на одну из категорий
    def get(self, request, *args, **kwargs):
        category = get_object_or_404(Category, pk=kwargs.get('pk'))

        if category.pk in Category.objects.values_list('parent_category', flat=True):
            context = {
                'categories': Category.objects.filter(parent_category=category.pk)
            }
            return render(request, 'categories.html', context)
        else:
            context = {
                'products': Product.objects.filter(category=category.pk)
            }
            return render(request, 'products.html', context)

class ProductView(View):
    def get(self, request, *args, **kwargs):
        product = get_object_or_404(Product, pk=kwargs.get('pk'))
        cart_product_form = CartAddProductForm()
        return render(request, 'product.html', context={'product': product,
            'cart_product_form': cart_product_form})

class ContactView(View):
    def get(self, request, *args, **kwargs):
        form = PostMessageForm()
        return render(request, 'contact.html', {'form': form})

    def post(self, request, *args, **kwargs):
        form = PostMessageForm(request.POST)
        if form.is_valid():
            print(form.cleaned_data)
            return render(request, 'contact.html', {'form': form})

class AboutUsView(TemplateView):
    template_name = "about.html"
```

Продолжение приложения Б

Список необходимых настроек ПО

```
import os
from pathlib import Path
import mimetypes

mimetypes.add_type("image/svg+xml", ".svg", True)
mimetypes.add_type("image/svg+xml", ".svgz", True)

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

CSRF_TRUSTED_ORIGINS = ['http://e638-95-56-25-214.ngrok.io', 'http://localhost:8000']

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-z#&rb1yys1($w_7fti%k-69f517d0$97s730vb$s^iko1q%xld'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = ['e638-95-56-25-214.ngrok.io', 'localhost']

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'django.contrib.sites',
    'allauth',
    'allauth.account',
```

Продолжение приложения Б

```
'allauth.socialaccount',

'allauth.socialaccount.providers.google',

'webapp',
'accounts',
'cart'
# 'django.contrib.postgres'
]

MIDDLEWARE = [
'django.middleware.security.SecurityMiddleware',
'django.contrib.sessions.middleware.SessionMiddleware',
'django.middleware.common.CommonMiddleware',
'django.middleware.csrf.CsrfViewMiddleware',
'django.contrib.auth.middleware.AuthenticationMiddleware',
'django.contrib.messages.middleware.MessageMiddleware',
'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'gydos.urls'

TEMPLATES = [
{
    'BACKEND': 'django.template.backends.django.DjangoTemplates',
    'DIRS': [os.path.join(BASE_DIR, 'templates'), os.path.join(BASE_DIR,
'webapp', 'templates')],
    'APP_DIRS': True,
    'OPTIONS': {
        'context_processors': [
            'django.template.context_processors.debug',
            'django.template.context_processors.request',
            'django.contrib.auth.context_processors.auth',
            'django.contrib.messages.context_processors.messages',
        ],
    },
},
]

WSGI_APPLICATION = 'gydos.wsgi.application'
```

Продолжение приложения Б

```
# Database
# https://docs.djangoproject.com/en/4.0/ref/settings/#databases

DATABASES = {

    'default': {
        'ENGINE': 'django.db.backends.postgresql_psycopg2',
        'NAME': 'gydos',
        'USER': 'gydos_user',
        'PASSWORD': 'gydos@123',
        'HOST': 'localhost',
        'PORT': '5432'
    }
}

AUTHENTICATION_BACKENDS = [
    # Needed to login by username in Django admin, regardless of `allauth`
    'django.contrib.auth.backends.ModelBackend',

    # `allauth` specific authentication methods, such as login by e-mail
    'allauth.account.auth_backends.AuthenticationBackend',
]

# Password validation
# https://docs.djangoproject.com/en/4.0/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME':
```


Продолжение приложения Б

```
'django.contrib.auth.password_validation.NumericPasswordValidator',  
    },  
]
```

```
# Internationalization  
# https://docs.djangoproject.com/en/4.0/topics/i18n/
```

