**Сайт**

**«PersonalSite»**

**РАБОЧАЯ ДОКУМЕНТАЦИЯ**

**Руководство программиста**

**Листов - 13**

Киров

2022

**ОГЛАВЛЕНИЕ**

[**1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГАММЫ 4**](#_Toc120097924)

[**2. ХАРАКТЕРИСТИКА И РАБОТА ПРОГРАММЫ 5**](#_Toc120097925)

[**ЛИСТИНГ ПРОГРАММЫ 10**](#_Toc120097926)

**АННОТАЦИЯ**

В данном программном документе приведено руководство программиста по работе с сайтом «elf».

В разделе «Назначение и условия применения программы» указаны назначение и функции, выполняемые сайтом, условия, необходимые для его работы.

В разделе «Характеристика сайта» приведено описание основных

характеристик и особенностей.

1. Назначение и условия применения программы

Главной задачей, которую необходимо было выполнить, было создание персонального сайта. Актуальность создания сайта состоит в том, что если необходимо донести информацию максимально быстро до огромного количества людей, то лучше, чем с помощью собственного сайта сделать это не получится. Веб-ресурс позволяет представить информацию о компании или человеке и их товарах или услугах сжато и одновременно полноценно.

Что значит персональный сайт?

Персональный сайт – это веб-ресурс, с помощью которого человек может рассказать о себе, представить пользователям свои услуги.

Главной задачей, которую необходимо было выполнить, было создание сайта «PersonalSite».

Создание сайта позволило бы пользователям просматривать информацию о том или ином человеке или компании, чтобы оценить его товары или предоставляемые услуги. Вследствие чего это привело бы к повышению дохода владельца сайта или популярность самого владельца или его продукта.

В ходе производственной практики был создан сайт «PersonalSite», предназначенный просмотра информации о человеке или компании, предоставляемый те или иные товары и услуги.

1. Характеристики и работа программы

Требования к логике сайта, которые были выполнены в ходе разработки:

1. Сайт находится в открытом доступе для любого человека, имеющий свободный доступ в интернет.
2. Плавное появление элементов страницы и разворачивание «Аккордеона», где размещена основная информация о владельце сайта.
3. Отправка сообщений на почту владельца сайта.
4. Проверка отправляемых данных (Поле ввода не должно быть пустым и иметь минимум 5 символов)
5. После отправки формы, должна быть реализована возможность вернуться на начальную страницу.

Для того, чтобы собрать проект требуется:

1. Скачать Visual Studio Code и установить, по следующей ссылке:

<https://code.visualstudio.com/docs/?dv=win>

1. Зайти в браузер и по ссылке: <https://github.com/RIXRIP/YP> нажмите кнопку «Code» и нажмите кнопку «Download ZIP» и выберите директорию для установки репозитория себе на устройство.
2. Установить PHP по ссылке: php.net/downloads.php
3. Открыть Visual Studio Code и открыть папку с решением
4. Установить расширения для VS Code:

 -HTML CSS Support

-PHP

-PHP Server

-code runner

7. Проект готов к сборке

На рисунке 1 представлена файловая архитектура сайта.

