|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. **Исследовательская часть**    1. **Проектирование программного продукта**   **2.1.1 Постановка задачи**  В процессе проектирования индивидуального задания необходимо разработать административную часть интернет-магазина по продаже настольных игр, которая предполагает наличие базы данных и интерфейса для работы с ней.  При разработке программного продукта необходимо учесть следующие требования:  - обеспечение работы в удобной для администратора форме,  - вывод информации на экран,  - добавление информации в базу данных,  - удаление информации из базы данных,  - редактирование информации из базы данных.  **2.1.2 Выбор инструментальных средств разработки**  Для реализации данного продукта были выбраны следующие средства разработки:  - «MySQL» – свободная реляционная система управления базами данных, которая отличается высокой скоростью работы, надежностью, гибкостью.  - PHP – язык программирования общего назначения с открытым исходным кодом, сконструированный специально для веб-разработки,  - HTML – язык гипертекстовой разметки документов, с помощью которого создаются веб-страницы, находящиеся в компьютерной сети Интернет, | | | | | | | | | | | |
|  |  |  |  |  | *ОПП.ПП.* | *508120* | | *ППП* | |  |  |
|  |  |  |  |  |
|  |  |  |  |  | *Проектирование программного продукта* | *Лит.* | | | *Масса* | | *Масштаб* |
| *Изм.* | *Лист* | *№ докум.* | *Подп.* | *Дата* | *У* |  |  |  | |  |
| *Разраб.* | | *Прохоренко Н. С.* |  | *06.12.21* |
| *Пров.* | | *Новикова Е.Е.* |  | *10.12.21* |
|  | |  |  |  | *Лист ( )* | | | | *Листов ( )* | |
|  | |  |  |  | *Отделение «Информационные системы и технологии»* | *Филиал БГТУ «Витебский*  *государственный технологический колледж», гр. 32o* | | | | | |
|  | |  |  |  |
|  | |  |  |  |

- CSS – свод стилевых описаний, тех или иных HTML тегов, который может быть применён как к отдельному тегу, так и одновременно ко всем идентичным элементам на всех страницах сайта, либо только к какой-то их части.

# 2.1.3 Проектирование программного продукта

На основании изученной предметной области разработана структура базы данных интернет-магазина. Перечень отношений базы данных представлен в таблицах 2.1 – 2.6.

# Таблица 2.1 – Таблица «user»

|  |  |  |
| --- | --- | --- |
| **Наименование атрибута** | **Тип атрибута** | **Назначение** |
| id | Int | Идентификатор пользователя, первичный ключ |
| name | Varchar(255) | Имя пользователя |
| email | Varchar(255) | Электронная почта |
| password | Varchar(255) | Пароль |
| role | Varchar(50) | Роль пользователя |

Составлено автором

В таблице 2.1 представлена информация о пользователях и информация необходимая для авторизации на сайте.

# Таблица 2.2 – Таблица «product»

|  |  |  |
| --- | --- | --- |
| **Наименование атрибута** | **Тип атрибута** | **Назначение** |
| id | Int | Идентификатор товара, первичный ключ |
| name | Varchar(255) | Наименование продукта |
| category\_id | int | Идентификатор категории товара |
| code | int | Артикул товара |
| price | float | Цена товара |
| availability | int | Наличие товара на складе |
| brand | Varchar(255) | Производитель |
| description | Varchar(255) | Описание товара |
| is\_new | int | Новизна товара |
| is\_recommended | int | Отношение к списку рекомендуемого |
| status | int | Статус товара |

Составлено автором

В данной таблице 2.2 представлено описание товаров в интернет-магазине.

# Таблица 2.3 – Таблица «category»

|  |  |  |
| --- | --- | --- |
| **Наименование атрибута** | **Тип атрибута** | **Назначение** |
| id | Int | Идентификатор категории, первичный ключ |
| name | Varchar(255) | Наименование категории |
| sort\_order | Int | Отображаемый номер |
| status | Int | Статус категории |

Составлено автором

В таблице 2.3 представлены категории интернет-магазина.

