**Внутрикорпоративный магазин**

**«Syn-Shop»**

**РАБОЧАЯ ДОКУМЕНТАЦИЯ**

**Руководство программиста**

**Листов – 17**

Киров

2022

ОГЛАВЛЕНИЕ

[1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММЫ 4](#_Toc119533585)

[2. ХАРАКТЕРИСТИКИ И РАБОТА ПРОГРАММЫ 5](#_Toc119533586)

[ЛИСТИНГ ПРОГРАММЫ 16](#_Toc119533587)

**АННОТАЦИЯ**

Данная рабочая документация вмещает в себя руководство программиста для работы с внутрикорпоративным интернет-магазином «Syn-Shop».

Документация состоит из двух разделов:

* «Назначение и условия применения программы» - функционал, выполняемый данной программой, и условия для корректного функционирования кода;
* «Характеристика сайта» - Основные характеристики программы.

1. Назначение и условия применения программы

Данный продукт был разработан с целью повышения продуктивности рабочего персонала. Продуктивность будет повышаться из-за своеобразных дополнительных вознаграждений, которые работники будут приобретать за внутреннюю валюту, начисляемую дополнительно при продуктивной работе сотрудника.

Сайт является дополнительным инструментом для руководителей компаний, заинтересованных в повышении качества и продуктивности работы собственных сотрудников.

Видом продукта был выбран сайт, по причине доступности большинству возможных пользователей. Данным продуктом могут пользоваться сотрудники компании, избранные директором, с целью покупки товаров на внутреннюю валюту, начисленную сотруднику директором за выдающуюся работу.

1. Характеристики и работа программы

Выполненные требования за время разработки:

* Чтобы данный сайт выполнял все современные требования, он должен обладать определенными элементами:
* понятный интерфейс – интерфейс программы должен быть информативен и минималистичен;
* интеграция с базой данных – каждый продукт подобного типа обладает интеграцией с БД;
* авторизация пользователей – предоставление определённому лицу или группе лиц прав на выполнение определённых действий;
* веб-витрина – перечень товаров с возможностью приема заявок от клиентов;
* пользовательский профиль – совокупность данных и настроек окружения пользователя.

Настройка рабочего окружения:

1. Установка кроссплатформенной сборки веб-сервера «XAMPP» – <https://www.xampp.ru/>.
2. Запуск, от имени администратора, файла «xampp-control.exe».

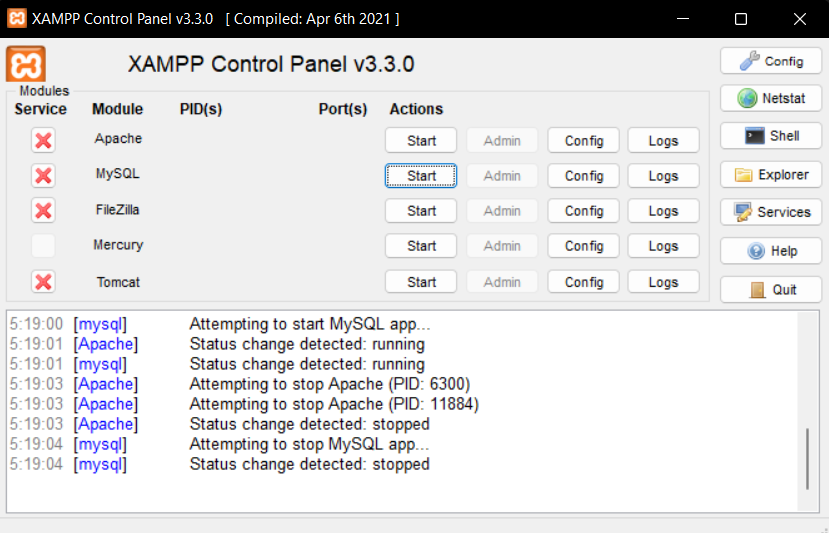


Рисунок 1 – Меню приложения «XAMPP»

1. Включите модули «Apache» и «MySQL», с помощью кнопок «Start».

