СОДЕРЖАНИЕ

Введение 3

1 Выбор технологии, языка и среды программирования 5

2 Анализ и уточнение требований к программному продукту 7

2.1 Выбор методов и разработка основных алгоритмов решения задания 7

3 Разработка структурной схемы программного продукта 13

3.1 Описание используемых процедур и библиотечных функций 13

3.2 Спецификация программы 14

4 Проектирование пользовательского интерфейса 16

4.1 Разработка форм ввода-вывода информации 16

Заключение 24

Список используемых источников 25

Приложение А Техническое задание 26

Приложение Б Алгоритм работы системы 36

Приложение В Структурная схема сайта 37

ВВЕДЕНИЕ

В современном мире электронная торговля занимает одно из ведущих мест в ведении бизнеса: она обеспечивает необходимыми товарами и услугами, объединяет континенты и страны. Благодаря ей, товар может быть доставлен в отдаленные уголки мира. В настоящее время термин «электронная коммерция» трактуется по-разному. Например, А. В. Юрасов определяет электронную коммерцию как «сферу экономики, которая включает в себя все финансовые и торговые транзакции, осуществляемые при помощи компьютерных сетей, и бизнес-процессы, связанные с проведением таких транзакций».

Электронная коммерция — это сфера экономики, которая включает в себя все финансовые и торговые транзакции, осуществляемые при помощи компьютерных сетей, и бизнес-процессы, связанные с проведением таких транзакций.

Многие люди очень часто предпочитает покупку товаров в Интернете, нежели в традиционных магазинах. Прежде всего, это объясняется низкими ценами, а также более широким ассортиментом товара. Помимо этого, затрата времени и нервов на совершение покупки посредством интернет-технологий гораздо меньше. На сегодняшний день функционируют и развиваются такие отрасли предпринимательства, которые наиболее соответствуют интересам и потребностям населения.

Целью курсового проекта является необходимость разработки программного продукта, который позволит реализовать торговлю при помощи сети интернет.

Программный продукт должен быть разработан в виде веб-приложения и обеспечивать возможность выполнения перечисленных ниже функций:

1. Просмотр категорий в каталоге товаров;
2. Просмотр конкретного товара в определенной категории;
3. Добавление, удаление и редактирование информации о категориях продаваемого товара;
4. Добавление, удаление и редактирование информации о продаваемом товаре;
5. Добавление товара в корзину;
6. Редактирование контактных данных клиентом;
7. Авторизация и регистрация клиентов;
8. Формирование заказов;
9. Постраничная навигация по страницам;
10. Оплата покупок в дебетовой электронной платежной системе PayPal;
11. Просмотр квитанций заказов;
12. Печать квитанций;
13. Редактирование и удаление заказов;
14. Поиск товаров, категорий, заказов, клиентов в контроль панели;
15. Редактирование информации о компании.

В веб-приложении должна быть предусмотрена система разграничения прав доступа, которая делится на две группы пользователей: клиентов и администраторов.

1 ВЫБОР ТЕХНОЛОГИИ, ЯЗЫКА И СРЕДЫ ПРОГРАММИРОВАНИЯ

После анализа большого количества сред программирования, было принято решение выбрать среду программирования «Visual Studio Code». «Visual Studio Code» — редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб и облачных приложений. Включает в себя отладчик, инструменты для работы с Git, подсветку синтаксиса, IntelliSense и средства для рефакторинга. Имеет широкие возможности для настройки: пользовательские темы, сочетания клавиш и файлы конфигурации. Распространяется бесплатно, разрабатывается как программное обеспечение с открытым исходным кодом.

Объектно-ориентированное программирование (ООП) — было выбрано для упрощения работы с данными и методами обьекта.

Для разметки структуры веб-приложения используется язык гипертекстовой разметки HTML, чтобы придать внешний вид интерфейсу приложения – язык таблиц стилей CSS. Для реализации корзины товаров выбрана библиотека JavaScript под названием jQuery и AJAX – технология обращения к серверу без перезагрузки страницы.

Основной технологией в разработке данного программного продукта будет язык программирования PHP – скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений.

Язык PHP, в первую очередь, был выбран из-за его кроссплатформенности и возможность запуска в любой операционной системе. Также он поддерживает большинство ведущих веб-серверов, что позволяет без проблем установить веб-приложение на любой хостинг в интернет. Статистические использования и рыночная доля данного языка программирования дает представление о его огромной популярности среди разработчиков, PHP занимает около 80% рынка среди наиболее широко применяемых серверных языков программирования, таких как:

* ASP.NET;
* Java:
* Python;
* Ruby;
* JavaScript.

PHP распространяется в виде бесплатного программного обеспечения по собственной лицензии. Является достаточно легким в освоении серверным языком, имеющим крупное сообщество разработчиков.

Для управления базами данных было принято решение использовать СУБД MySQL. Эта серверная система способна эффективно функционировать во взаимодействии с интернет-сайтами и веб-приложениями. При этом MySQL очень проста в освоении. Выбрана данная СУБД была из-за приведенных ниже преимуществ перед другими системами.

* Простота в использовании. Достаточно легко устанавливается, имеет множество плагинов и вспомогательных приложений упрощающих работу с базами данных;
* Обширный функционал. Обладает всем необходимым инструментарием, который может понадобиться в реализации практически любого проекта;
* Безопасность. Система изначально создана таким образом, что множество встроенных функций безопасности в ней работают по умолчанию;
* Масштабируемость. MySQL одинаково легко может быть использована для работы с малыми и большими объемами данных;
* Скорость. Высокая производительность системы обеспечивается за счет упрощения некоторых используемых в ней стандартов.

2 АНАЛИЗ И УТОЧНЕНИЕ ТРЕБОВАНИЙ К ПРОГРАММНОМУ ПРОДУКТУ

2.1 Выбор методов и разработка основных алгоритмов решения задачи

При проектировании программного продукта была разработана схема диаграммы прецедентов, предназначенная для отображения связи между программным продуктом и пользователями.

Диаграмма прецедентов изображена на рисунке 1.