```
LANGUAGE_CODE = 'en-us'
```

```
TIME_ZONE = 'UTC'
```

```
USE_I18N = True
```

```
USE_TZ = True
```

```
# Static files (CSS, JavaScript, Images)  
# https://docs.djangoproject.com/en/4.0/howto/static-files/
```

```
STATIC_URL = 'static/'
```

```
SETTINGS_EXPORT = [  
    'STATIC_URL',  
]
```

```
SITE_ID = 1
```

```
SESSION_ENGINE = 'cart.session_backend'
```

```
# Default primary key field type  
# https://docs.djangoproject.com/en/4.0/ref/settings/#default-auto-field
```

```
DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'
```

```
MEDIA_ROOT = os.path.join(BASE_DIR, 'uploads')  
MEDIA_URL = '/uploads/'
```

```
LOGIN_REDIRECT_URL = '/'  
LOGOUT_REDIRECT_URL = 'webapp:index'  
ACCOUNT_AUTHENTICATION_METHOD = 'username_email'
```

Продолжение приложения Б

```
ACCOUNT_EMAIL_REQUIRED = True
ACCOUNT_EMAIL_VERIFICATION = 'mandatory'
EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
# LOGOUT_REDIRECT_URL = 'accounts:login'
ACCOUNT_USERNAME_REQUIRED = False
```

```
EMAIL_HOST = 'smtp.gmail.com'
EMAIL_USE_TLS = True
EMAIL_PORT = 587
```

```
EMAIL_HOST_USER = 'gydoscompany@gmail.com'
EMAIL_HOST_PASSWORD = 'EFS%FYWj'
DEFAULT_FROM_EMAIL = EMAIL_HOST_USER
```

```
SOCIALACCOUNT_ADAPTER =
'webapp.adapter.MySocialAccountAdapter'
```

```
# Provider specific settings
SOCIALACCOUNT_PROVIDERS = {
    'google': {
        'SCOPE': [
            'profile',
            'email',
        ],
        'AUTH_PARAMS': {
            'access_type': 'online',
        }
    }
}
```

```
if not DEBUG:
    STATIC_ROOT = 'static'
else:
    STATICFILES_DIRS = [
        os.path.join(BASE_DIR, 'webapp/static'),
        os.path.join(BASE_DIR, 'cart/static'),
    ]
```

```
CART_SESSION_ID = 'cart'
```

```
LOGGING = {
    'version': 1,
```

Продолжение приложения Б

```
'disable_existing_loggers': False,

'formatters': {
    "main_format": {
        "format": "{asctime} - {levelname} - {module} - {filename} -
{message}",
        "style": "{",
    },
},

'handlers': {
    'console': {

        'class': 'logging.StreamHandler',
        'formatter': 'main_format',
    },
    'file': {
        "class": "logging.FileHandler",
        "formatter": "main_format",
        'filename': 'information.log'
    },
},

'loggers': {
    'main': {
        'handlers': ['console', 'file'],
        'level': 'DEBUG',
        'propagate': True,
    },
},
}
```

Продолжение приложения Б

Список необходимых расширений

asgiref==3.5.0
backports.zoneinfo==0.2.1
certifi==2021.10.8
cffi==1.15.0
charset-normalizer==2.0.12
cryptography==36.0.2
defusedxml==0.7.1
Django==4.0.2
django-allauth==0.50.0
django-autoslug==1.9.8
idna==3.3
oauthlib==3.2.0
Pillow==9.0.1
pkg_resources==0.0.0
psycpg2-binary==2.9.3
pycparser==2.21
PyJWT==2.3.0
python3-openid==3.2.0
requests==2.27.1
requests-oauthlib==1.3.1
sqlparse==0.4.2
urllib3==1.26.9

Продолжение приложения Б

Необходимые модели базы данных