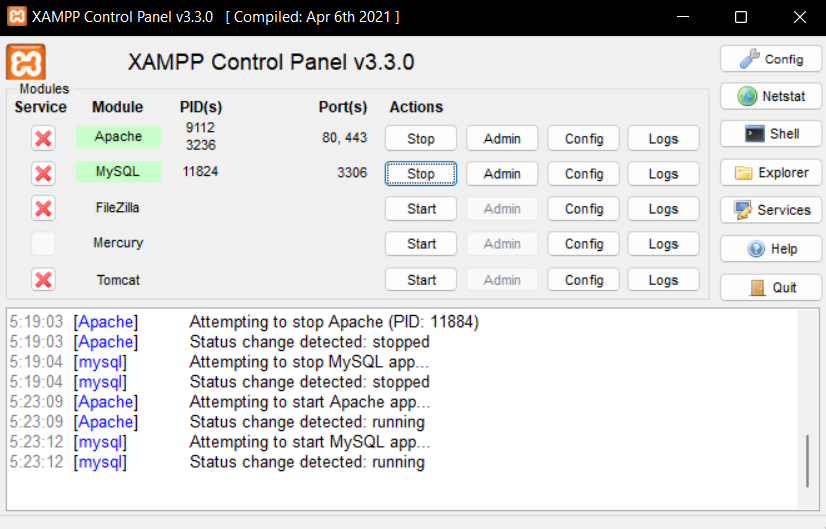


Рисунок 2 – «XAMPP» с включенными модулями

1. Откройте phpMyAdmin (по ссылке - <http://localhost/phpmyadmin/>) и создайте пустую базу данных с наименованием – «shop\_01».
2. Импортируйте в созданную БД файл – «shop\_01.sql», отключив все дополнительные параметры как указанно на рисунке.