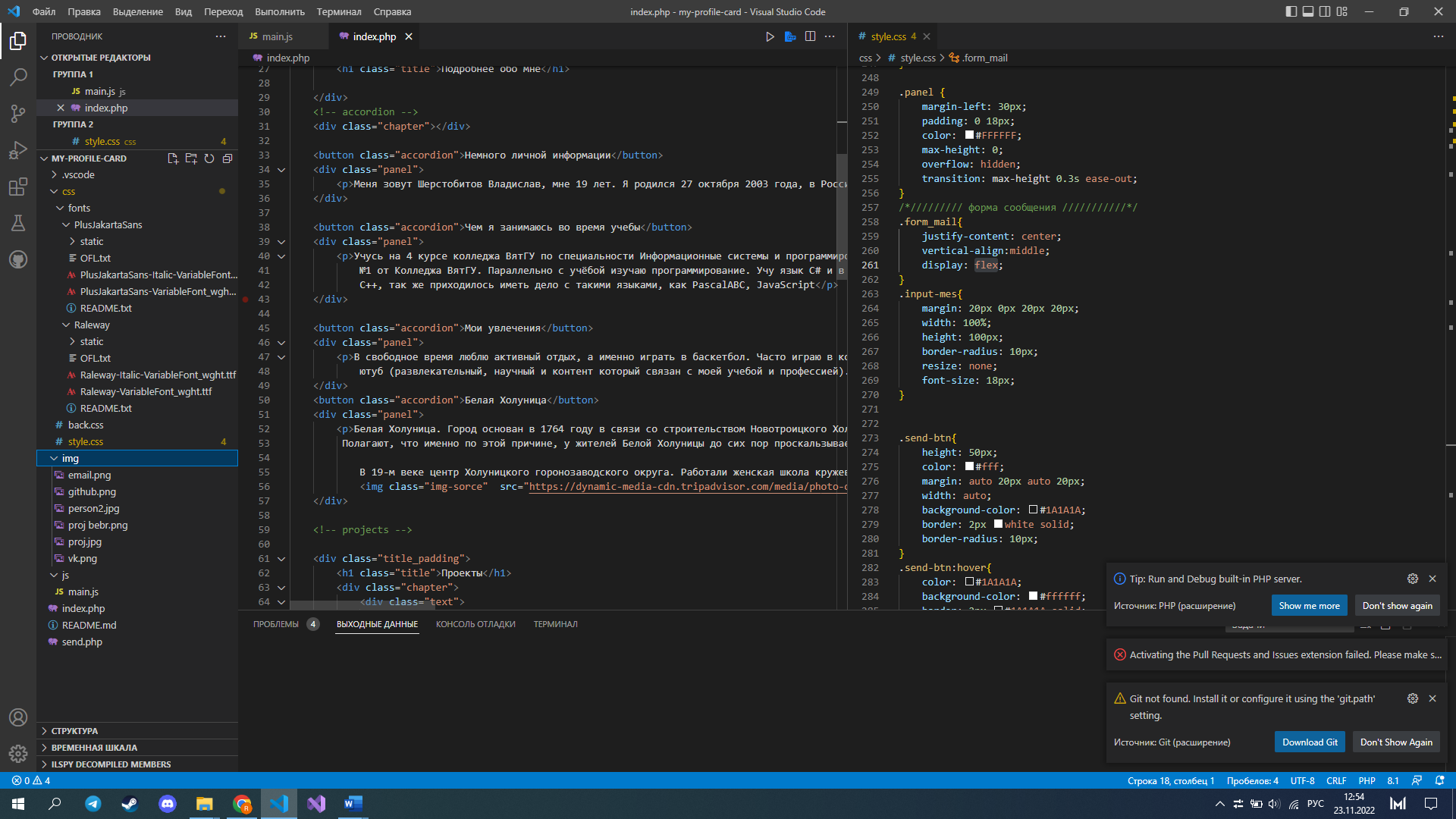


Рисунок 1 – Файловая архитектура сайта

Для работы сайта необходимы все файлы, представленные на рисунке 1 их можно разделить на следующие группы:

1. Файлы сценариев.
2. Файлы ресурсов.
3. Файлы интерфейсов.

При создании сайта использовались языки JS для фронт разработки, PHP для бэка, HTML для разметки страницы и CSS для написания стилей элементов.

Сайт состоит из одностраничного лендинга файла index.php, данный файлы используют расширение .php так как он применяется для динамического отображения информации на сайте, send.php файлы являются исполняемым кодом отправки сообщений на почту, необходимым для функционирования продукта. Все стили для элементов сайта находятся в файле style.css.

На рисунке 2 изображена начальная страница персонального сайта.

