Министерство высшего образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

Владимирский государственный университет

им. А.Г. и Н.Г. Столетовых

Кафедра информационных систем и программной инженерии

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовому проекту по дисциплине   
" Технологии разработки веб-приложений "**

**на тему:**

Клиент-серверное приложение для организации работы цветочного магазина.

Проектирование и тестирование системы

Выполнила: студентка гр. ИСТм-120

Зеленцова А.А.

Принял: доц. кафедры ИСПИ

Салех Х.М.

Владимир, 2021

**Аннотация**

В рамках курсового проекта разработано веб–приложение «Цветочный портал». Программная система представляет собой веб–приложение с организацией методов и графических интерфейсов для взаимодействия пользователей с системой. В основе лежит клиент-серверная архитектура. Для хранения информации была разработана база данных. Проведено тестирование приложения.

Пояснительная записка содержит:

* 17 страниц;
* 9 рисунков.

**AN ANNOTATION**

As part of the course project developed a web application "Flower shop". A software system is a web application with an organization of methods and graphical interfaces for user interaction with the system. It is based on a client-server architecture. A database was developed to store the information. The application was tested.

<https://github.com/Arina7628/FlowerShop>

The explanatory memorandum contains:

* 17 pages;
* 9 figures.

Содержание

[ВВЕДЕНИЕ 5](#_Toc71984807)

[2. Постановка задачи 6](#_Toc71984808)

[3. Описание предметной области 7](#_Toc71984809)

[3.1. Общее описание 7](#_Toc71984810)

[3.2. Словарь предметной области 7](#_Toc71984811)

[3.3. Функциональные требования 8](#_Toc71984812)

[3.4. Нефункциональные требования 9](#_Toc71984813)

[3.5. Предполагаемая аудитория 10](#_Toc71984814)

[4. проектирование 11](#_Toc71984815)

[заключение 12](#_Toc71984817)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 13](#_Toc71984818)

[Приложение 14](#_Toc71984819)

ВВЕДЕНИЕ

В современном мире очень важно быть мобильным, поэтому нужно стараться привести решение простых задач к минимальной потере времени. И в этом помогают веб приложения.

Целью данного курсового проекта является создание приложения «Цветочный портал».

Разрабатываемое приложение поможет клиентам выбрать букеты в каталоге и заказать их, не посещая магазин, что очень актуально в эпидемию короновируса.

Данное приложение предназначено для клиентов цветочного магазина, а также для продавцов.

1. Постановка задачи

Целями создания веб–приложения «Цветочный портал» является возможность обеспечения информационной поддержки клиентов магазина, также возможность осуществления заказа.

Данное мобильное приложение поможет клиентам выбрать букеты в каталоге и заказать их, не посещая магазин.

Люди, покупающие цветы, не имеют возможности сравнивать цены на букеты в ближайших магазинах района, когда они уже находятся в магазине, так как эти данные не агрегированы и не доступны. Чтобы пользователи могли выбрать лучший вариант и купить лучшие букеты по самой низкой цене, скорректировать свое расписание, мы даем им возможность сделать заказ внутри приложения. Это должно увеличить выручку цветочных магазинов и помочь людям сделать выбор.

Для достижения поставленной цели необходимо реализовать следующие этапы:

1. Оформление шапки портала
2. Оформление главной страницы
3. Создание форм авторизации/регистрации
4. Оформление корзины
5. Создание чата
6. Описание предметной области
   1. Общее описание

Продукт представляет собой веб-приложение. При запуске открывается главная страница с каталогом товаров. В верхней части страницы находится меню. Нажав на конкретный товар, можно посмотреть детальную информацию о товаре: фотографии, цены, описание. Также на странице детальной информации о товаре присутствует кнопка «Добавить в корзину».

Пользователи приложения могут не только просматривать информацию о товарах, но и добавлять их в корзину, а также оформлять заказы и общаться с продавцом в чате.

Данный сервис должен хранить данные товаров. Так же данный сервис должен хранить данные пользователей, такие как пароли, электронные почты, дополнительная информация.

Приложение содержит реляционную базу данных, содержащую данные о товарах и заказах. Так же данное приложение должно хранить данные пользователей, такие как пароли, электронные почты, дополнительная информация.

* 1. Словарь предметной области

Пользователь – человек, который посещает сайт магазина. Неавторизованный посетитель сайта.