```
from django.db import models
from autoslug import AutoSlugField
from django.contrib.auth.models import User

class Category(models.Model):
    title = models.CharField(max_length=200, null=False, blank=False,
        verbose_name='Название категории')
    parent_category = models.ForeignKey('self',
        on_delete=models.CASCADE, null=True, blank=True,
        verbose_name='Родительская категория')
    created_at = models.DateTimeField(auto_now_add=True,
        verbose_name='Время создания')
    image = models.ImageField(null=True, blank=True,
        upload_to='categories_images', verbose_name='Фотография категории')

    def __str__(self):
        return "{}. {} > {}".format(self.pk, self.title, self.parent_category)

    class Meta:
        verbose_name = 'Категория'
        verbose_name_plural = 'Категории'

class Product(models.Model):
    category = models.ForeignKey('webapp.Category',
        related_name='products', on_delete=models.CASCADE,
        verbose_name='Категория товара')
    title = models.CharField(max_length=200, verbose_name='Название
        товара')
    code = models.PositiveIntegerField(unique=True, verbose_name='Код
        товара', null=True, blank=True)
    image = models.ImageField(null=True, blank=True,
        upload_to='products_images', verbose_name='Фотография товара')
    description = models.TextField(null=True, blank=True,
        help_text='Дополнительное необязательное описание товара',
        verbose_name='Текстовое описание товара')
    price = models.DecimalField(max_digits=12, decimal_places=2,
        verbose_name='Цена товара')
```

Продолжение приложения Б

```
available = models.BooleanField(verbose_name='Доступность товара')
created_at = models.DateTimeField(auto_now_add=True,
verbose_name='Время создания товара')
updated_at = models.DateTimeField(auto_now=True,
verbose_name='Время
изменения товара')
attributes = models.JSONField()
in_stock = models.PositiveIntegerField(verbose_name='Количество
товара', default=0)
# slug = AutoSlugField(populate_from='title', editable=True, blank=True)

def __str__(self):
    return "{}. {} > {}".format(self.pk, self.title, self.category)

class Meta:
    verbose_name = 'Товар'
    verbose_name_plural = 'Товары'

class Order(models.Model):
    client_id = models.ForeignKey(User, on_delete=models.CASCADE,
related_name='orders', verbose_name='id клиента')
    created_at = models.DateTimeField(auto_now_add=True,
verbose_name='Время создания заказа')
    updated_at = models.DateTimeField(auto_now=True,
verbose_name='Время изменения заказа')
    total_price = models.DecimalField(max_digits=12, decimal_places=2,
verbose_name='Сумма заказа')
    is_paid = models.BooleanField(default=False, verbose_name='Оплачен
ли заказ')
    is_received = models.BooleanField(default=False,
verbose_name='Получен ли заказ')
    description = models.TextField(null=True, blank=True,
verbose_name='Комментарий к заказу (со стороны сотрудника)')

def __str__(self):
    return "{}".format(self.pk)

class Meta:
    verbose_name = 'Заказ'
    verbose_name_plural = 'Заказы'
```

Продолжение приложения Б

```
class OrderDetails(models.Model):
    order_id = models.ForeignKey('webapp.Order', related_name='details',
on_delete=models.CASCADE,
        verbose_name='id заказа')
    product_id = models.ForeignKey('webapp.Product',
related_name='order_details', on_delete=models.CASCADE,
        verbose_name='id продукта')
    quantity = models.PositiveIntegerField(verbose_name='Количество
товара')
    discount = models.ForeignKey('webapp.Order', null=True, blank=True,
related_name='order_details', on_delete=models.SET_NULL,
        verbose_name='id скидки')

    def __str__(self):
        return "{}".format(self.pk)

    class Meta:
        verbose_name = 'Детали заказа'
        verbose_name_plural = 'Детали заказов'

class Discount(models.Model):
    percent = models.DecimalField(max_digits=5, decimal_places=2,
verbose_name='процент скидки')
    title = models.CharField(max_length=200, verbose_name='название
скидки')
    description = models.TextField(null=True, blank=True,
verbose_name='описание скидки')

    def __str__(self):
        return "{}".format(self.pk)

    class Meta:
        verbose_name = 'Скидка'
        verbose_name_plural = 'Скидки'
```

Продолжение приложения Б

Текст программы forms.py