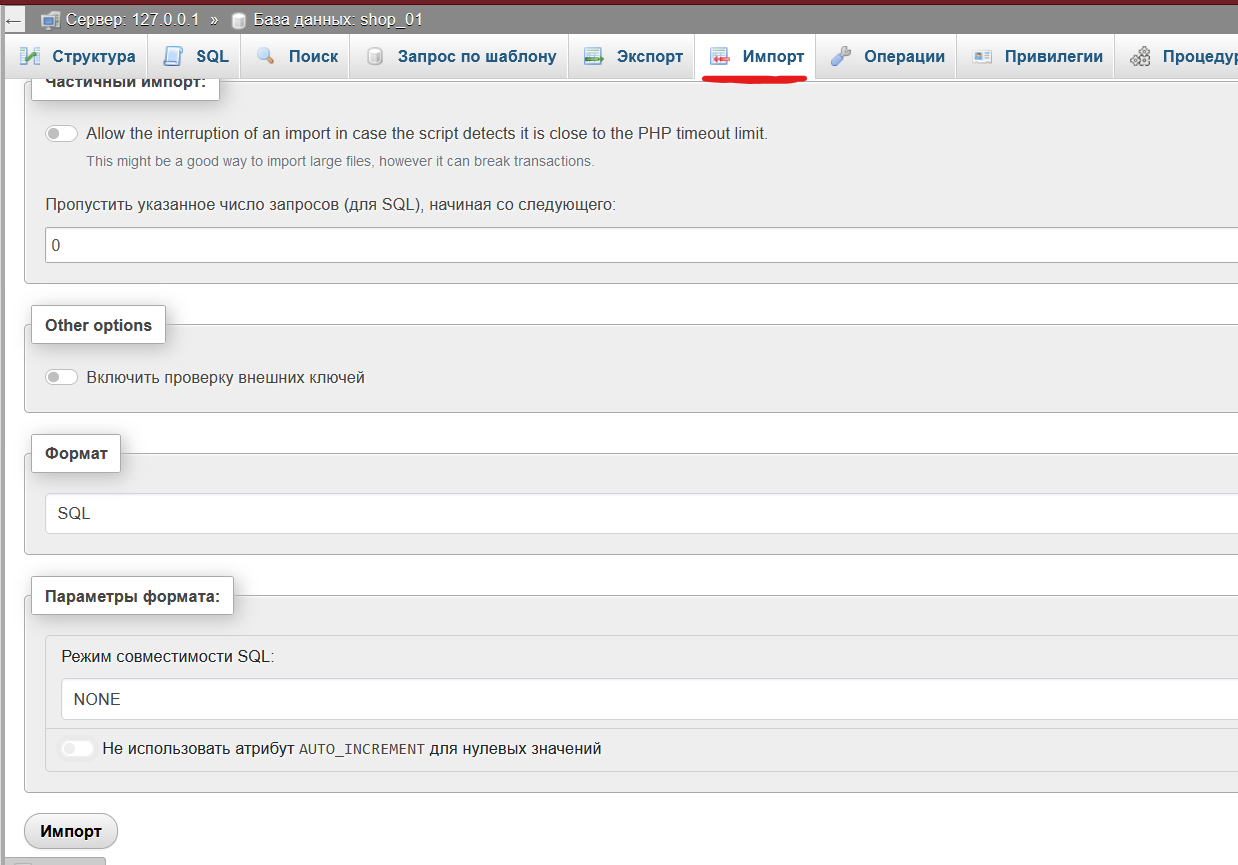


Рисунок 3 – Импорт БД в phpMyAdmin

1. Чтобы импортировать саму программу, нужно в проводнике папку – «shop\_fin\_2» перенести в папку – «xampp\htdocs».
2. После данных действий открыть конечный продукт можно по ссылке – <http://localhost/shop_fin_2/content/>.

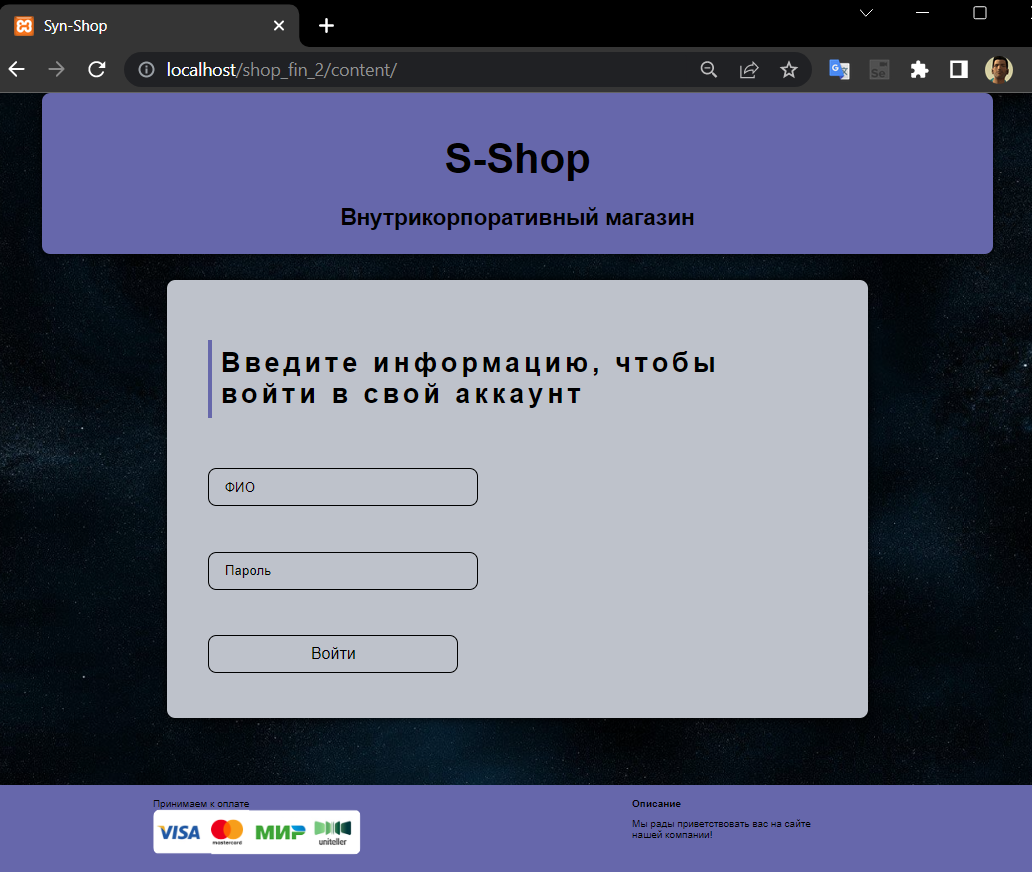


Рисунок 4 – Конечный результат

**Архитектура сайта состоит из следующих групп:**

1. Страницы сайта
2. Элементы на страницах
3. Стили
4. Картинки
5. Код части администратора
6. Код части клиента