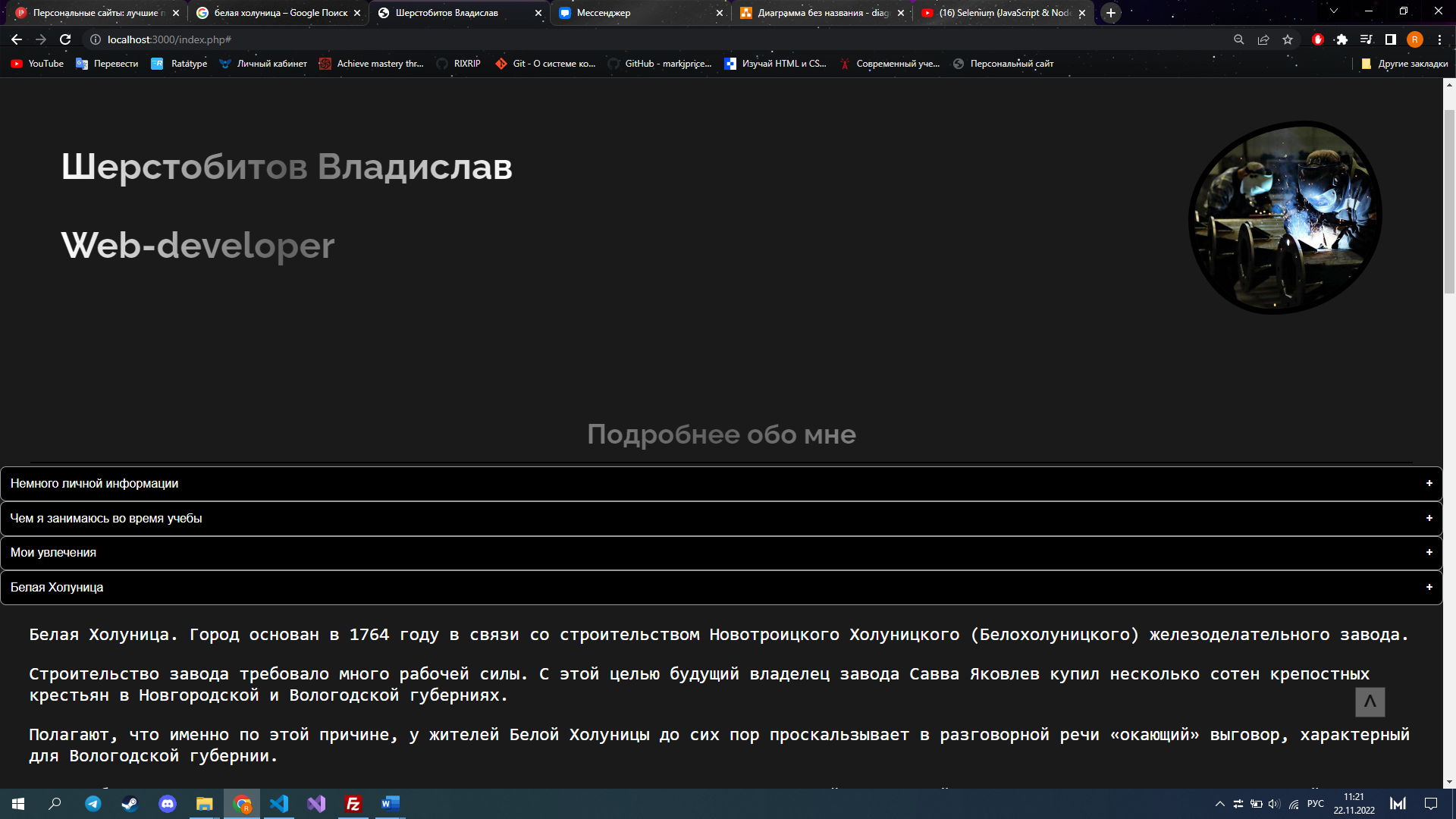


Рисунок 2 – Начальная страница

На рисунке 3 предстален код плавного перехода от прозрачного состояния текста к полностью непрозрачному с коментариями.