```
from django.contrib.auth.forms import UserCreationForm

from django.contrib.auth.forms import UserCreationForm

class MyUserCreationForm(UserCreationForm):
    class Meta(UserCreationForm.Meta):
        fields = ['username', 'password1', 'password2',
                'first_name', 'last_name', 'email']

from django import forms

PRODUCT_QUANTITY_CHOICES = [(i, str(i)) for i in range(1, 21)]

class CartAddProductForm(forms.Form):
    quantity =
forms.TypedChoiceField(choices=PRODUCT_QUANTITY_CHOICES,
coerce=int, widget=forms.Select(attrs={'class': 'form-select'}))
    update = forms.BooleanField(required=False, initial=False,
widget=forms.HiddenInput)

from django import forms

class PostMessageForm(forms.Form):
    first_name = forms.CharField(max_length=255, label='Ваше имя*',
widget=forms.TextInput(attrs={'class': 'contact_input'}))
    last_name = forms.CharField(max_length=255, label='Ваша фамилия*',
widget=forms.TextInput(attrs={'class': 'contact_input'}))
    subject = forms.CharField(max_length=255, label='Тема письма*',
widget=forms.TextInput(attrs={'class': 'contact_input'}))
    message = forms.CharField(widget=forms.Textarea(attrs={'cols': 60,
'rows': 10, 'class': 'contact_input contact_textarea'}), label="Текст
письма*")
```


Продолжение приложения Б

Код программы платежной системы на JavaScript

Код API:

```
var halyk;

(function(halyk) {

    var isTest = true;

    var testConfig = {
        pageUrL : "https://test-epay.homebank.kz/payform/",
        origin: "https://test-epay.homebank.kz",
        TokenAPIConfig : {
            url: "https://testoauth.homebank.kz/epay2/oauth2/token",
            clientId: "test"
        }
    };

    var prodConfig = {
        pageUrL : "https://epay.homebank.kz/payform/",
        origin: "https://epay.homebank.kz",
        TokenAPIConfig : {
            url: "https://epay-oauth.homebank.kz/oauth2/token",
            clientId: "uberflower"
        }
    };

    halyk.Config = function Config() {
        if (isTest)
            return testConfig;
        else
            return prodConfig;
    }

    var pageUrL = halyk.Config().pageUrL;

    var paymentPageOrigin = halyk.Config().origin;

    function pay(params) {
```

Продолжение приложения Б

```
location.href = pageUrl + "?params=" +
```

```
LZString.compressToEncodedURIComponent(encodeParams(params));
}
var paymentWidgedCallBack = undefined;
var widgetNode = undefined;
var LZString = function() {function o(o, r) {if (!t[o]) {t[o] = {}}; for (var n
= 0; n < o.length; n++) {t[o][o.charAt(n)] = n; }}return t[o][r]; }var r =
String.fromCharCode, n =
"ABCDEFGHJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz01
23456789+/", e =
"ABCDEFGHJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz01
23456789+-$", t = {}, i = {compressToBase64: function(o) {if (null == o)
{return""; }var r = i._compress(o, 6, function(o) {return n.charAt(o); });
switch (r.length % 4) {default: case 0: return r; case 1: return r + "==="; case
2: return r + "=="; case 3: return r + "="; }}, decompressFromBase64:
function(r) {return null == r ? "" : "" == r ? null : i._decompress(r.length, 32,
function(e) {return o(n, r.charAt(e)); }); }, compressToUTF16: function(o)
{return null == o ? "" : i._compress(o, 15, function(o) {return r(o + 32); }) +
" "; }, decompressFromUTF16: function(o) {return null == o ? "" : "" == o ?
null : i._decompress(o.length, 16384, function(r) {return o.charCodeAt(r) -
32; }); }, compressToUint8Array: function(o) {for (var r = i.compress(o), n
= new Uint8Array(2 * r.length), e = 0, t = r.length; t > e; e++) {var s =
r.charAt(e); n[2 * e] = s >>> 8, n[2 * e + 1] = s % 256; }return n; },
decompressFromUint8Array: function(o) {if (null === o || void 0 === o)
{return i.decompress(o); }for (var n = new Array(o.length / 2), e = 0, t =
n.length; t > e; e++) {n[e] = 256 * o[2 * e] + o[2 * e + 1]; }var s = []; return
n.forEach(function(o) {s.push(r(o)); }), i.decompress(s.join(""))); },
compressToEncodedURIComponent: function(o) {return null == o ? "" :
i._compress(o, 6, function(o) {return e.charAt(o); }); },
decompressFromEncodedURIComponent: function(r) {return null == r ? "" :
"" == r ? null : (r = r.replace(/ /g, "+"), i._decompress(r.length, 32,
function(n) {return o(e, r.charAt(n)); })); }, compress: function(o) {return
i._compress(o, 16, function(o) {return r(o); }); }, _compress: function(o, r,
n) {if (null == o) {return""; }var e, t, i, s = {}, p = {}, u = "", c = "", a = "", l
= 2, f = 3, h = 2, d = [], m = 0, v = 0; for (i = 0; i < o.length; i += 1) {if (u =
o.charAt(i), Object.prototype.hasOwnProperty.call(s, u) || (s[u] = f++, p[u] =
!0), c = a + u, Object.prototype.hasOwnProperty.call(s, c)) {a = c; } else {if
(Object.prototype.hasOwnProperty.call(p, a)) {if (a.charCodeAt(0) < 256)
{for (e = 0; h > e; e++) {m <<= 1, v == r - 1 ? (v = 0, d.push(n(m)), m = 0) :
v++; }for (t = a.charCodeAt(0), e = 0; 8 > e; e++) {m = m << 1 | 1 & t, v ==
r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++, t >>= 1; } } else {for (t = 1, e = 0;
```

Продолжение приложения Б