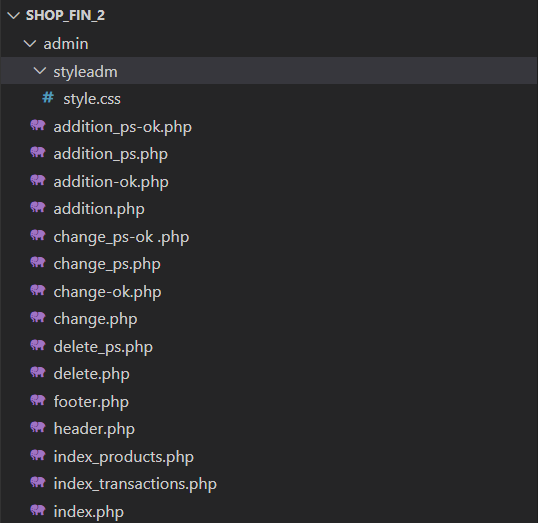


Рисунок 5 – Файловая архитектура админской части

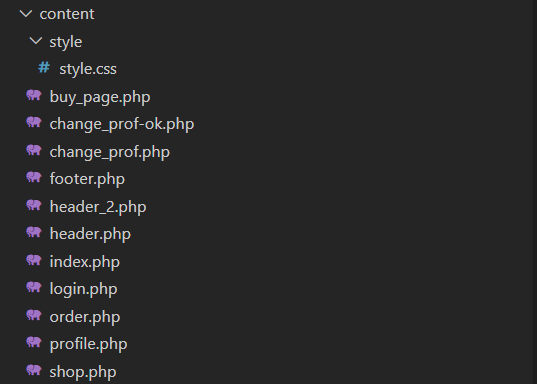


Рисунок 6 – Файловая архитектура пользовательской части

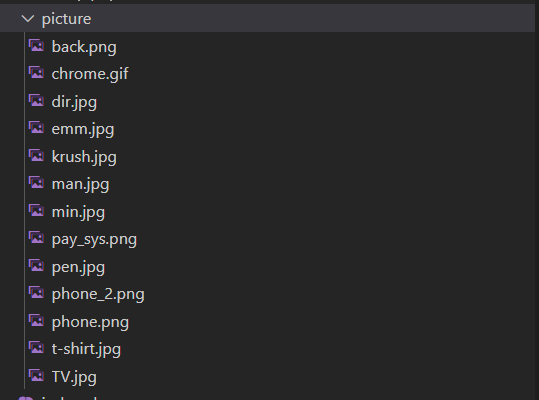


Рисунок 7 – Картинки

Для создания данного проекта использовались языки программирования: PHP – бэкенд, HTML – разметка и CSS – стили элементов.

С помощью языка разметки HTML было, но был создан набросок примерного расположения элементов на сайте. Язык описания внешнего вида CSS использовался для назначения визуальных характеристик каждому элементу сайта, все данные о визуальных характеристиках элементов хранятся в отдельном документе «style.css».

При создании страницы регистрации создаются три документа с расширением «php». Первый документ называется «header.php» его основным функционалом является верхняя часть сайта, на котором располагаются название и подзаголовок. Вторым элементом является нижняя часть сайта, на котором располагается дополнительная информация о сайте. Основная часть сайта и называется она «index.php» на нём располагаются вызовы документов «header.php», также последней строкой кода вызов документа «footer.php», в основной части кода прописаны: небольшой текст для пользователя, поля для ввода ФИО и пароля пользователя и третьим элементом является кнопка «войти» которая после нажатия, при совпадении введенных данных с данными в БД, будет пересылать пользователя на другую страницу.