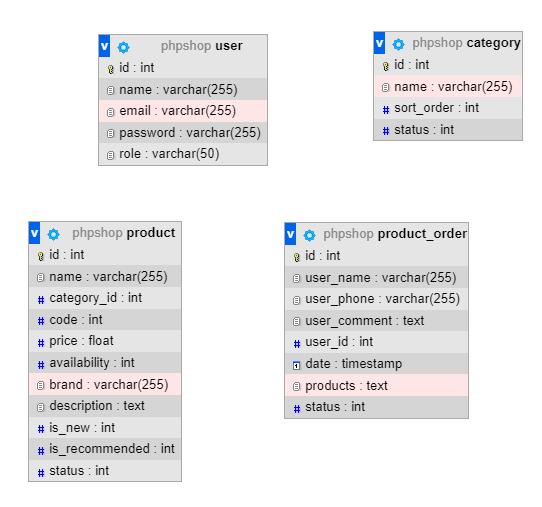
# Таблица 2.4 – Таблица «product\_order»

|  |  |  |
| --- | --- | --- |
| **Наименование атрибута** | **Тип атрибута** | **Назначение** |
| id | Int | Идентификатор заказа, первичный ключ |
| user\_name | Varchar(255) | Имя заказчика |
| user\_phone | Varchar(255) | Телефон заказчика |
| user\_comment | Text | Комментарий заказа |
| user\_id | Int | Идентификатор пользователя |
| date | Timestamp | Дата заказа |
| products | Text | Список товаров в заказе |
| status | Int | Статус заказа |

Составлено автором

В таблице 2.4 представлена данные о заказах пользователей.

Логическая схема базы данных представлена на рисунке 2.1.



# Рисунок 2.1 – Логическая схема базы данных

Таким образом проведено проектирование программного продукта.

# Реализация программного продукта

Процесс создания программного продукта делится на несколько стадий:

* изучение выбранной предметной области,
* обработка полученной информации,
* выбор средств реализации,
* реализация.

Для получения доступа к админ-панели сайта для начала требуется авторизоваться на сайте, после чего перейти по пути «’корневой каталог’/admin» в адресной строке. При этом происходит проверка согласно данным в БД.

# Листинг 2.1 – Проверка администратора

public static function checkAdmin()

{

// Проверяем авторизирован ли пользователь. Если нет, он будет переадресован

$userId = User::checkLogged();

// Получаем информацию о текущем пользователе

$user = User::getUserById($userId);

// Если роль текущего пользователя "admin", пускаем его в админпанель

if ($user['role'] == 'admin') {

return true;

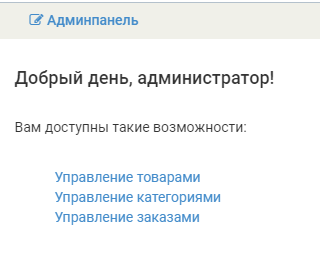
}

// Иначе завершаем работу с сообщением об закрытом доступе

die('Access denied');

}

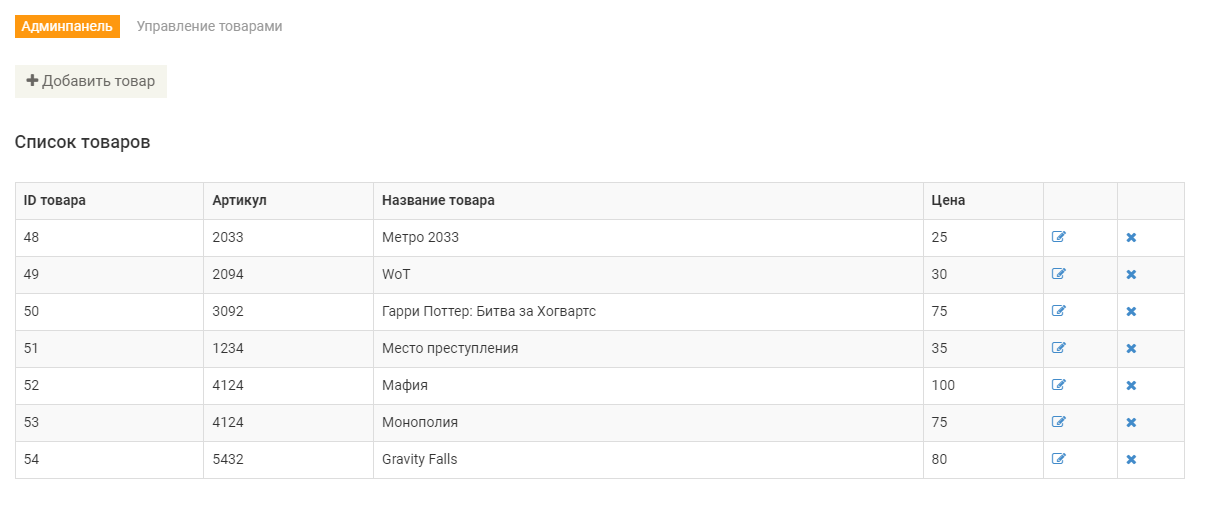
В случае прохождения проверки администратор попадает в основное меню, представленное на рисунке 2.2. В нем представлены ссылки для работы с различными таблицами базы данных.