```
h > e; e++) { m = m << 1 | t, v == r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++, t
= 0; }for (t = a.charCodeAt(0), e = 0; 16 > e; e++) { m = m << 1 | 1 & t, v ==
r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++, t >>= 1; } }l--, 0 == 1 && (l =
Math.pow(2, h), h++), delete p[a]; } else { for (t = s[a], e = 0; h > e; e++) { m
= m << 1 | 1 & t, v == r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++, t >>= 1; }
}l--, 0 == 1 && (l = Math.pow(2, h), h++), s[c] = f++, a = String(u); } }if ("
!:= a) {if (Object.prototype.hasOwnProperty.call(p, a)) {if
(a.charCodeAt(0) < 256) {for (e = 0; h
```

```
> e; e++) { m <<= 1, v == r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++; }for (t =
a.charCodeAt(0), e = 0; 8 > e; e++) { m = m << 1 | 1 & t, v == r - 1 ? (v = 0,
d.push(n(m)), m = 0) : v++, t >>= 1; } } else {for (t = 1, e = 0; h > e; e++)
{ m = m << 1 | t, v == r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++, t = 0; }for (t
= a.charCodeAt(0), e = 0; 16 > e; e++) { m = m << 1 | 1 & t, v == r - 1 ? (v =
0, d.push(n(m)), m = 0) : v++, t >>= 1; } }l--, 0 == 1 && (l = Math.pow(2, h),
h++), delete p[a]; } else { for (t = s[a], e = 0; h > e; e++) { m = m << 1 | 1 &
t, v == r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++, t >>= 1; } }l--, 0 == 1 &&
(l = Math.pow(2, h), h++); }for (t = 2, e = 0; h > e; e++) { m = m << 1 | 1 &
t, v == r - 1 ? (v = 0, d.push(n(m)), m = 0) : v++, t >>= 1; }for (;) {if (m
<<= 1, v == r - 1) {d.push(n(m)); break; }v++; }return d.join(""); },
decompress: function(o) {return null == o ? "" : "" == o ? null :
i._decompress(o.length, 32768, function(r) {return o.charCodeAt(r); }); },
_decompress: function(o, n, e) {var t, i, s, p, u, c, a, l, f = [], h = 4, d = 4, m =
3, v = "", w = [], A = {val: e(0), position: n, index: 1}; for (i = 0; 3 > i; i +=
1) {f[i] = i; }for (p = 0, c = Math.pow(2, 2), a = 1; a != c;) {u = A.val &
A.position, A.position >>= 1, 0 == A.position && (A.position = n, A.val =
e(A.index++)), p |= (u > 0 ? 1 : 0) * a, a <<= 1; }switch (t = p) {case 0: for
(p = 0, c = Math.pow(2, 8), a = 1; a != c;) {u = A.val & A.position,
A.position >>= 1, 0 == A.position && (A.position = n, A.val =
e(A.index++)), p |= (u > 0 ? 1 : 0) * a, a <<= 1; }l = r(p); break; case 1: for
(p = 0, c = Math.pow(2, 16), a = 1; a != c;) {u = A.val & A.position,
A.position >>= 1, 0 == A.position && (A.position = n, A.val =
e(A.index++)), p |= (u > 0 ? 1 : 0) * a, a <<= 1; }l = r(p); break; case 2:
return ""; }for (f[3] = 1, s = 1, w.push(l);) {if (A.index > o) {return ""; }for (p
= 0, c = Math.pow(2, m), a = 1; a != c;) {u = A.val & A.position, A.position
>>= 1, 0 == A.position && (A.position = n, A.val = e(A.index++)), p |= (u >
0 ? 1 : 0) * a, a <<= 1; }switch (l = p) {case 0: for (p = 0, c = Math.pow(2,
8), a = 1; a != c;) {u = A.val & A.position, A.position >>= 1, 0 ==
A.position && (A.position = n, A.val = e(A.index++)), p |= (u > 0 ? 1 : 0) *
a, a <<= 1; }f[d++] = r(p), l = d - 1, h--; break; case 1: for (p = 0, c =
Math.pow(2, 16), a = 1; a != c;) {u = A.val & A.position, A.position >>= 1,
0 == A.position && (A.position = n, A.val = e(A.index++)), p |= (u > 0 ? 1 :
```

Продолжение приложения Б