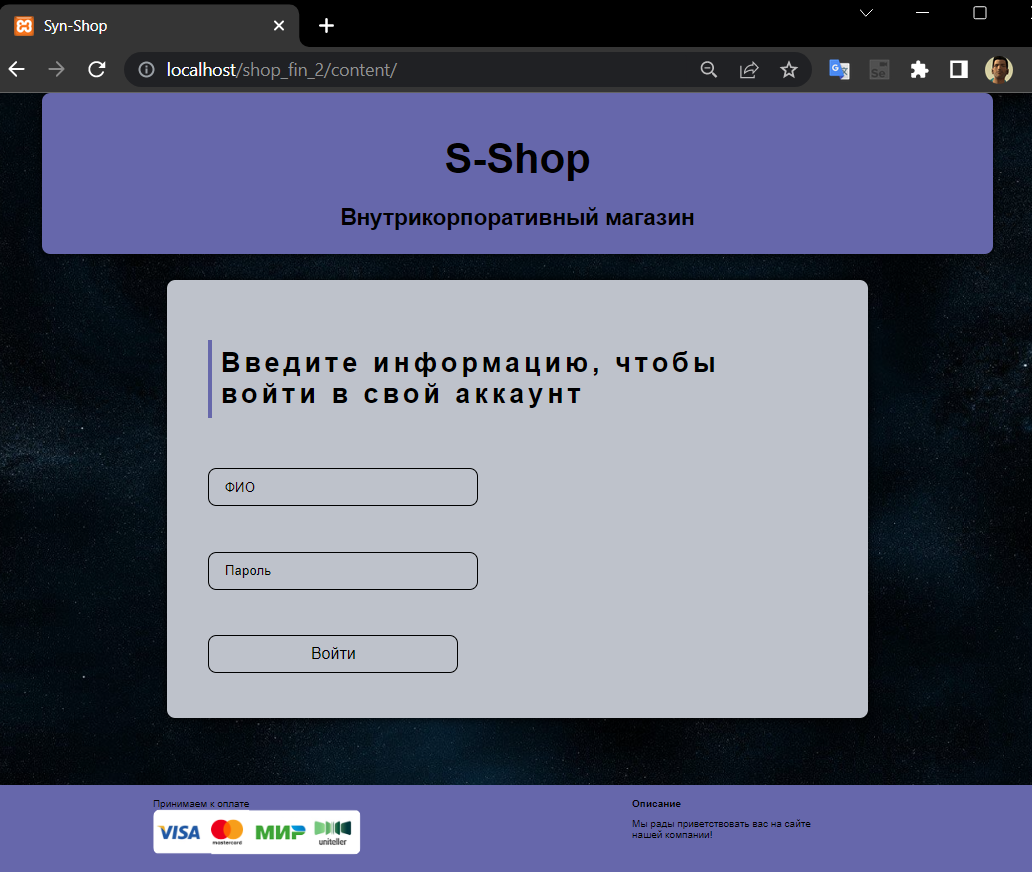


Рисунок 8 – Страница авторизации

Для реализации программной части кода был создан отдельный документ под названием «login.php», который вел сравнение введённых пользователем данных с созданной заранее базой данных. Данная операция реализовалась путём использования двух SQL функции: SELECT - которая обращалась к определённой таблице в базе данных и LIKE - которая проводила сравнение двух значений. Также было поставлено условие которые при несовпадениях какого-либо из введённых пользователем значений отправляло на окно авторизации, в случае совпадения двух полей с имеющимися в базе данных пользователя переносит на витрину сайта, функционал которой реализовывается в документе «shop.php».