Рисунок 1 – Диаграмма прецедентов

Для отображения связи между сущностями в базе данных программного продукта, была разработана ER-диаграмма, изображенная на рисунке 2.

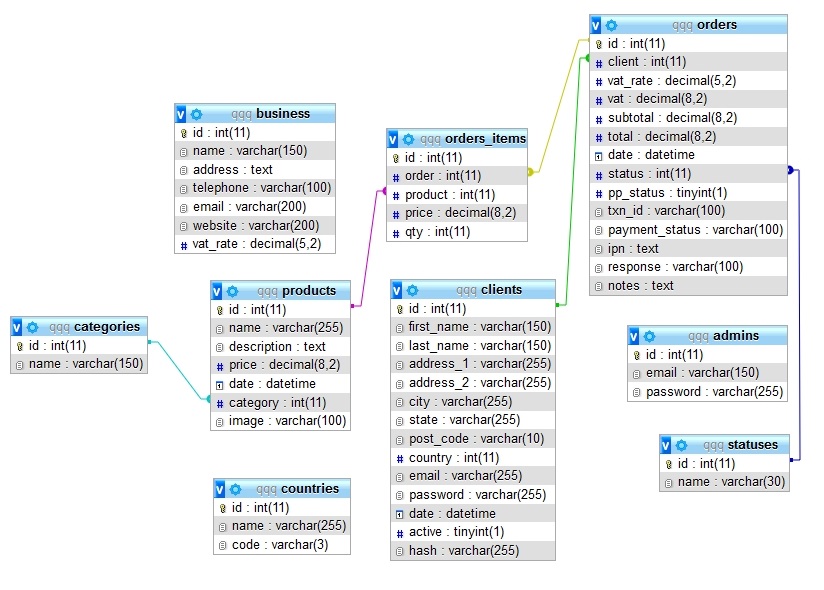


Рисунок 2 – ER-диаграмма

Список сущностей и их краткое описание представлено в таблице 1.

Таблица 1 – Список сущностей веб-приложения

|  |  |
| --- | --- |
| Наименование | Описание сущности |
| clients | Содержит в себе информацию о клиентах |
| orders | Содержит в себе информацию о заказах |
| products | Содержит в себе информацию о товарах |
| orders\_items | Содержит в себе информацию о заказанных товарах |
| categories | Содержит в себе информацию о категориях товаров |
| statuses | Содержит в себе информацию о статусах заказов |
| admins | Содержит в себе информацию об администраторах |

Продолжение таблицы 1

|  |  |
| --- | --- |
| Наименование | Применение |
| countries | Содержит в себе информацию о странах |
| business | Содержит в себе информацию о веб-приложении |

Набор и подробное описание свойств сущностей представлено в таблицах 2-10.

Таблица 2 – Сущность «clients»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор пользователя | int (11) | Первичный ключ.  Обязательное |
| first\_name | Имя пользователя | varchar (150) | Обязательное |
| last\_name | Фамилия пользователя | varchar (150) | Обязательное |
| address\_1 | Адрес проживания | varchar (255) | Обязательное |
| address\_2 | Дополнительный адрес проживания | varchar (255) | Обязательное |
| city | Город проживания | varchar (255) | Обязательное |
| state | Область | varchar (255) | Обязательное |
| post\_code | Почтовый индекс | varchar (10) | Обязательное |
| country | Страна проживания | int (11) | Внешний ключ.  Обязательное |
| email | Электронная почта | varchar (255) | Обязательное |
| password | Пароль | varchar (255) | Обязательное |
| date | Дата регистрации | datetime | Обязательное |
| active | Флаг активации аккаунта | tinyint (1) | Не обязательное |
| hash | Уникальная хэш-функция | varchar (255) | Обязательное |

Таблица 3 – Сущность «orders»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор заказа | int (11) | Первичный ключ.  Обязательное |
| client | Идентификатор пользователя | int (11) | Внешний ключ.  Обязательное |
| vat\_rate | Процент НДС | decimal (5,2) | Обязательное |
| vat | Сумма НДС | decimal (8,2) | Обязательное |
| subtotal | Сумма без НДС | decimal (8,2) | Обязательное |
| total | Сумма с НДС | decimal (8,2) | Обязательное |
| date | Дата заказа | datetime | Обязательное |
| status | Статус заказа | int (11) | Внешний ключ.  Обязательное |
| pp\_status | Статус оплаты | tinyint (1) | Обязательное |
| txn\_id | Идентификатор транзакции PayPal | varchar (100) | Не обязательное |
| payment\_status | Статус оплаты со стороны PayPal | varchar (100) | Не обязательное |
| ipn | Логи, полученные от PayPal после оплаты заказа | text | Не обязательное |
| response | Статус IPN PayPal | varchar (100) | Не обязательное |
| notes | Примечания | text | Не обязательное |

Таблица 4 – Сущность «products»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор товара | int (11) | Первичный ключ.  Обязательное |
| name | Название товара | varchar (255) | Обязательное |
| description | Описание товара | text | Обязательное |
| price | Цена товара | decimal (8,2) | Обязательное |
| date | Дата записи в БД | datetime | Обязательное |
| category | Категория товара | int (11) | Внешний ключ.  Обязательное |
| image | Имя файла изображения | varchar (100) | Не Обязательное |

Таблица 5 – Сущность «orders\_items»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор заказанного товара | int (11) | Первичный ключ.  Обязательное |
| order | Идентификатор заказа | int (11) | Внешний ключ.  Обязательное |
| product | Идентификатор товара | int (11) | Внешний ключ.  Обязательное |
| price | Цена товара | decimal (8,2) | Обязательное |
| qty | Количество товара | int (11) | Обязательное |

Таблица 6 – Сущность «categories»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор категории | int (11) | Первичный ключ.  Обязательное |
| name | Название категории | varchar (15) | Обязательное |

Таблица 7 – Сущность «statuses»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор статуса | int (11) | Первичный ключ.  Обязательное |
| status | Наименование статуса | varchar (30) | Обязательное |

Таблица 8 – Сущность «admins»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор администратора | int (11) | Первичный ключ. Обязательное |
| email | Логин администратора | varchar (150) | Обязательное |
| password | Пароль | varchar (255) | Обязательное |