Изображение выглядит как текст, снимок экрана, монитор, черный

Автоматически созданное описание

Рисунок 3 – Переход от прозрачного состояния текста к полностью непрозрачному.

На рисунке 4 показано, что при нажатии на элемент «аккордеона», он разворачивается и отображает информацию размещенную в нем.

На рисунке 5 изображена форма отправки сообщения на почту владельца сайта.

Изображение выглядит как текст, снимок экрана, монитор

Автоматически созданное описание

Рисунок 5 – форма отправки сообщения

На рисунке 6 изображена страница с сообщением что письмо успешно доставлено и кнопкой «Вернуться», чтобы вернуться на главную страницу.

Изображение выглядит как текст, снимок экрана, электроника

Автоматически созданное описание

Рисунок 6 – форма возврата на главную страницу.

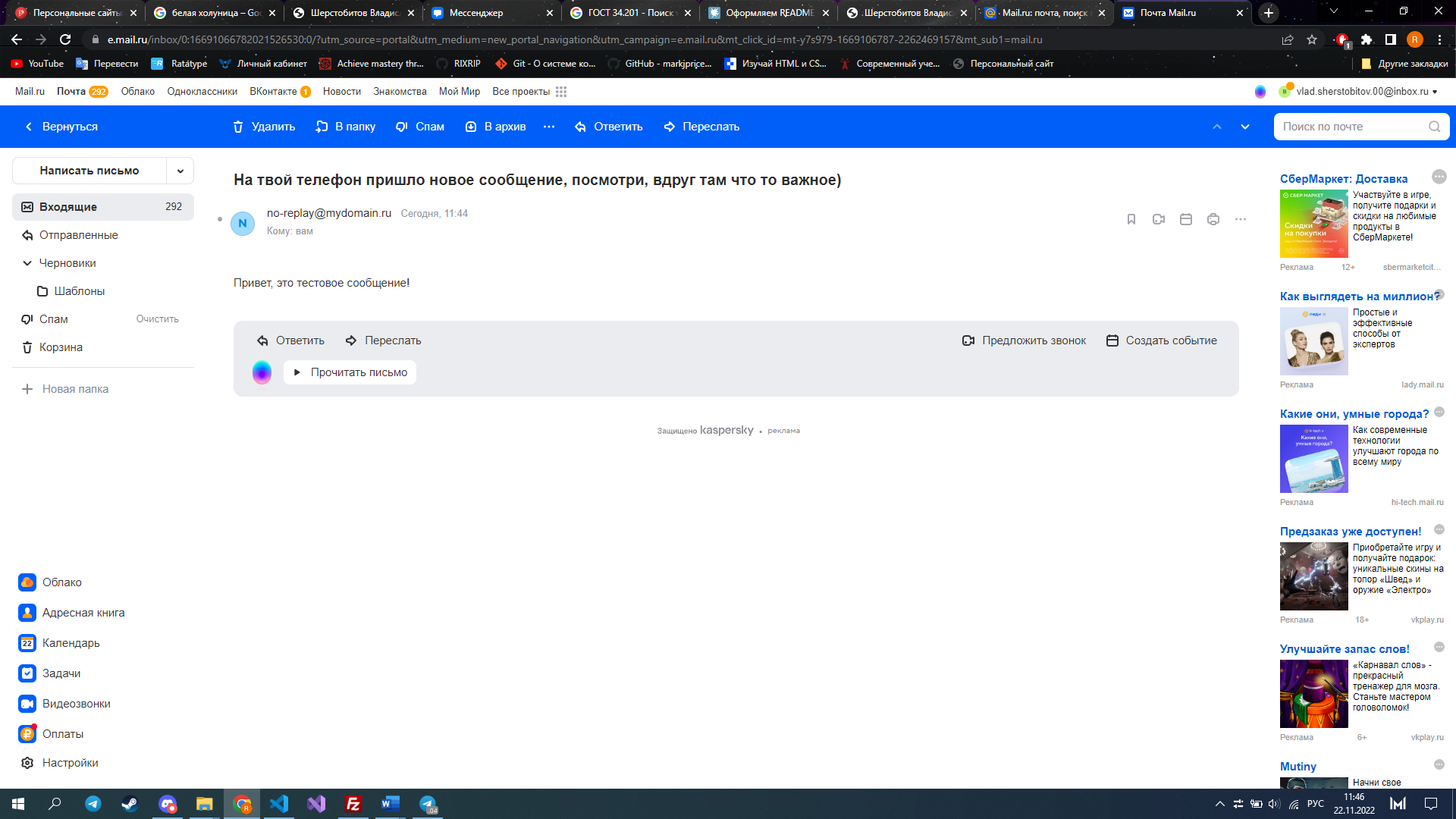
На рисунке 7 изображено письмо отправленое пользователем.