Клиент (покупатель) – авторизованный пользователь, который просматривает каталог товаров, добавляет товары в корзину, оформляет заказы.

Продавец (менеджер) – авторизованный пользователь, который продает товары, помогает клиенту оформить заказ с помощью чата.

Товар – цветы, которые продает магазин.

Чат – общение продавца и покупателя по вопросам оформления заказа и детализации товаров.

Оформление заказа – покупка выбранных товаров.

Корзина – список сохраненных пользователем выбранных товаров для заказа.

Браузер — программа-клиент (Internet Explorer, FireFox, Opera, Safari, Chrome и т.п.), предоставляющая пользователю возможности навигации по сайтам, просмотру сайтов и скачивания файлов.

Сайт — набор веб-страниц, объединенных общей тематикой, графическим оформлением, навигацией и ссылками. Сайт является информационной единицей, доступной для просмотра пользователям через интернет.

Веб-страница (страница) — HTML-документ сайта, отображаемый браузером пользователя и имеющий уникальный URL-адрес.

Содержимое сайта/приложения (контент) — текстовая, графическая или табличная информация, размещенная на сайте, без учета оформления страниц.

Административная часть (панель) сайта — не видимая для посетителей часть сайта, позволяющая редактировать контент сайта, а также добавлять или удалять разделы.

Главная страница — первая страница, которую видит пользователь при входе в приложение.

Внутренняя страница — любая другая страница, кроме главной страницы.

* 1. Функциональные требования

Реализуемые функции:

1. просмотр каталога товаров;
2. просмотр подробной информации о товаре;
3. регистрация;
4. авторизация.
5. добавление товара в корзину;
6. просмотр корзины;
7. оформление заказа;
8. чат;
9. просмотр всех заказов;
10. добавление, редактирование, удаление товаров;
11. просмотр списка пользователей и их ролей;
12. управление ролями пользователей.
    1. Нефункциональные требования
13. Требования к интерфейсу:
14. Контент, представленный на каждой странице, должен быть содержательным и лаконичным.
15. Дизайн сайта должен быть простым, ненавязчивым, но красивым.
16. Грамотный пользовательский интерфейс. Максимально удобный, ориентированный на пользователя.
17. Логичная навигация.
18. Адекватность цветовой гаммы.
19. Все экранные формы должны быть выполнены в едином стиле.
20. Пользовательский интерфейс данного приложения должен быть удобным и понятным для конечного пользователя, а также предоставлять быстрый доступ ко всем основным функциональным возможностям.
21. Интерфейсы связи и коммуникации: пользователь не может добавлять в корзину товары, общаться в чате с продавцом(менеджером) и оформлять заказ, для этого необходимо авторизоваться.
22. Требования к производительности.

Время реакции: система должна обрабатывать запрос пользователя в течение 60 секунд.

Для устойчивой работы приложения пользователю необходимо предоставить доступ к Интернету.

Серверная часть написана на языке программирования C#.

Серверная часть – ASP.NET Core (Visual Studio)

Клиентская часть – React.js

База данных: MS SQL SERVER.

1. Требования к сохранности данных.

База данных: MS SQL SERVER.

Система должна обеспечивать постоянное хранение важных данных. Также необходимо обеспечить валидацию данных, не допуская ввод некорректных значений.

1. Требования к защите информации.

Необходимо ограничить доступ к информации с помощью идентификации, аутентификации и авторизации.

1. Рамки, ограничения, правила и стандарты.
2. Для устойчивой работы приложения пользователю необходимо предоставить доступ к Интернету.
3. Для просмотра страниц с каталогом товаров и информацией о магазине не требуется авторизация/регистрация.
   1. Предполагаемая аудитория

Данное приложение предназначено для клиентов цветочного магазина, а также для продавцов.

1. проектирование и тестирование
   1. Первой задачей была разработка шапки приложения. Шапка страницы выполнена с помощью библиотеки material ui, куда входят шрифты Roboto и иконок svg. Так же использовался framework react bootstrap, который позволяет использовать react bootstrap компоненты в самом реакте.

Переход по страницам осуществляется на основе маршрутизации react роутера. Она позволяет сопоставить запросы приложения к запросам с определёнными компонентами. Для незарегистрированного пользователя доступны все вкладки, кроме корзины.