```
0) * a, a <= 1; }f[d++] = r(p), l = d - 1, h--; break; case 2: return w.join("");
}if (0 == h && (h = Math.pow(2, m), m++), f[l]) {v = f[l]; } else {if (l !=
d) {return null; }v = s + s.charAt(0); }w.push(v), f[d++] = s + v.charAt(0),
h--, s = v, 0 == h && (h = Math.pow(2, m), m++); } } }; return i; }();
"function" == typeof define && define.amd ? define(function() {return
LZString; }) : "undefined" != typeof module && null != module &&
(module.exports = LZString);
```

```
function encodeParams(params) {

    if (params === undefined || params === null) { return ""; }

    if (typeof params != "object") { return "" + params; }

    var result = [];

    for (var name in params) {
        if(name){
            result.push(name + "=" +
encodeURIComponent(encodeParams(params[name])));
        }
    }

    return result.join("&");
}
function addCssClass() {
var style = document.createElement("style");
    style.type = "text/css";
    var styleClasses = ".widgetScreen {position: absolute; width: 100vw;
height: 100vh; top: 0; bottom: 0; left: 0; right: 0; z-index: 1000; background-
color: rgba(5, 5, 5, 0.5); display: flex; justify-content: center; align-items:
center;}";
    styleClasses += ".iframeBox{border-radius: 4px; position: absolute;
width: 420px; height: 900px; z-index: 1010; background-color: #fff;
padding: 10px; -ms-overflow-style: none; scrollbar-width: none;}";
    styleClasses += `.iframeHolder::-webkit-scrollbar {display: none;}`;
    styleClasses += ".iframeBoxHeader{padding: 0px;}";
    styleClasses += ".iframeBoxHeaderCloseButton{border-radius: 8px;
cursor: pointer; width: 15px; height: 15px; content: 'X'; text-align: center;
float: right; background-color: #ccc; font-family: Arial;}";
    styleClasses += ".iframeBoxHeaderCloseButtonText{font-size: 10px;
```

Продолжение приложения Б