# Рисунок 2.2 – Главная страница

Из главной страницы администратор может перейти к управлению списком товаров, категорий и заказов.

При переходе к определенному разделу администратор имеет возможность добавлять, просматривать и редактировать данные. Пример страницы для работы с конкретной таблицей представлен на рисунке 2.3.

****

**Рисунок 2.3 – Управление данными списка товаров**

Реализация метода добавления нового товара в базу данных представлена в листинге 2.2.

# Листинг 2.2 – Реализация добавления нового товара

public function actionCreate(){

self::checkAdmin(); // Проверка доступа

// Получаем список категорий для выпадающего списка

$categoriesList = Category::getCategoriesListAdmin();

if (isset($\_POST['submit'])) /\* Обработка формы \*/{

// Если форма отправлена // Получаем данные из формы

$options['name'] = $\_POST['name'];

$options['code'] = $\_POST['code'];

$options['price'] = $\_POST['price'];

$options['category\_id'] = $\_POST['category\_id'];

$options['brand'] = $\_POST['brand'];

$options['availability'] = $\_POST['availability'];

$options['description'] = $\_POST['description'];

$options['is\_new'] = $\_POST['is\_new'];

$options['is\_recommended'] = $\_POST['is\_recommended'];

$options['status'] = $\_POST['status'];

$errors = false; // Флаг ошибок в форме

// При необходимости можно валидировать значения нужным образом

if (!isset($options['name']) || empty($options['name']))

$errors[] = 'Заполните поля';}

if ($errors == false) { // Если ошибок нет добавляем товар

$id = Product::createProduct($options);

if ($id) { // Если запись добавлена

if // Проверим, загружалось ли через форму изображение

(is\_uploaded\_file($\_FILES["image"]["tmp\_name"])) {

move\_uploaded\_file($\_FILES["image"]["tmp\_name"], $\_SERVER['DOCUMENT\_ROOT'] . "/upload/images/products/{$id}.jpg");}};

header("Location: /admin/product");}}

require\_once(ROOT . '/views/admin\_product/create.php');

return true;}

Реализация метода удаления товара представлена в листинге 2.3.

# Листинг 2.3 – Реализация удаления товара

public function actionDelete($id){

self::checkAdmin(); // Проверка доступа

if (isset($\_POST['submit'])) { // Обработка формы

Product::deleteProductById($id);

// Перенаправляем пользователя на страницу управлениями товарами

header("Location: /admin/product");

}

// Подключаем вид

require\_once(ROOT . '/views/admin\_product/delete.php');

return true;}

Остальные методы добавления и удаления обработаны аналогичным образом.

Таким образом была реализована админ-панель интернет-магазина.

* 1. **Тестирование и отладка программного продукта**

**2.3.1 Основные концепции тестирования и отладки программного подхода**

Тестирование программного обеспечения – проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом.

Отладка программного обеспечения – процесс локализации и исправления ошибок, обнаруженных при тестировании программного обеспечения. Локализацией называют процесс определения оператора программы, выполнение которого вызвало нарушение нормального вычислительного процесса. Для исправления ошибки необходимо определить ее причину, то есть найти оператор или фрагмент, содержащие ошибку.

Тестирование и отладка программ различаются тем, что при тестировании устанавливается факт ошибки, а отладка выявляет причину и предлагает способы ее устранения.

**2.3.2 Методы испытаний**

Для тестирования программного продукта использовался метод «Тестовых случаев».