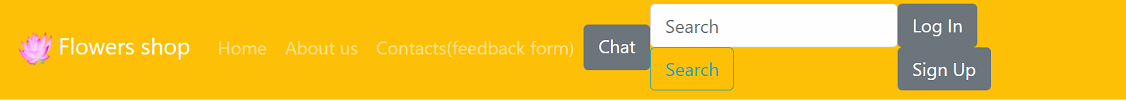


Рисунок 1 – Вид для неавторизованного пользователя

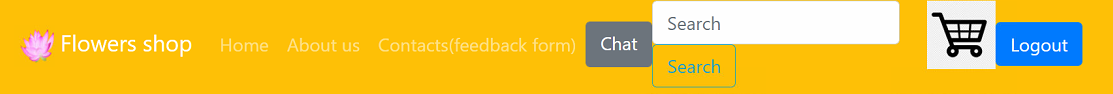


Рисунок 2 – Вид для авторизованного пользователя

* 1. В разделе о нас у нас имеются вкладки – элемент интерфейса, который состоит из 2 частей: самих вкладок и определенного контента, связанного каждого из них. Они используются когда нужно разделить некоторое содержимое на части и выводить пользователю в определенный момент времени только одну из них, которая в данной момент времени связана с активной вкладкой. Они позволяют улучшить организацию данных и внешний вид слайда.