```
font-family: sans-serif; font-weight: bold; color: #fff; padding-top: 2px; }";
    styleClasses += ".iframeBoxHeaderLabel{ height:30px; text-align:
center; float: left; }";
    styleClasses += ".iframeClass{ width: 100%; height: 100%; border:
none; }";
    styleClasses += ".iframeHolder{ width: 100%; height: 100%; }";
    style.innerHTML = styleClasses;
    document.getElementsByTagName("head")[0].appendChild(style);
};
function onCloseDialog(result) {
    paymentWidgedCallBack({ success: result });
}

document.getElementsByTagName("body")[0].removeChild(widgetNode);
    widgetNode = undefined;
}

function onCommandRecieved(evnt) {
    if (evnt.origin.indexOf(paymentPageOrigin) === 0) {
        const resultObject = JSON.parse(evnt.data);
        onCloseDialog(resultObject.success === true);
    }
}

function showPaymentWidget(params, callBack) {
    paymentWidgedCallBack = callBack;
    if (!widgetNode) {
        addCssClass();
        widgetNode = document.createElement("DIV");
        widgetNode.className = "widgetScreen";
        var iframeBox = document.createElement("DIV");
        //var iframeBoxHeader = document.createElement("DIV");
        //var iframeBoxLabel = document.createElement("DIV");
        //var iframeBoxCloseButton = document.createElement("DIV");
        //iframeBoxLabel.className = "iframeBoxHeaderLabel";
        //iframeBoxCloseButton.className =
"iframeBoxHeaderCloseButton";
        //iframeBoxLabel.innerHTML = "";
        //var iframeBoxHeaderCloseButtonText =
document.createElement("DIV");
        //iframeBoxHeaderCloseButtonText.innerHTML = "X";
        //iframeBoxHeaderCloseButtonText.className =
```

Продолжение приложения Б

```
"iframeBoxHeaderCloseButtonText";

//iframeBoxCloseButton.appendChild(iframeBoxHeaderCloseButtonText);
// iframeBoxCloseButton.addEventListener("click", function(){
//   onCloseDialog(false)
// });
//iframeBoxHeader.appendChild(iframeBoxLabel);
//iframeBoxHeader.appendChild(iframeBoxCloseButton);
//iframeBoxHeader.className = "iframeBoxHeader";
iframeBox.className = "iframeBox";
var iframe = document.createElement("IFRAME");
var iframeHolder = document.createElement("DIV");
iframeHolder.className = "iframeHolder";
iframeHolder.appendChild(iframe);
//iframeBox.appendChild(iframeBoxHeader);
iframeBox.appendChild(iframeHolder);
iframe.src = halyk.Config().pageUrL + "?params=" +
LZString.compressToEncodedURIComponent(encodeParams(params)) +
'&isShortForm=true';
iframe.className = "iframeClass";
window.addEventListener("message", onCommandRecieved, false);
widgetNode.appendChild(iframeBox);

document.getElementsByTagName("body")[0].appendChild(widgetNode);
}

}

function p2p(params) {
    location.href = pageUrL + "?params=" +
LZString.compressToEncodedURIComponent(encodeParams(params)) +
'&isTransfer=true';
}

halyk.p2p = p2p;

    halyk.pay = pay;
    halyk.showPaymentWidget = showPaymentWidget;

})(halyk || (halyk = {}));
```

Продолжение приложения Б

Код скрипта по подключению к платежной системе

```
function functionToExec() {
  const price = JSON.parse(document.getElementById('price-
data').textContent);
  const code = JSON.parse(document.getElementById('code-
data').textContent);
  const URL = 'http://e638-95-56-25-214.ngrok.io'
  // 4003039802969821
  $.post('https://testoauth.homebank.kz/epay2/oauth2/token',
  {
    grant_type: "client_credentials",
    scope: "payment",
    client_id: "test",
    client_secret: "yF587AV9Ms94qN2QShFzVR3vFnWkhjbAK3sG",
    invoiceID: code,
    amount: price,
    currency: "KZT",
    terminal: "67e34d63-102f-4bd1-898e-370781d0074d",
    postLink: URL + 'cart/check/',
    failurePostLink: URL + 'cart/check/'
  }, function(data){
    console.log(data);
    var createPaymentObject = function (auth, invoiceId, amount) {
      var paymentObject = {
        invoiceId: invoiceId,
        backLink: URL,
        failureBackLink: URL,
        postLink: URL + 'cart/check/',
        failurePostLink: URL + 'cart/check/',
        language: "RU",
        description: "Оплата в интернет-магазине",
        accountId: "testuser1",
        terminal: "67e34d63-102f-4bd1-898e-370781d0074d",
        amount: amount,
        currency: "KZT",
        phone: "",
        email: "",
        cardSave: true
      };
      paymentObject.auth = auth;
```

Продолжение приложения Б