Таблица 9 – Сущность «countries»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор страны | int (11) | Первичный ключ.  Обязательное |
| name | Название страны | varchar (255) | Обязательное |
| code | Код страны | varchar (3) | Не обязательное |

Таблица 10 – Сущность «business»

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Описание | Тип | Примечание |
| id | Идентификатор бизнес профиля | int (11) | Первичный ключ.  Обязательное |
| name | Название компании | varchar (150) | Обязательное |
| address | Адрес компании | text | Обязательное |
| telephone | Телефон компании | varchar (100) | Обязательное |
| email | Адрес электронной почты | varchar (200) | Обязательное |
| website | Веб-сайт компании | varchar (200) | Обязательное |
| vat\_rate | Процент НДС | decimal (5,2) | Обязательное |

Для разъяснения процессов, протекающих в программном продукте, была разработана блок-схема работы системы, которая представлена в Приложении Б.

3 РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ ПРОГРАММНОГО ПРОДУКТА

3.1 Описание используемых процедур и библиотечных функций

Для реализации поставленной задачи в программном продукте была использована библиотека jQuery – библиотека языка программирования JavaScript, фокусирующаяся на взаимодействии JavaScript и HTML. Она позволяет легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими.

Также для реализации автоматической отправки электронных писем через код PHP с веб-сервера, содержащих в себе ссылку, необходимую для подтверждения аккаунта, была использована библиотека PHPMailer.

Список используемых классов и их назначение, используя язык PHP, представлен в таблице 2.

Таблица 2 – Список классов веб-приложения

|  |  |
| --- | --- |
| Имя класса | Назначение |
| Admin | Содержит в себе методы для работы с контроль-панелью администратора |
| Application | Содержит в себе конструктор класса для подключения к базе данных веб-приложения |
| Business | Содержит в себе методы для работы с информацией о веб-приложении |
| Cart | Содержит в себе методы для работы с корзиной покупок |
| Catalog | Содержит в себе методы для работы с категориями товаров |
| Core | Содержит в себе метод запуска буферизации и получения текущего url |
| Country | Содержит в себе методы для работы со списком стран |
| Dbase | Содержит в себе методы для работы с базой данных |

Продолжение таблицы 2

|  |  |
| --- | --- |
| Имя класса | Назначение |
| Email | Содержит в себе методы для работы с smtp протоколом (отправка электронной почты) |
| Form | Содержит в себе методы для работы с формами и массивом |
| Helper | Содержит в себе методы вспомогательные методы, такие как: перенаправление на другую страницу, установка даты, энкодинг HTML в кодировку UTF-8, удаление из строки специальных символов |
| Login | Содержит в себе методы для работы с авторизацией |
| Order | Содержит в себе методы для работы с заказами |
| Paging | Содержит в себе методы для работы с постраничной навигацией веб-приложения |
| PayPal | Содержит в себе методы для работы с дебетовой электронной платежной системой PayPal |
| Session | Содержит в себе методы для работы с сессией |
| Upload | Содержит в себе методы, позволяющие загрузить файл изображение на сервер |
| Url | Содержит в себе методы для работы с url веб-приложения |
| User | Содержит в себе методы для работы с информацией о клиентах |
| Validation | Содержит в себе методы для работы с валидацией форм |

3.2 Спецификация программы

В ходе работы над курсовым проектом был выделен основной список исполняемых файлов веб-приложения электронной коммерции, представленный в таблице 11.

|  |  |  |
| --- | --- | --- |
| Наименование | Назначение | Примечание |
| index.php | Файл расширения PHP | Файл, запускающий ядро веб-приложения, отображает главную страницу |
| cart.php | Файл расширения PHP | Файл, обеспечивающий работу корзины покупок |
| paypal.php | Файл расширения PHP | Файл работы с платежной системой PayPal |
| cart.js | Файл расширения JS | Скрипты, необходимые для работы ajax с корзиной покупок |
| autoload.php | Файл расширения PHP | Файл, загружающийся на всех страницах. Необходим для подключения классов и конфигурационного файла |
| config.php | Файл расширения PHP | Конфигурационный файл, содержащий в себе пути к директориям, записанные в константы. Запускает сессию |
| core.css | Файл расширения CSS | Файл, содержащий каскадные стили CSS для всех страниц веб-приложения |
| invoice.css | Файл расширения CSS | Файл, содержащий стили для квитанций заказов |
| add.php | Файл расширения PHP | Файл страницы удаления записей из базы данных |
| edit.php | Файл расширения PHP | Файлы страницы редактирования записей |
| list.php | Файл расширения PHP | Файл страницы, на которой отображается полный список записей из базы данных |
| remove.php | Файл расширения PHP | Файл страницы удаления записей из базы данных |
| catalog.php | Файл расширения PHP | Файл страницы каталога товаров |
| login.php | Файл расширения PHP | Файл страницы авторизации и регистрации пользователей |
| invoice.php | Файл расширения PHP | Файл страницы с квитанцией |
| checkout.php | Файл расширения PHP | Файл оформления покупок |

4 ПРОЕКТИРОВАНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

4.1 Разработка форм ввода-вывода информации

В ходе проектирования веб-приложения электронной коммерции, была разработана схема древовидной структура сайта, необходимая для удобства навигации по веб-приложению. Структурная схема сайта представлена в Приложении В.

На главной странице веб-приложения, пользователю предоставляется возможность выбора категорий товаров, товаров в выбранной категории и добавление их в корзину, просмотр корзины, перехода к корзине, оформлению покупок, авторизации и регистрации. Главная страница изображена на рисунке 3.