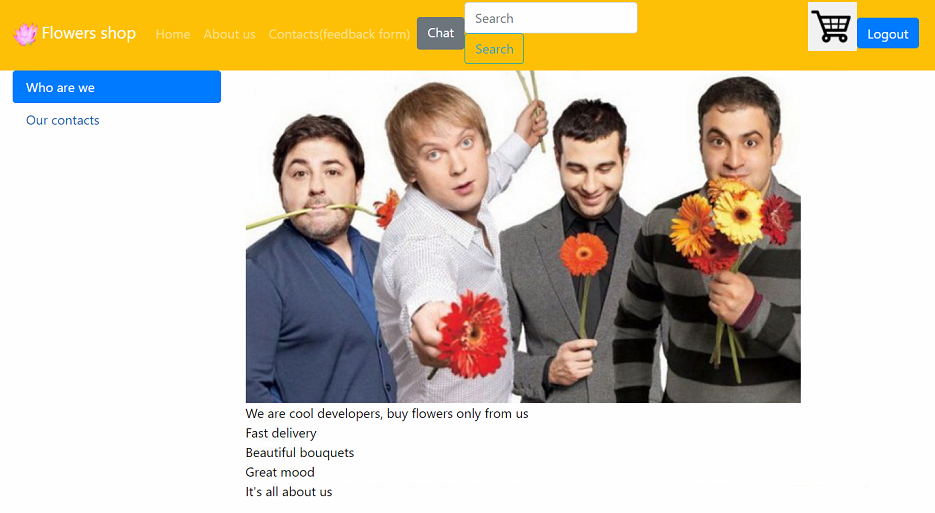


Рисунок 3 – Who are we

* 1. Так же здесь представлены следующие формы:
* Форма обратной связи

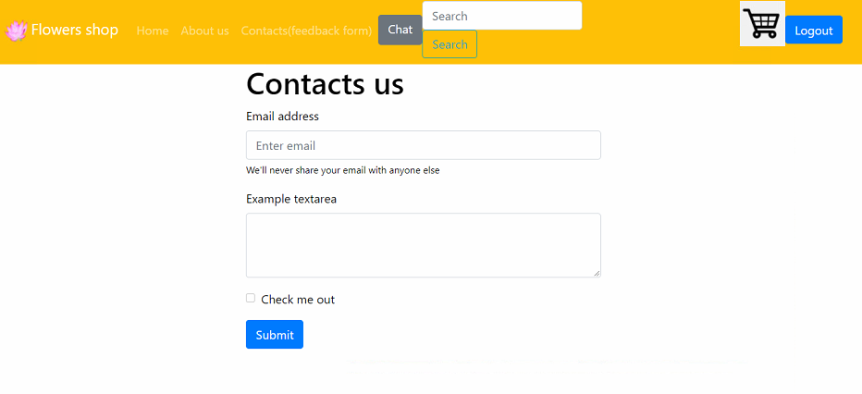


Рисунок 4 – Contact us

* Форма регистрации

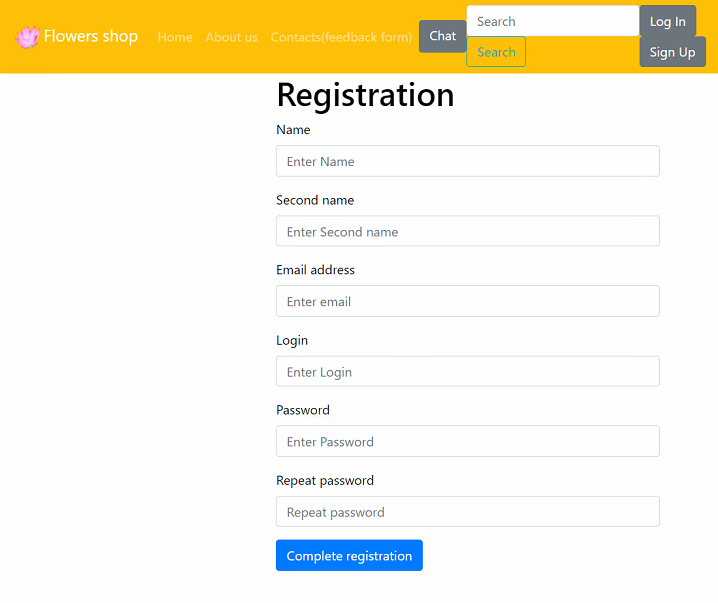


Рисунок 5 – Registration

* Форма авторизации

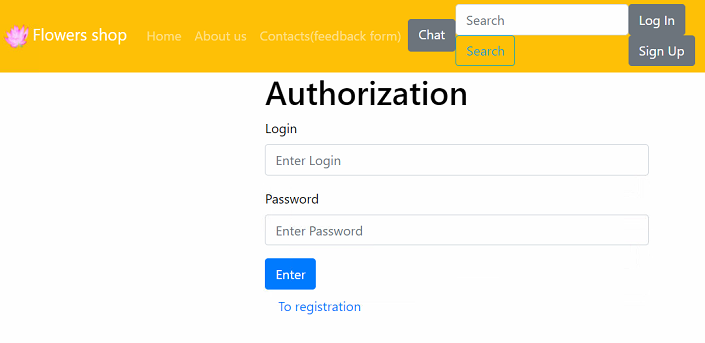


Рисунок 6 – Authorization

* 1. При нажатии на кнопку Home или значок Flowers Shop открывается главная страница, где сразу можно увидеть бутстраповский слайдер, который добавляет интерактива странице. Картинки сменяются друг друга примерно раз в полминуты. Так же чуть ниже у нас отображаются товары (цветы) нашего магазина. Все картинки и информация о них не статична и подгружается из базы данных. Для приемлемого отображения использовались стили css и была произведена работа с таблицами html.

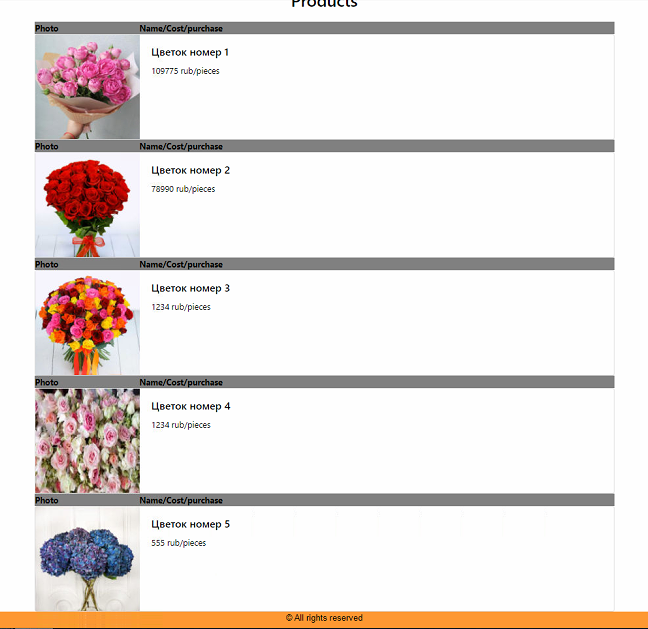


Рисунок 7 – главная страница с товарами

4.5 Так же была исправлена корзина при помощи стилей css. Все данные отображаются в центре.

4.6 При нажатии на шапке сайта по кнопке Chat, пользователь попадает на страницу непосредственно самого чата. Эта кнопка доступна для всех пользователей, как для зарегистрированных , так и для незарегистрированных. На этой странице пользователь вводит свое имя и сообщение.