Страница витрины отображает продукты возможные для покупки пользователем взаимодействуя с таблицей предметов в БД. Нововведением также является новый хедер для «shop.php» и последующих создаваемых страниц именующийся как «header\_2.php». На новом хедере присутствует отображение счета пользователя, данный счет индивидуален для каждого пользователя, а подсоединение к нему создается с использованием введенного ранее пароля. Вторым новым элементом являются кнопки-ссылки на витрину и профиль пользователя. На странице витрины пользователь может нажать кнопку «Купить» у любого товара (доступного по цене), после чего его перенесет на страницу с благодарностью за покупку, функционал которой прописан в документе «buy\_page.php», кроме визуальной данный документ реализует списание баллов со счета пользователя в зависимости от стоимости товара (эта процедура реализуется с использованием SQL функции - UPDATE). Изменения счета пользователь может заметить при переходе на страницу профиля или витрины, используя хедер.

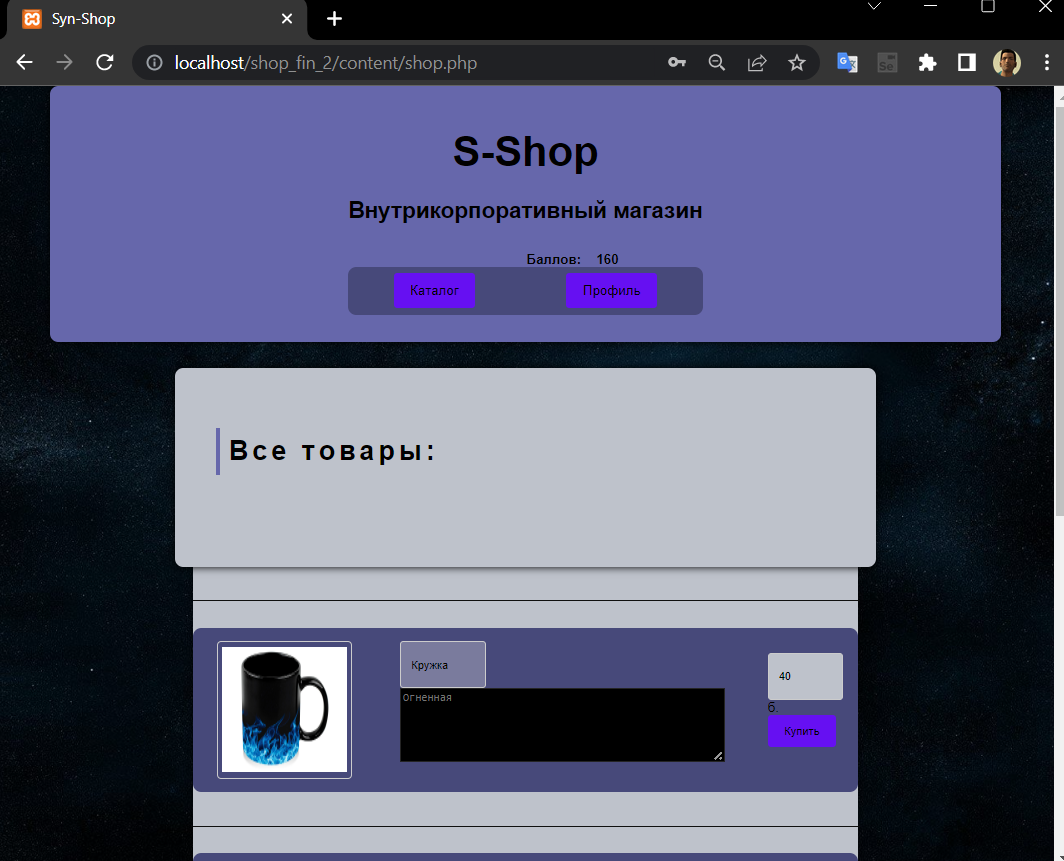


Рисунок 9 – Страница витрины

Взаимодействие с кнопкой «Профиль», находящейся в хедере переносит на страницу с дополнительной информацией о пользователе, которая берется из таблицы с пользователями в БД по принципу баллов в хедере. Дополнительным элементом на этой странице является кнопка «Изменить», нажав на которую можно изменить информацию о пользователе в новом окне с пустыми строками. После нажатия кнопки сохранить в целях безопасности пользователя переносит обратно на страницу авторизации.