Тестовый случай (тест-кейс) – это артефакт, описывающий совокупность шагов, конкурентных условий и параметров, необходимых для проверки реализации тестируемой функции или ее части. Каждый тестовый случай составляется после разработки общего плана тестирования. Набор из нескольких тест-кейсов называется тестовым набором.

Тестирование проводится путем сравнения ожидаемого и фактического результата. Если ожидаемый и фактический результаты совпадают, то можно считать тест успешно пройденным. Если ожидаемый и фактический результаты отличаются, то тест считается пройденным неуспешно и говорит о необходимости доработки программного средства.

# 2.3.3 Тестирование сайта

План тестирования:

* тестирование запуска приложения,
* тестирование перехода на другие формы,
* тестирование добавления данных,
* тестирование удаления данных.

Тест-кейсы представлены в таблицах 2.5-2.8.

# Таблица 2.5 - Тест-кейс «Запуск сайта»

|  |  |  |
| --- | --- | --- |
| **TC ID/Priority** | **32о001** | **Крайне высокий** |
| **IDEA**: Запуск приложения и открытие главной страницы. | | |
| **Revision History** | | |
| **Created on**: / /2021 by Н. С. Прохоренко | | Новый тест-кейс |
| **Execution part** | | |
| **PROCEDURE** | | **EXPECTED RESULT** |
| 1) Запустить сайт и авторизоваться под администратора | | Появилась главная страница. Успешная авторизация |

Составлено автором

Фактический результат совпадает с ожидаемым. После запуска приложения открылась главная страница. Тест пройден успешно.

# Таблица 2.6 – Тест-кейс «Переход на другие страницы»

|  |  |  |
| --- | --- | --- |
| **TC ID/Priority** | **32о002** | **Средний** |
| **IDEA**: Проверка перехода на другие страницы. | | |
| **Revision History** | | |
| **Created on**: / /2021 by Н. С. Прохоренко | | Новый тест-кейс |
| **Execution part** | | |
| **PROCEDURE** | | **EXPECTED RESULT** |
| 1. Запустить сайт и авторизоваться, 2. Нажать на ссылку для перехода на другую форму. | | Выполнен переход на другую страницу. |

Составлено автором

Фактический результат совпадает с ожидаемым. Переход на другие страницы происходит без ошибок. Тест пройден успешно.

# Таблица 2.7 – Тест-кейс «Добавление нового товара»

|  |  |  |
| --- | --- | --- |
| **TC ID/Priority** | **32о003** | **Средний** |
| **IDEA**: Добавление нового товара. | | |
| **Revision History** | | |
| **Created on**: / /2021 by Н. С. Прохоренко | | Новый тест-кейс |
| **Execution part** | | |
| **PROCEDURE** | | **EXPECTED RESULT** |
| 1. Запустить сайт и авторизация, 2. Перейти к админ-панели, 3. Выбрать категорию «Управление товарами», 4. Нажать на кнопку «Добавить товар», 5. Заполнить поля формы соответствующими данными, 6. Нажать кнопку «Добавить». | | Новый товар добавлен в базу данных. |

Составлено автором

Фактический результат совпадает с ожидаемым. Новый товар успешно добавлен в систему и отображается в таблице. Тест пройден успешно.

**Таблица 2.8– Тест-кейс «Удаление товара»**

|  |  |  |
| --- | --- | --- |
| **TC ID/Priority** | **32о004** | **Средний** |
| **IDEA**: Удаление товара. | | |
| **Revision History** | | |
| **Created on**: / /2021 by Н. С. Прохоренко | | Новый тест-кейс |
| **Execution part** | | |
| **PROCEDURE** | | **EXPECTED RESULT** |
| 1. Запустить сайт и авторизация, 2. Перейти к админ-панели, 3. Выбрать категорию «Управление товарами», 4. Выбрать товар, 5. Нажать кнопку «Удалить товар» | | Товар удален. |

Составлено автором

В результате проведенных испытаний было установлено, что разработанный программный продукт отвечает предъявленным требованиям. При проведении тестирования на первом этапе были выявлены незначительные ошибки. После исправления программный продукт показал стабильность и надежность при исполнении. Проект готов к эксплуатации.