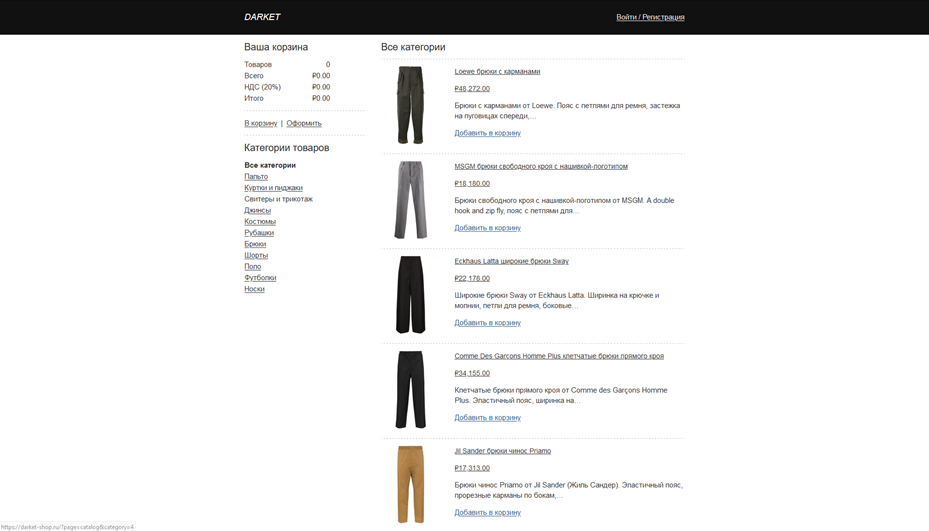


Рисунок 3 – Главная страница веб-приложения

Страница авторизации и регистрации была разработана с целью дать возможность пользователям авторизоваться, или если они еще не были зарегистрированы ранее, зарегистрироваться. На странице присутствуют поля ввода личной информации и кнопки для отправки данных на сервер. Страница авторизации и регистрации изображена на рисунке 4.

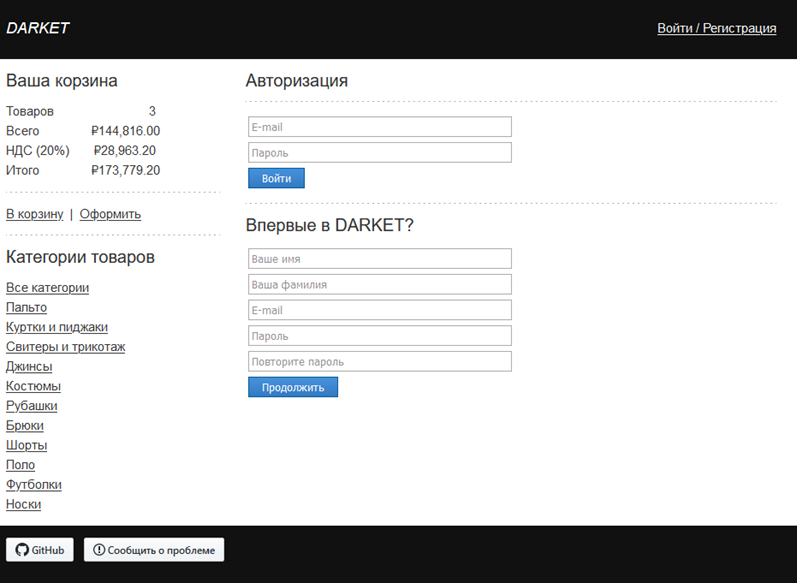


Рисунок 4 – Страница авторизации и регистрации пользователей

Страница корзины товаров была разработана для хранения добавленных пользователем товаров в корзину. Здесь в таблице отображается наименование товара, его количество и цена, а также кнопка удаления из корзины. Кнопка «обновить» позволяет обновить количество товара и их цену в корзине. Кнопка «Перейти к оформлению покупки» перенаправляет пользователя на страницу оформления. Корзина товаров изображена на рисунке 5.

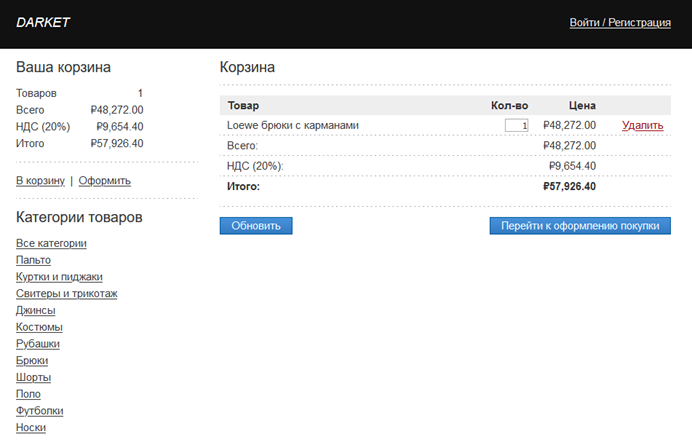


Рисунок 5 – Страница корзины товаров

На странице ввода контактной информации пользователю предлагается ввести данные, необходимые для доставки товаров. Пользователь не сможет перейти к оформлению покупок пока не заполнит все необходимые поля. Страница ввода контактной информации изображена на рисунке 6.

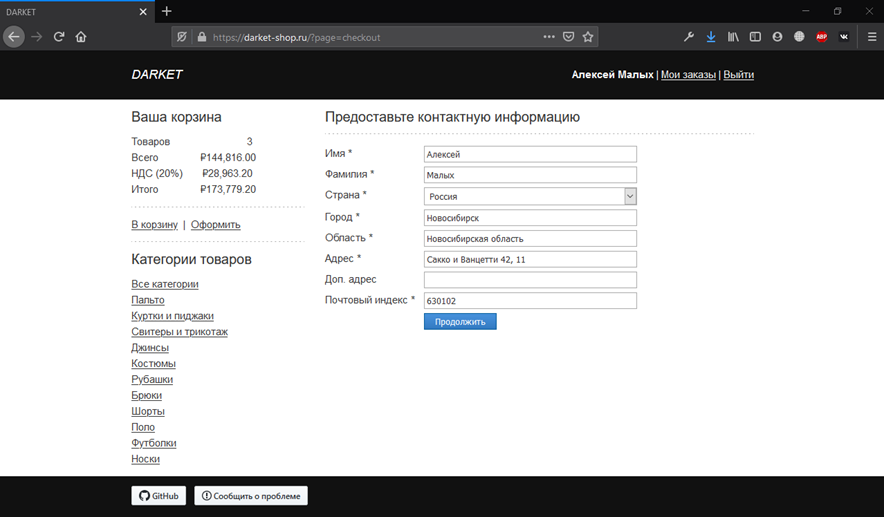


Рисунок 6 – Страница ввода контактной информации

Страница списка заказов была разработана для того, чтобы пользователь имел возможность просмотреть список своих заказов, их статус, дату и распечатать квитанцию. Страница заказов отображена на рисунке 7.