Рисунок 7 – отправленое сообщение дошло до получателя

На рисунке 8 изображена логика выполнения php скрипта отправки сообщений на почту.

Изображение выглядит как текст, снимок экрана, монитор, экран

Автоматически созданное описание

Рисунок 8 – Логика отправки сообщений на почту

На рисунке 9 изображен код анимации фотографии владельца сайта.

Так же на рисунке 9 изображен код подключения сторонних шрифтов для корректного их отображения на сайте.

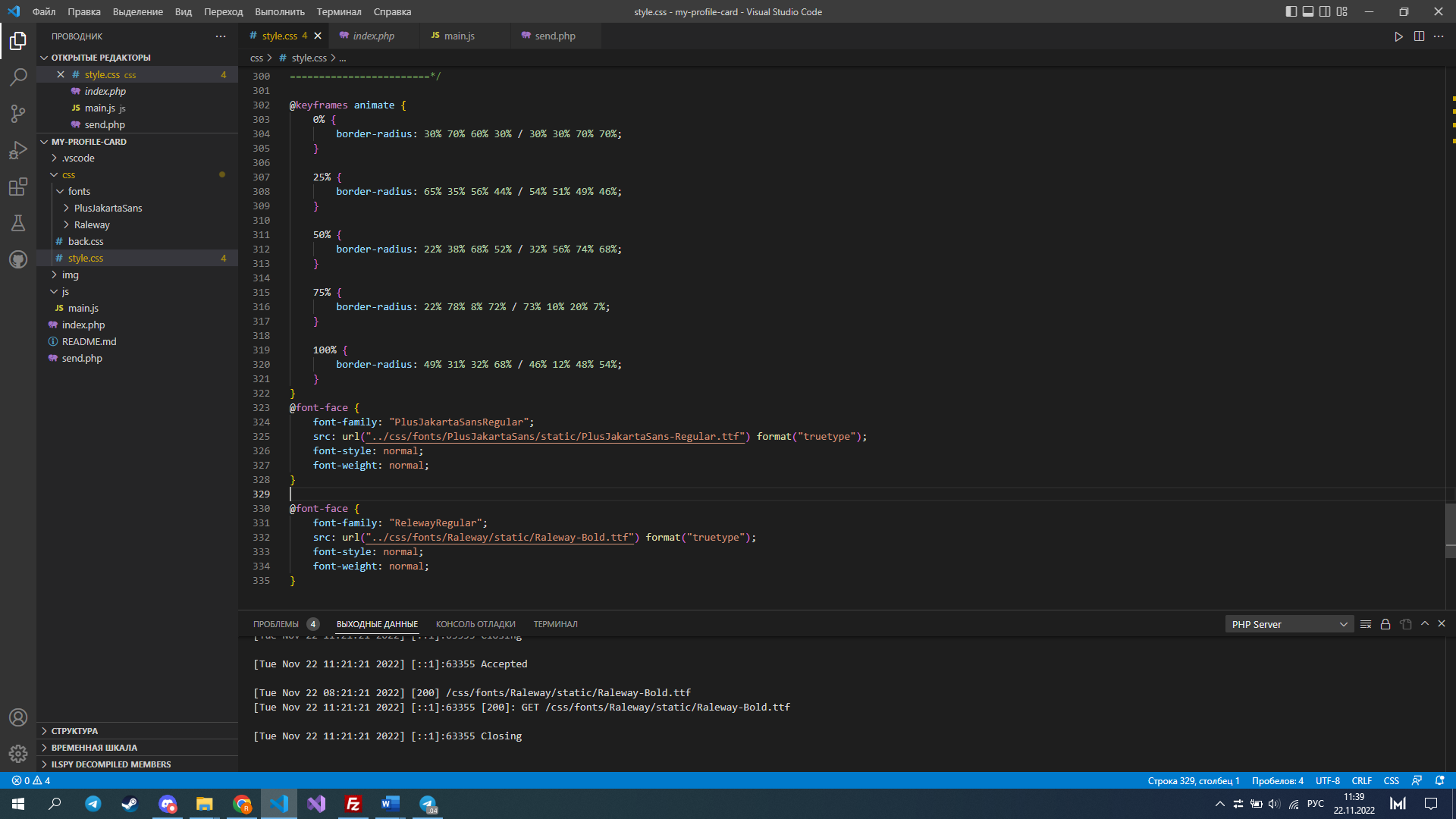


Рисунок 9 – код анимации фотографии и встраивание сторонних шрифтов

# **ЛИСТИНГ ПРОГРАММЫ**

**«Отправка сообщений»**

<?php

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

$login=htmlspecialchars($login); //перобразование символов в сущности

$login=urldecode($login);//декодирование URL

$login=trim($login);//удаление пробельных символов с обеих сторон сообщения

//отправка данных на почту

if(mail("vlad.sherstobitov.00@inbox.ru",

"На твой телефон пришло новое сообщение, посмотри, вдруг там что то важное)",

"".$login."\n",

"From: no-replay@mydomain.ru")

) {

    echo('Письмо успешно отправлено!');

}

else{

    echo('Письмо не отправлено, проверьте ошибки!');

}

?>

**«Аккордеон»**

let acc = document.getElementsByClassName("accordion"); //accordion функция для переключения между скрытием и отображением содержимого некоторого контента страницы

for (let i = 0; i < acc.length; i++) {//В цикле for, мы перебираем весь массив элементов с классом accordion

