

РАЗРАБОТКА ИНФОРМАЦИОННО-ПОИСКОВОЙ СИСТЕМЫ В СФЕРЕ УСЛУГ

Кухта Р. С.

Научный руководитель – ст. преподаватель Русак Л. В.

Процессы развития глобальных информационно-коммуникационных технологий очень динамичны в настоящее время, а их возможности для общества и экономики еще только начинают масштабно использоваться.

В настоящий «коммуникационный» этап своего развития главной задачей сети интернет является помощь в поиске желательных партнеров и предоставление средств для организации с ними нужного вида коммуникаций с необходимой интенсивностью. Сегодня очень мало организаций в городе, которые не имеют своего сайта в интернете. Развиваются интернет-магазины, интернет-казино, интернет-аукционы. Некоторые фирмы предоставляют возможность заказа продукции или услуги через интернет с оплатой курьеру по факту доставки.

Каждый потребитель ежедневно пользуется различными услугами. Покупка готового кофе, починка обуви, или других вещей, от электронных устройств до зонтика, поездка в автобусе, посещение стоматолога, стрижка в парикмахерской – все это примеры потребления услуг на индивидуальном уровне. К сожалению, следует отметить, что клиенты далеко не всегда довольны качеством и стоимостью услуг, которые они получают. Люди жалуются на несвоевременную доставку, некомпетентный персонал, неудобное время работы организации, низкое качество обслуживания, сложные процедуры предоставления услуг. Чаще всего люди стараются обращаться к знакомым, или с теми к то живет бок о бок за какими-то услугами.

Цели системы:

- упрощение поиска услуг в своем микрорайоне;
- заложить средства интеграции с картографическими сервисами для поиска, и заложить основу для дальнейшего развития системы;
- предусмотреть возможность размещения услуги в разных вариантах;
- создание простого web приложения, которое можно открыть через браузер;
- облегчить обмен услугами ближе к дому и упростить обслуживание населения.

Система будет представлять собой web приложение с регистрацией, и авторизации. Возможностями добавления, удаления, редактирования услуг. Описания оплаты или обмена услугами, возможность работы с разными вариантами услуг, сортировкой и фильтрацией. Возможность интеграции с картографическими сервисами (Google Maps/Yandex). Размещение заявок должно быть разделено на несколько простых этапов.

Проанализировав поставленные задачи, определим ряд следующих технических средств для разработки системы.

Веб-приложение представляет собой клиент-серверное приложение. Клиент взаимодействует с сервером при помощи браузера. Сервер принимает HTTP запросы от клиентов и выдает им HTTP ответы. Логика распределяется между сервером и клиентом. Хранение информации осуществляется в базе данных.

Преимущества веб-приложений следующие:

- установка веб-приложений дешевле и проще. Снижаются затраты на содержание отделов, отвечающих за установку и сопровождение ПО;

- обновление веб-приложений дешевле и проще. Для того чтобы обновить веб-приложение, достаточно его обновить только на сервере;

- веб-приложения универсальны. Они могут работать на любом устройстве, под управлением многими ОС и в любом современном браузере.

Для реализации серверной части приложения был выбран язык программирования Java. В качестве платформы разработки был выбран «Spring Boot Framework». В основном это низкоуровневые компоненты и абстракции.

PostgreSQL выбрана как СУБД. PostgreSQL это СУБД с открытым исходным кодом которая завоевала доверие на рынке разработки как быстрое безопасное и многофункциональное решение для хранения данных. Лицензия распространения – бесплатная.

Для реализации клиентской части приложения была использована открытая и свободная платформа для разработки веб-приложений, написанная на строго типизированном, объектно-ориентированном языке TypeScript – Angular, Angular. Предназначен для разработки одностраничных приложений. Его цель – расширение браузерных приложений на основе MVC-шаблона, а также упрощение тестирования и разработки.

Docker – программное обеспечение для автоматизации развертывания и управления приложениями в средах с поддержкой контейнеризации, контейнеризатор приложений. Позволяет «упаковать» приложение со всем его окружением и зависимостями в контейнер, который может быть развернут на любой системе, а также предоставляет набор команд для управления этими контейнерами.

Система проектируется в виде клиент-серверного приложения. Взаимодействие между клиентом и сервером строится в REST стиле, информация будет передаваться в JSON формате по протоколу HTTP.

В ходе анализа проектируемой системы, были выявлены подсистемы, способные обеспечивать необходимую функциональность, поэтому серверная часть приложения можно построить в рамках модульной монолитной архитектуры, где каждый модуль – это самостоятельная часть системы со свои API, обеспечивающий доступ и обработку информации.

Схема взаимодействия клиент-серверного приложения представлена на рис. 1.