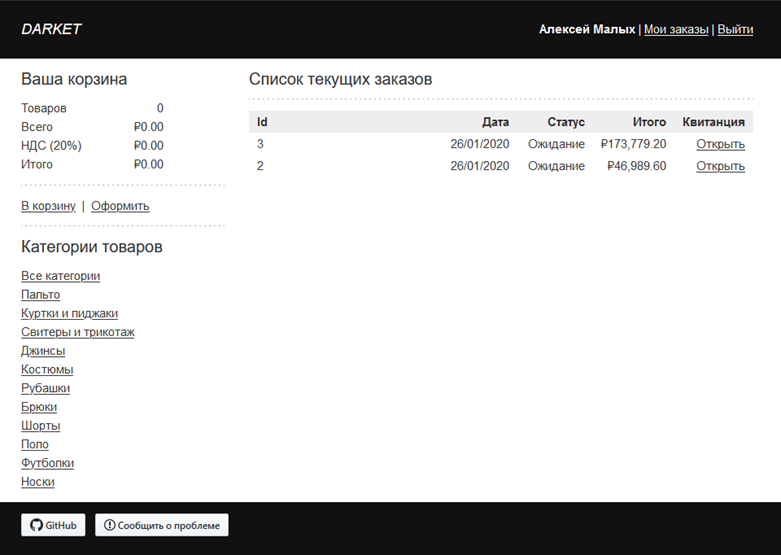


Рисунок 7 – Страница списка заказов

Страница квитанции была разработана с целью дать возможность пользователям просмотреть детали заказа и возможность печати этой квитанции, и которая показана на рисунке 8.

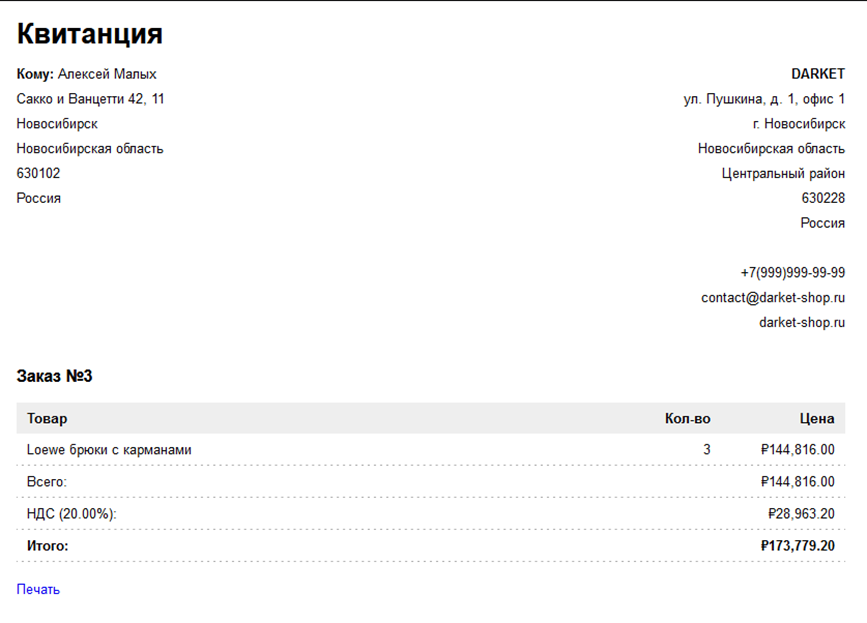


Рисунок 9 – Страница квитанции заказа

Страница авторизации администраторов была разработана с целью ограничения доступа к контроль-панели. Здесь присутствуют поля для ввода логина, пароля и кнопка «Войти», при нажатии на которую происходит авторизация. Страница авторизации администраторов отображена на рисунке 10.

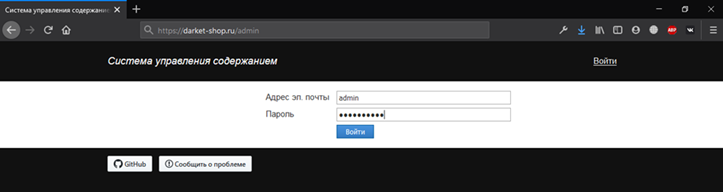


Рисунок 10 – Страница авторизации администраторов

Страница контроль-панели была разработана с целью дать возможность администраторам взаимодействовать с элементами системы управления содержанием:

* производить поиск;
* добавлять записи в базу данных;
* редактировать записи;
* удалять.

Контроль панель представлена на рисунке 11.

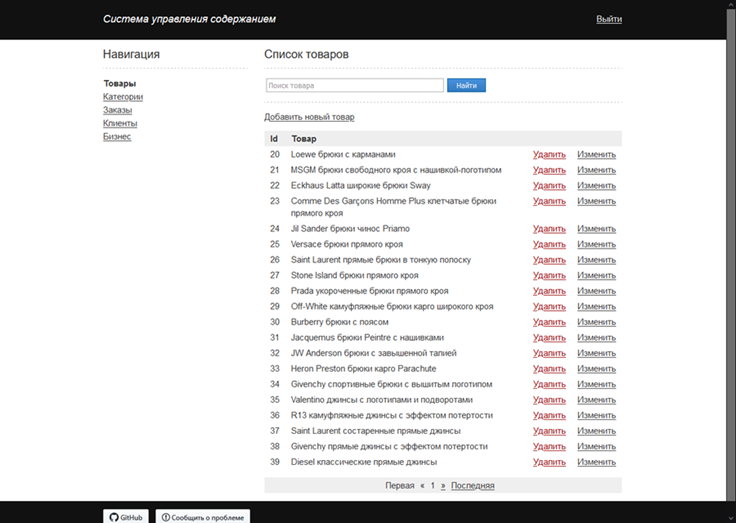


Рисунок 11 – Страница контроль-панели администратора

На странице редактирования администратору предоставляется возможность изменения записи в базе данных. Здесь присутствуют поля ввода, которые могут подвергаться изменениям и кнопки:

* Кнопка «Назад». Позволяет вернуться в контроль-панель;
* Кнопка «Сохранить». Позволяет сохранить запись в базе данных.