  acc[i].addEventListener("click", function () {//На каждый из элементов мы добавляем действие по клику, будет выполняться функция.

    let panel = this.nextElementSibling; //Назначаем переменной с названием panel – свойство nextElementSibling, это переключатель между скрытием и показом активной панели.

    if (panel.style.maxHeight)

      panel.style.maxHeight = null;

    else

      panel.style.maxHeight = panel.scrollHeight + "px";//Для раскрытия, спрятанных блоков с текстом, надо менять максимальную высоту, в зависимости от высоты панели на разных размерах экрана. Проверяем, если значение максимальной высоты в стилях для блока с классом panel, равно нулю, то новая максимальная высота станет больше на энное количество пикселей, в зависимости от размера экрана пользователя.

  });

}

**«Прозрачность текста»**

let val = 0;//элемент val предназначет для будущего определения значения свойства opacity

let opac = document.getElementById("opacAnimation");// переменная opac записывает в себя свойства и значения элемента opacAnimation

let interval = 0; //переменная interval предназначена для записи функции setInterval которая определяет интервал анимации функции opacAnimation

const opacAnimation = () => {// функция opacAnimation предназначена для изменения значения свойства opacity

  val++;

  let opacIncrement = String(val) + "%";

  opac.style.opacity = opac.style.opacity = opacIncrement;

  if (val == 100) (

    clearInterval(interval)

  )

}

interval = setInterval(opacAnimation, 15);