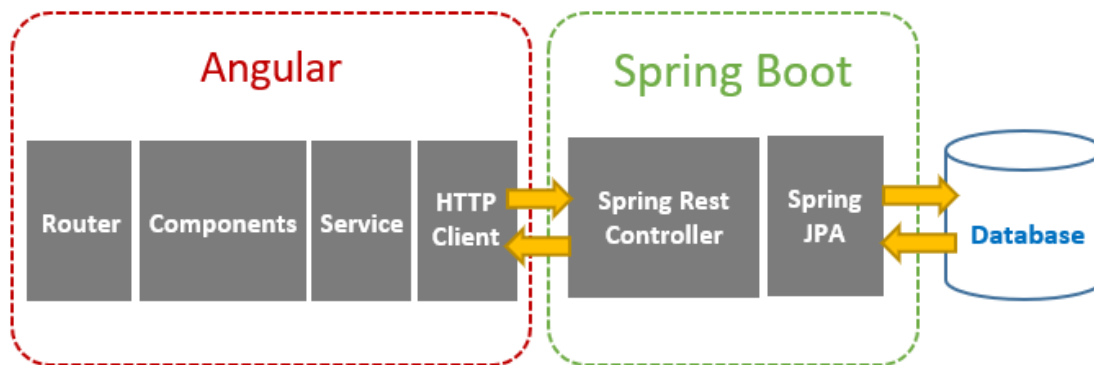


Рис. 1 – Схема взаимодействия клиент-серверного приложения

Модули будут строиться в архитектурном стиле REST. Клиент через общий пользовательский интерфейс получает доступ к отдельно слабо связанным между собой модулям.

Такой подход имеет следующие преимущества:

- простота развертывания. Можно развертывать одну систему с множеством модулей быстрее за счет создания необходимых настроек и файлов для быстрого автоматического развертывания;
- высокая производительность. Системе не требуется большое количество сетевых взаимодействий. Кодовое взаимодействие и работа с данными в модулях происходит в одной оперативной памяти;
- небольшие команды разработки;
- уменьшение дублирования функциональности.

Модули серверной части имеют традиционную для веб-приложений структуру, зачастую описываемую архитектурным паттерном MVC: модель, бизнес-логику и слой представления, через который и происходит обмен данными.

Уровень базы данных представляет собой отдельный сервер, на котором развернута СУБД. База данных хранит в себе всю информацию, необходимую для корректной работы системы (профили пациентов, врачей, документы и т. д.).

Уровень доступа к данным содержит программный модуль отвечающий за взаимодействие ядра приложения и базы данных. Он содержит два набора классов:

- сущности системы;
- классы, отвечающие за CRUD и другие специфические операции над сущностями.

Сервисный уровень – вся бизнес логика приложения. Здесь происходят все основные вычисления, манипуляции над данными и работа алгоритмов системы. Он взаимодействует с уровнем доступа к данным для получения необходимой информации и проведения дальнейших бизнес операций.

Уровень взаимодействия с пользователем представлен в виде классов-контроллеров, которые принимают запросы клиентов и направляют их

необходимому сервису для последующей манипуляции с данными, после возвращает ответ клиенту.