Страница редактирования представлена на рисунке 12.

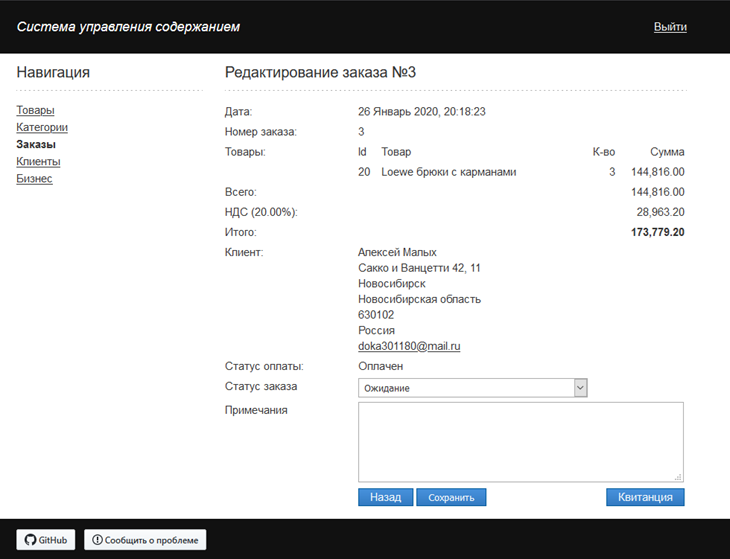


Рисунок 13 – Страница редактирования

Страница удаления была разработана для предотвращения случайного удаления записей из базы данных. На данной странице отображается сообщение с подтверждением удаления и две кнопки: «Да» - согласиться на удаления и «Нет, вернуться в список товаров» - отменить удаление, вернуться в список товаров. Страница удаления изображена на рисунке 14.

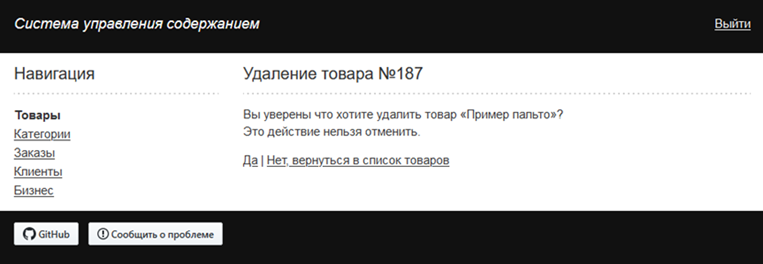


Рисунок 14 – Страница удаления

Страница добавления нужна для того, чтобы администратор имел возможность добавить новые записи в базу данных. На этой странице отображаются поля ввода и кнопки: «Добавить» - добавление новой записи в базу данных и «Назад» - возвращает администратора в контроль-панель. Страница добавления представлена на рисунке 15.

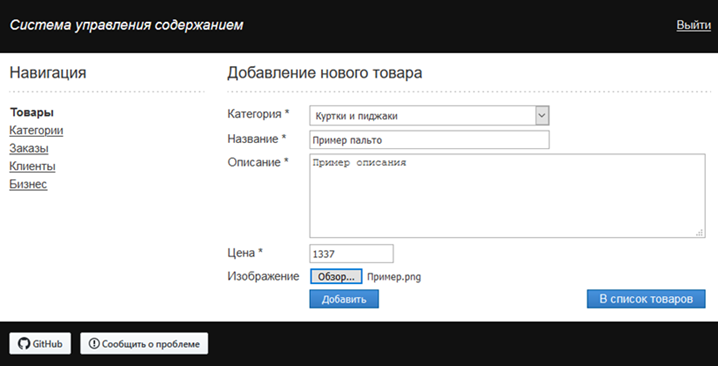


Рисунок 15 – Страница добавления

Страница успешного выполнения какого-либо действия была разработана для того, чтобы дать администратору понять, что выполняемое им действие было успешно завершено. Данная страница открывается после выполнения следующих действий:

* добавления;
* редактирования;
* удаления.

Страница успешного выполнения отображена на рисунке 16.

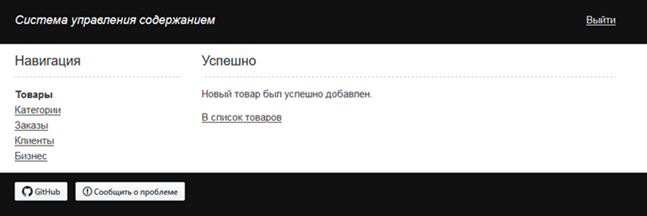


Рисунок 16 – Страница успешного выполнения

ЗАКЛЮЧЕНИЕ

В процессе проектирования курсового проекта было разработано веб-приложение электронной коммерции, а именно интернет-магазин по продаже дизайнерской одежды.

Интернет-магазин по продажам одежды предназначен для продажи различного рода одежды посредством сети интернет. Областью применения данного веб-приложения выступает интернет-торговля.

Интернет-магазин имеет удобный и понятный интерфейс пользователя, благодаря которому не требуется излишних усилий в освоении работы с веб-приложением.

Результаты тестирования показали, что система работает без ошибок, имеется возможность работы с веб-сайтом во всех популярных браузерах. Тестирования проводились в таких браузерах как:

* Google Chrome;
* Chromium;
* Mozilla Firefox;
* Safari;
* Opera;
* Яндекс Браузер.

В результате разработки программного продукта для курсового проекта были приобретены практические умения работы с HTML, CSS, PHP, JavaScript и PayPal.