Чат реализован при помощи платформы Socket.io, а также Node JS и Express. Socket.IO — JavaScript-библиотеки, выполняющие роль программы для обмена данными между клиентами и сервером. Все обновления происходят в режиме реального времени, что позволяет создавать функционал на подобии веб чата, онлайн игры или других мультиплеерных сервисов.

Благодаря сокетам мы можем отправлять данные на один сервер, а далее все клиенты (пользователи), подключенные к этому серверу, будут видеть всю информацию и взаимодействовать с ней. На основе Socket.IO можно создавать как небольшие чат программы, так и полноценные мультиплеерные игры, запускаемые через браузер.

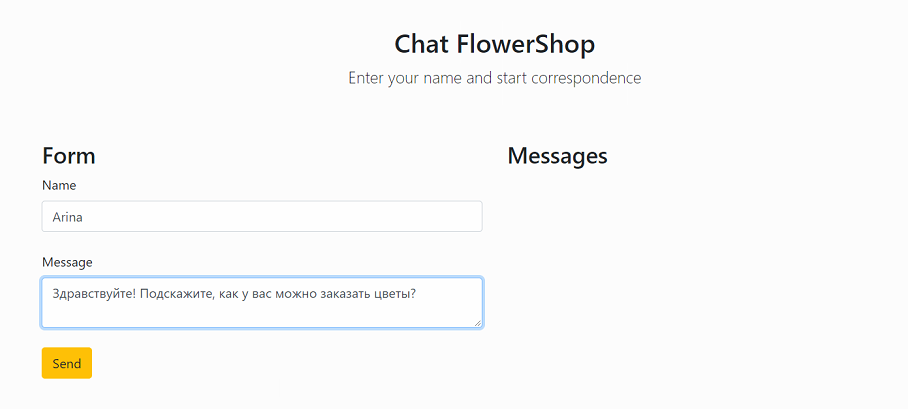


Рисунок 8 – Форма для заполнения пользователем

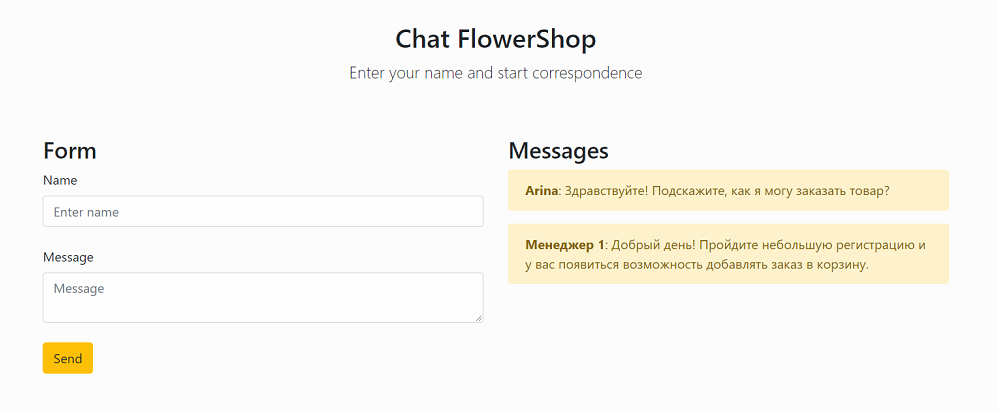


Рисунок 9 – Сообщения

заключение

В рамках выполнения курсового проекта разработано веб-приложение «Цветочный портал». Выполнена логика взаимодействия с backend’ом. Реализованы основные страницы для работы с сайтом. Реализован чат. Были получены практические навыки программирования на react js, так же получены некоторые знания по работе с node js. Получены практические навыки по работе с командами для работы с github.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Github [Электронный ресурс] <https://gist.github.com/rdnvndr/cb21a06c5a71fd71213aed1619380b8e>
2. HTML 5 Book [Электронный ресурс] <https://yandex.ru/search/?text=html%20%D1%82%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D1%8B%20%20%D0%BF%D0%BE%D0%B4%D1%80%D0%BE%D0%B1%D0%BD%D0%BE&lr=10664&clid=2224313&src=suggest_Pers>
3. Официальный сайт по работе с React [Электронный ресурс] <https://reactjs.org/>
4. Учебник. Введение в React [Электронный ресурс] <https://learn-reactjs.ru/tutorial> (дата обращения 06.01.2020)

Приложение