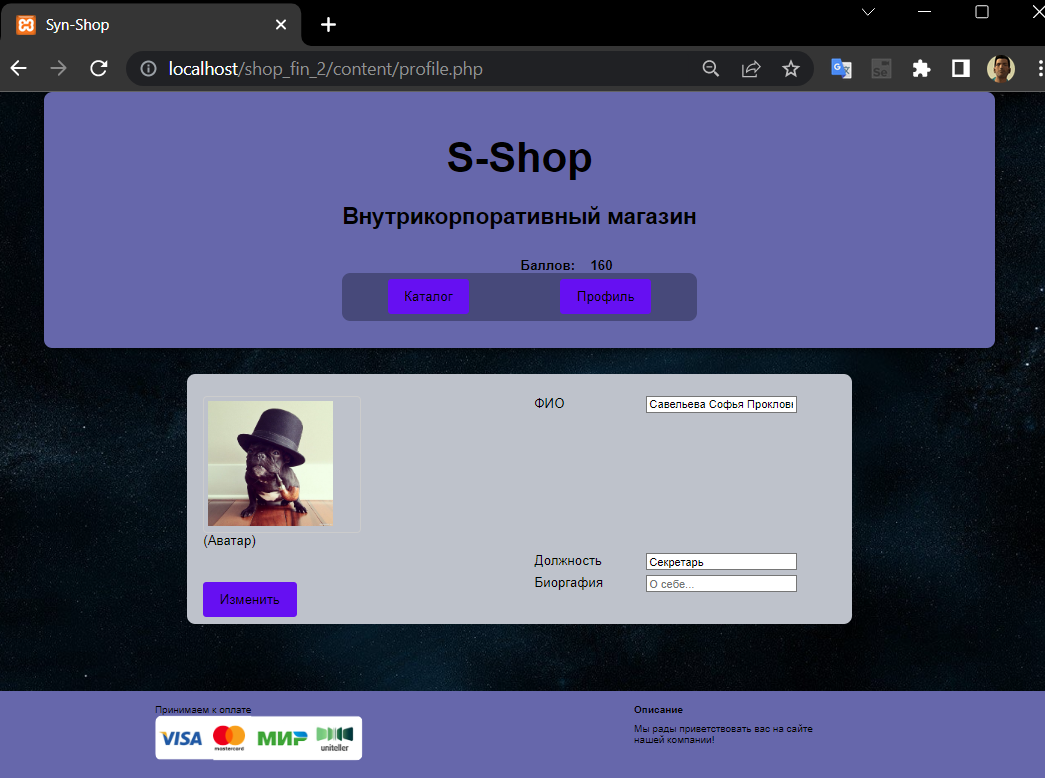


Рисунок 10 – Страница профиля

**Связь с БД**

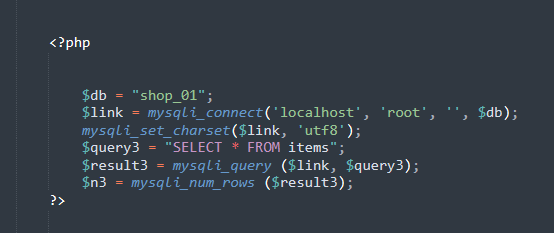


Рисунок 11 – Подвязывание кода к БД

Взаимодействие с базой данных с помощью администратора СУБД «phpMyAdmin»