В дальнейшем планируется расширить функционал веб-приложения: расчет стоимости доставки в зависимости от места назначения (внутри страны или между странами) и веса всех товаров в корзине, добавление склада товаров, добавление возможности выбора пола, размеров одежды, а также разработка нового адаптивного дизайна.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Руководство по PHP: [Электронный ресурс]. – Режим доступа: <https://www.php.net/manual/ru/index.php>

2. Справочник по HTML: [Электронный ресурс]. – Режим доступа: <http://htmlbook.ru/html>

3. Самоучитель по CSS: [Электронный ресурс]. – Режим доступа: <http://htmlbook.ru/samcss>

4. Современный учебник JavaScript: [Электронный ресурс]. – Режим доступа: <https://learn.javascript.ru>

5. Веб-технологии для разработчиков - JavaScript: [Электронный ресурс]. – Режим доступа: <https://developer.mozilla.org/ru/docs/Web/JavaScript>

6. Русская документация по API jQuery: [Электронный ресурс]. – Режим доступа: <https://jquery-docs.ru>

7. Документация к технологии AJAX: [Электронный ресурс]. – Режим доступа: <https://jquery-docs.ru/jQuery.ajax>

8. GitHub репозиторий библиотеки PHPMailer: [Электронный ресурс]. – Режим доступа: <https://github.com/PHPMailer/PHPMailer>

9. Сайт вопросов и ответов для программистов: [Электронный ресурс]. – Режим доступа: <https://stackoverflow.com>

Приложение А

Техническое задание

Министерство образования Новосибирской области ГБПОУ НСО

«Новосибирский авиационный технический колледж

имени Б. С. Галущака»

Разработка веб-приложения электронной коммерции

НАТК.440700.400 ПЗ

Разработал

студент группы ПР-455

Малых А.В.

2020

СОДЕРЖАНИЕ

Введение 3

1 Основания для разработки 4

2 Назначение разработки 4

3 Требования к программному продукту 4

3.1 Требования к функциональным характеристикам 4

3.2 Требования к надежности 6

3.3 Условия эксплуатации 6

3.4 Требования к составу и параметрам технических средств 7

3.5 Требование к защите информации и программ 8

3.6 Требование к маркировке и упаковке 9

3.7 Специальные требования 9

4 Требование к программной документации 9

5 Технико-экономические показатели 9

6 Стадии и этапы разработки 9

7 Порядок контроля и приемки 10

ВВЕДЕНИЕ

Настоящее техническое задание распространяется на разработку веб-приложения электронной коммерции. Данный программный продукт предназначен для эффективной продажи дизайнерской одежды в сети интернет, содержащий в себе полную информацию о компании: чем занимается компания, контактные данные, адрес главного офиса и т.д. Для оплаты покупок клиента используется крупнейшая [дебетовая](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%B1%D0%B5%D1%82" \o "Дебет) [электронная платёжная система](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BB%D0%B0%D1%82%D1%91%D0%B6%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0" \o "Электронная платёжная система) PayPal.

Основными преимуществами интернет-магазина перед оффлайн магазином являются:

* Стоимость аренды. Аренда виртуального приватного сервера намного дешевле, чем аренда помещения под оффлайн магазин;
* Охват аудитории. Заключается в том, что нет ограничений в виде улицы, района или города;
* Постоянная доступность. Интернет-магазин работает 24 часа в сутки и 7 дней в неделю;
* Широкий ассортимент товаров. Для увеличения ассортимента не нужно арендовать склады или дополнительные помещения;
* Автоматизация. Большая часть работы интернет-магазина автоматизирована.

Краткая характеристика области применения: областью применения данного веб-приложения выступает интернет торговля.

Условные обозначения и сокращения:

ООП – объектно-ориентированное программирование.

1 ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

Основанием для проведения разработки является приказ колледжа на выполнение курсового проекта.

Наименование темы разработки – веб-приложение электронной коммерции «DARKET».

2 НАЗНАЧЕНИЕ РАЗРАБОТКИ

Функциональным назначением веб-приложения является информационное обслуживание клиента (покупателя), обработка заказов, проведение платежей с помощью электронной платежной системы PayPal.

3 ТРЕБОВАНИЯ К ПРОГРАММНОМУ ПРОДУКТУ

3.1 Требования к функциональным характеристикам

Программный продукт должен обеспечивать возможность выполнения функций, описанных в таблице 1.

Таблица 1 – Функции веб-приложения

|  |  |
| --- | --- |
| № | Наименование функции |
| 1 | Просмотр категорий в каталоге товаров |
| 2 | Просмотр конкретного товара в определенной категории |
| 3 | Добавление, удаление и редактирование информации о категориях продаваемого товара |
| 4 | Добавление, удаление и редактирование информации о продаваемом товаре |
| 5 | Добавление товара в корзину |
| 6 | Редактирование контактных данных клиентом |
| 7 | Авторизация и регистрация клиентов |
| 8 | Формирование заказов |

Продолжение таблицы 1

|  |  |
| --- | --- |
| № | Наименование функции |
| 9 | Постраничная навигация по страницам веб-приложения |
| 10 | Оплата покупок в дебетовой электронной платежной системе PayPal |
| 11 | Просмотр клиентом квитанций заказов |
| 12 | Просмотр администратором квитанций всех заказов в системе |
| 13 | Печать квитанций |
| 14 | Редактирование и удаление заказов |
| 15 | Поиск товаров, категорий, заказов, клиентов в контроль панели |
| 16 | Редактирование информации о компании |

Входные данные веб-приложения должны быть организованы в виде файлов с информацией о категориях товаров и информацией о товарах, расширение таблиц должно быть ‘.csv’. Таблицы данного формата должны импортироваться в базу данных MySQL с кодировкой UTF-8.

Выходные данные должны быть организованы в виде квитанций с возможностью печати, содержащих в себе информацию о клиенте, его покупках, и компании:

* Имя, фамилия клиента;
* Страна, город, область, адрес доставки и почтовый индекс;
* Название компании;
* Страна, город, область, адрес главного офиса компании;
* Почтовый адрес компании;
* Список покупаемых клиентом товаров, их количество и стоимость с учетом НДС.

Квитанции формируются после создания заказа в приложении на стороне пользователя и хранятся в личном кабинете на странице «Мои заказы».

Файлы формата ‘.csv’ должны быть размещены (храниться) на сервере, на котором будет запускаться данное веб-приложение.