```
        return paymentObject;
    };
    halyk.pay(createPaymentObject(data, code, price));
});
}
```


РЕЦЕНЗИЯ

на дипломный проект студента 4 курса Казахского национального
исследовательского технического университета им. К.И. Сатпаева
специальности 5B070400 «Вычислительная техника и программное
обеспечение»

Изтай Зарина Жақайқызы

на тему «Разработка интернет-магазина для компании Gydos»

Структура дипломной работы включает в себя: введение, три раздела, заключение, список используемых источников литературы и приложений, а также 26 рисунков.

Во введении определяется актуальность выбранной темы, цели и задачи исследования, объект и предмет, обосновывается цель разработки проекта.

В первой главе настоящего дипломного проекта исследуется основное понятие интернет-магазина, проводится анализ по аналоговым системам, представлен обзор по средам разработки и используемым технологиям, а также на основании проведенного исследования рынка, описана значимость и актуальность работы.

Во второй главе рассматривается проектирование системы, определяются функциональные требования к системе, проектируется диаграмма базы данных и интерфейса веб-приложения.

В третьей главе описывается разработка интернет-магазина, среда разработки, демонстрация разработанного функционала в виде рисунков. Для разработки веб-приложения выбран фреймворк Django, в качестве системы управления базами данных выбрана PostgreSQL, интерфейс представляет собой веб-страницы, разработанные с помощью технологий HTML, CSS, языка JavaScript и свободного набора инструментов Bootstrap 5.

В заключение приведены выводы о проделанной работе.

Замечания: имеются технические и стилистические ошибки в тексте дипломного проекта.

Общее заключение и оценка: В целом работа представлена завершённой и может быть оценена на «отлично», а при успешной защите Изтай З. достойна присвоения степени бакалавра техники и технологий по специальности 5B070400 «Вычислительная техника и программное обеспечение».

Рецензент:

Доктор PhD, директор института
информационных технологий НАО
Алматинский университет

энергетики и связи имени Г. Даукеева

Қолтаңбаны растаймын
Подпись заверяю

Алматы	
Қызметі	аты-жөні
« 20 »	05 2022 ж.



Досжанова А.А.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет
имени К.И.Сатпаева

Институт автоматки и информационных технологий

Изтай Зарина Жакайқызы

5B070400 – Вычислительная техника и программное обеспечение

ОТЗЫВ НАУЧНОГО РУКОВОДИТЕЛЯ

к дипломному проекту

Тема: «Разработка интернет-магазина для компании Gydos».

Представленный дипломный проект довольно содержателен и соответствует выданному заданию. Изтай Зарина проявила обширную заинтересованность, а также компетентность при разработке веб-приложения. Все поставленные задачи были выполнены.

Структура дипломной работы включает в себя: введение, три раздела, заключение, список используемых источников литературы и приложений, а также 26 рисунков.

Во введении определяется актуальность выбранной темы, цели и задачи исследования, обосновывается цель разработки проекта.

В первой главе исследуется основное понятие интернет-магазина, проводится анализ по аналоговым системам, представлен обзор по средам разработки и технологиям, а также на основании проведенного исследования рынка, описана значимость и актуальность работы.

Во второй главе рассматривается проектирование системы, определяются функциональные требования к системе, приведены диаграммы базы данных и интерфейса веб-приложения.

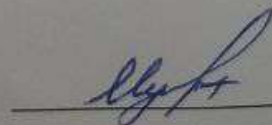
В третьей главе описывается разработка интернет-магазина, демонстрация разработанного функционала в виде рисунков.

В заключении приведены выводы о проделанной работе.

Пояснительная записка соответствует требованиям, предъявленным к данным видам работ. Считаю, что в целом работа представлена завершенной и может быть оценена на «хорошо», а при успешной защите Изтай З. достойна присвоения степени бакалавра техники и технологий.

Научный руководитель:

Ассоциированный профессор



Мукажанов Н.К.

« 18 » 05 2022 г.