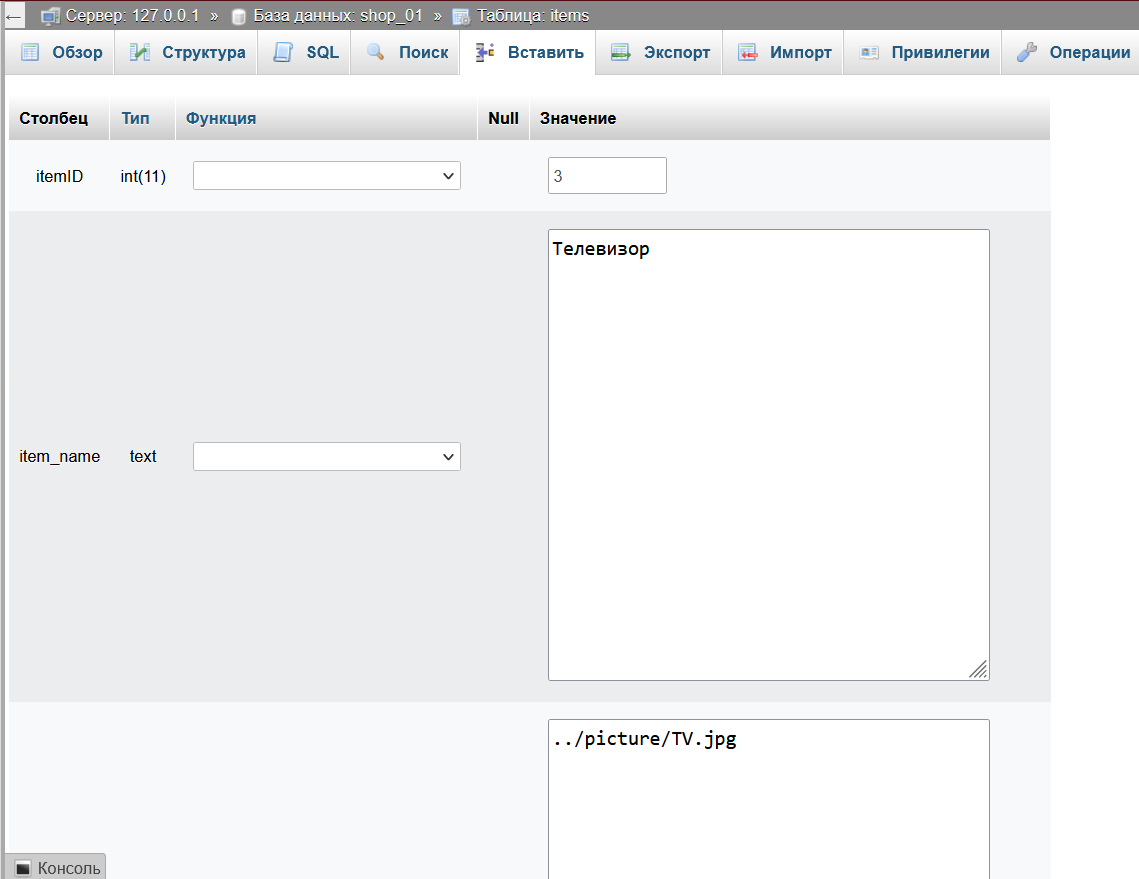


Рисунок 13 – Изменение данных о продукте в «phpMyAdmin»

ЛИСТИНГ ПРОГРАММЫ

Код страницы авторизации:

<?php include 'header.php';

?>

<div class="form-box">

<div class="center">

<h1>Введите информацию, чтобы войти в свой аккаунт</h1>

<form action="login.php" method="post">

<div class="inputbox">

<input type="text" name="fullname" required="required">

<span>ФИО</span>

</div>

<div class="inputbox">

<input type="text" name="org\_pas" required="required">

<span>Пароль</span>

</div>

<div class="inputbox">

<input type="submit" class="knopka" value="Войти">

</div>

</form>

</div>

</div>

</main>

</body>

<?php include 'footer.php' ?>

<?php

session\_start();

if (!isset($\_POST['fullname']) || !isset($\_POST['org\_pas'])){

die ("Введены не все данные.");

}

$fullname = $\_POST['fullname'];

$org\_pas = $\_POST['org\_pas'];

$db = "shop\_01";

$connect = mysqli\_connect('localhost', 'root', '', $db);

if (!$connect) die ("Невозможно подключение к MySQL");

mysqli\_select\_db ($connect, $db) or die ("Невозможно открыть базу данных");

$check\_user = mysqli\_query($connect, "SELECT userID FROM user WHERE full\_name LIKE '%$fullname%' and org\_pas='$org\_pas'");

#var\_dump($check\_user);

mysqli\_close ($connect);

if (mysqli\_num\_rows($check\_user) > 0){

$\_SESSION['org\_pas']=$org\_pas;

header('Location: shop.php');

}else{

header('Location: index.php');

}

?>