3.2 Требования к надежности

Надежное или устойчивое функционирование веб-приложения должно быть обеспечено выполнением Заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

* Организация бесперебойного питания серверного и коммуникационного оборудования;
* Организация стабильного соединения с сети Интернет;
* Регулярным выполнением требований ГОСТ 51188-98. «Защита информации. Испытания программных средств на наличие компьютерных вирусов».

В веб-приложении должен быть предусмотрен контроль валидности вводимых пользователем данных, а также необходимо предусмотреть защиту от SQL инъекций, то есть должна присутствовать фильтрация специальных символов для форм, с которыми работает клиент для отправки информации в базу данных.

Время восстановления после отказа, вызванного сбоем электропитания технических средств или иными внешними факторами, не фатальным сбоем операционной или файловой системы, не должно превышать времени на перезагрузку задействованных технических и программных средств, при условии соблюдений условий эксплуатации самих технических и программных средств.

3.3 Условия эксплуатации

Пользователь должен иметь навыки практической работы на персональном компьютере с предустановленной Unix-подобной операционной системой (macOS или Linux), или операционной системой Windows.

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

Обслуживание веб-приложения включает в себя:

* Информационное обслуживание – добавление, удаление и редактирование информации в базе данных;
* Системное администрирование базы данных.

Минимальное количество персонала, требуемого для работы веб-приложения, должно составлять не менее двух штатных единиц – администратор и конечный пользователь.

Администратор должен иметь техническое образование, связанное с администрированием серверов. В перечень задач, выполняемых администратором, должны входить:

* Задача поддержания работоспособности технических средств;
* Задача установки и поддержания работоспособности системных программных средств;
* Информационное обслуживание базы данных – добавление, редактирование и удаление информации в базе данных.

3.4 Требования к составу и параметрам технических средств

Сервер базы данных:

* Центральный процессор Intel или AMD с тактовой частотой не ниже 2 ГГц;
* Оперативная память с объемом не меньше 16 Гб;
* Объем свободного дискового пространства не меньше 80 Гб.

Рабочее место пользователя:

* Центральный процессор Intel или AMD с тактовой частотой не ниже 1 ГГц;
* Оперативная память с объемом не меньше 4 Гб;
* Объем свободного дискового пространства не меньше 300 Мб.

3.5 Требование к информационной и программной совместимости

Интерфейс пользователя должен быть интуитивно понятным. Счета-накладные не должны содержать в себе излишнюю информацию о клиенте. Пароли в базе данных должны храниться в виде хеш-функций. Проектирование структуры базы данных должно быть выполнено в рамках разработки технического проекта. На сервере должно быть установлено следующее программное обеспечение:

* MySQL Database;
* Apache HTTP-сервер 2.4.41 и новее;
* PHP 7.2.25 и новее;
* phpMyAdmin 4.9.2 и новее.

Исходный код веб-приложения должен быть реализован с помощью языка гипертекстовой разметки HTML, языка таблиц стилей CSS, JavaScript и PHP. При реализации исходного кода на JavaScript должен использоваться стандарт ECMAScript 6 (ES6), а при реализации на PHP ‒ PHP Standarts Recommendations (PSR). Взаимодействие с системой управления базами данных и создание базы данных реализуется на языке SQL.

3.5 Требование к защите информации и программ

Доступ к информации базы данных на чтение, редактирование или удаление предоставляется только авторизованным пользователям с соответствующими правами доступа. Должна быть предусмотрена защита от SQL инъекций. Для авторизации в веб-приложении необходимо корректно ввести имя пользователя и пароль.

Разграничение доступа к объектам базы данных должно быть реализовано средством веб-приложения.

3.6 Требование к маркировке и упаковке

Требования маркировке и условиям упаковки не предъявляются.

3.8 Специальные требования

Резервное копирование данных веб-приложения и базы данных должно выполняться на сервере автоматически каждые сутки в 00:00 по Московскому времени.

4 ТРЕБОВАНИЕ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

Предварительный состав программной документации должен включать в себя:

* Техническое задание;
* Пояснительную записку;
* Руководство пользователя.

5 ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

Экономические преимущества разработки и ориентировочная экономическая эффективность не рассчитываются.

6 СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

Таблица 2 – Стадии разработки

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование этапа | Срок, даты | Отчетность |
| 1 | Исследование предметной области | 11.11.2019-16.11.2019 | Пояснительная записка |
| 2 | Выбор моделей, описывающих предметную область | 17.11.2019-19.11.2019 | Пояснительная записка |
| 3 | Разработка технического задания | 20.11.2019-29.11.2019 | Техническое задание |
| 4 | Изучение ООП в PHP | 30.11.2019-08.12.2019 | Пояснительная записка |
| 5 | Изучение технологии AJAX | 09.12.2019-22.12.2019 | Пояснительная записка |
| 6 | Анализ требований и уточнение спецификаций | 11.12.2019-23.12.2019 | Спецификации программного обеспечения |
| 7 | Проектирование структуры веб-приложения, проектирование компонентов (технический проект) | 24.12.2019-30.12.2019 | Схема структурная системы и спецификации компонентов |
| 8 | Кодирование серверной части | 12.01.2020-19.01.2020 | Программный продукт |
| 9 | Кодирование клиентского веб-приложения | 20.01.2020-01.02.2020 | Программный продукт |

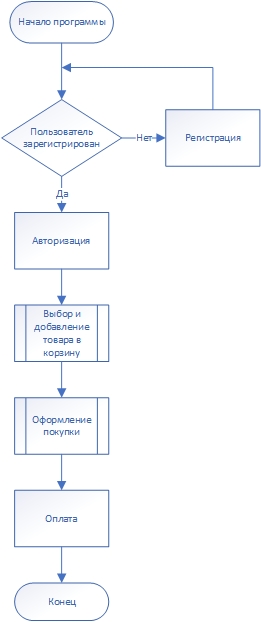
7 ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

Виды испытаний – защита курсового проекта.

Общие требования к приемке:

* Техническое задание;
* Пояснительная записка;
* Программный продукт;
* Презентация.

Приложение Б



Приложение В