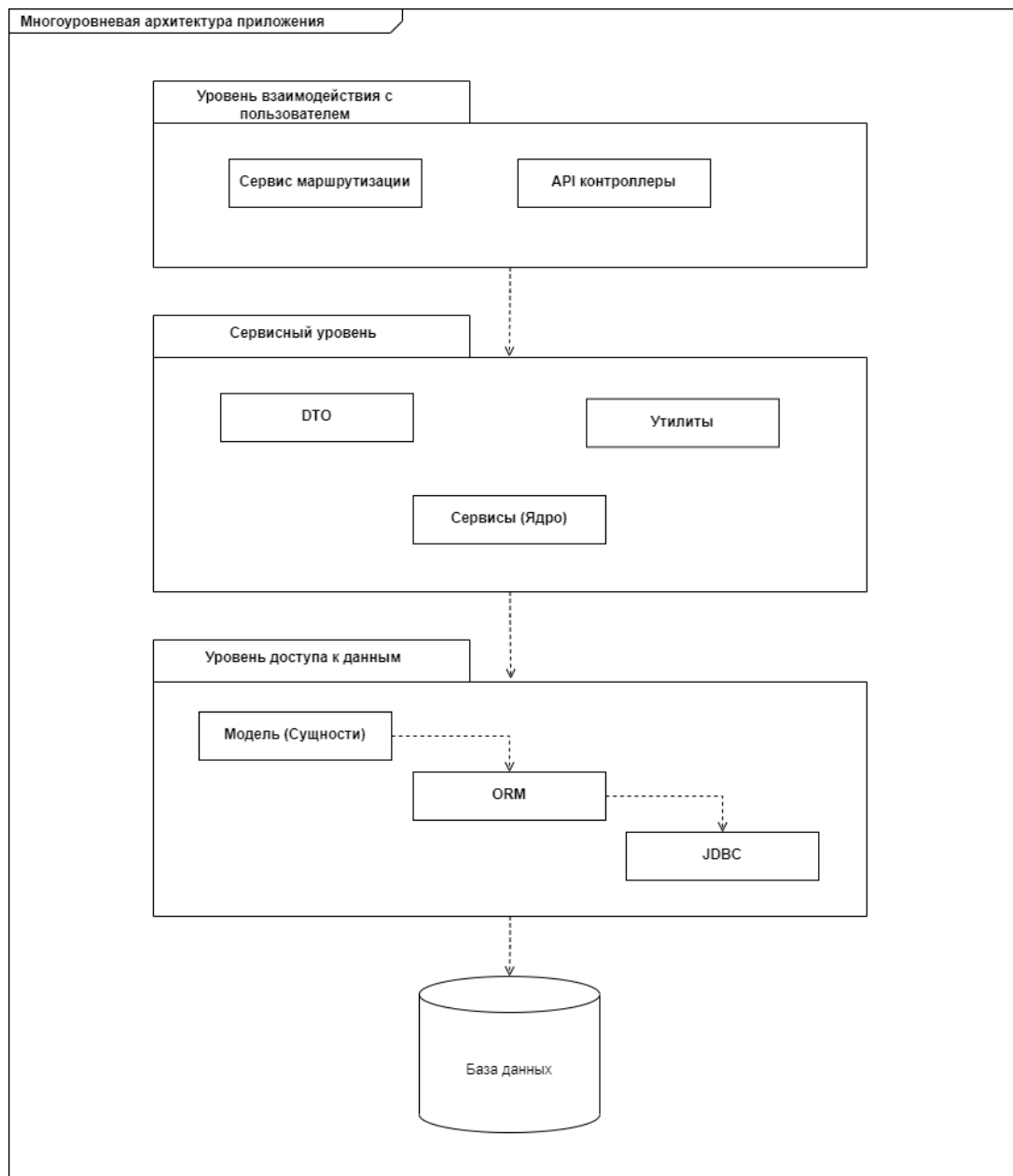


Рис. 2 – Многоуровневая архитектура серверной части

Приложение представляет собой набор отображений (view) услуг. Услуги представлены в виде плитка-образной таблицы. Есть возможность сортировать и фильтровать услуги как показано на рис. 3.

При клике на конкретную услугу открывается отображение этой услуги с деталями и описанием, что можно увидеть на рис. 4. А также варианты услуги. К примеру, там может быть размещена упрощенная версия или же версия с бонусами и дополнительной работой. Добавление к торту надписи или дополнительная полировка обуви после ремонта.

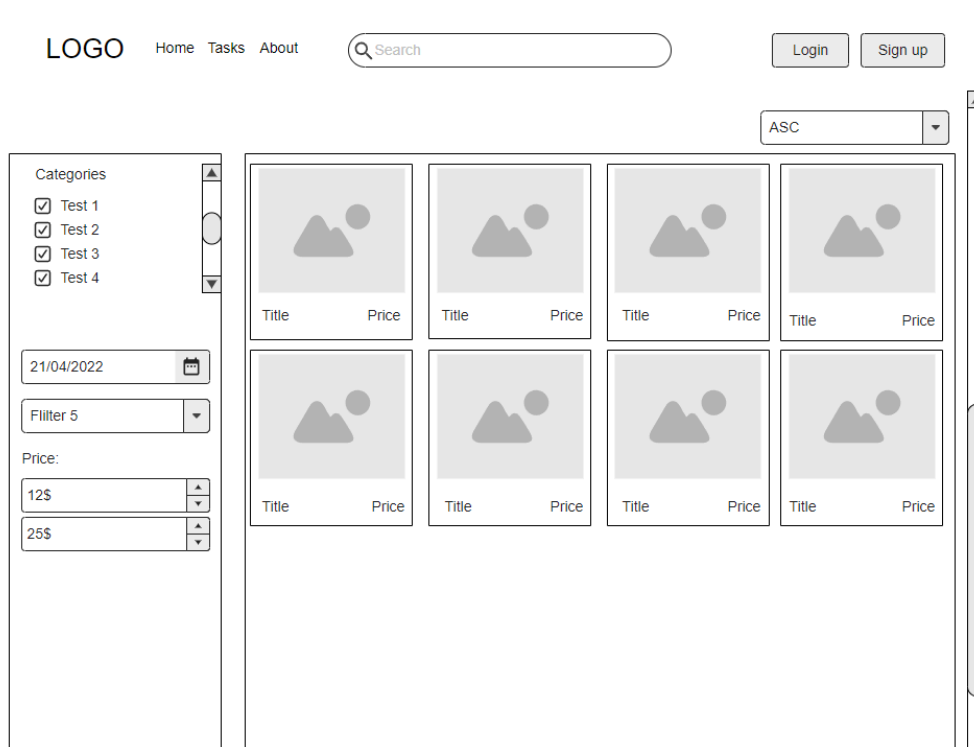


Рис. 3 – Отображение доступных услуг

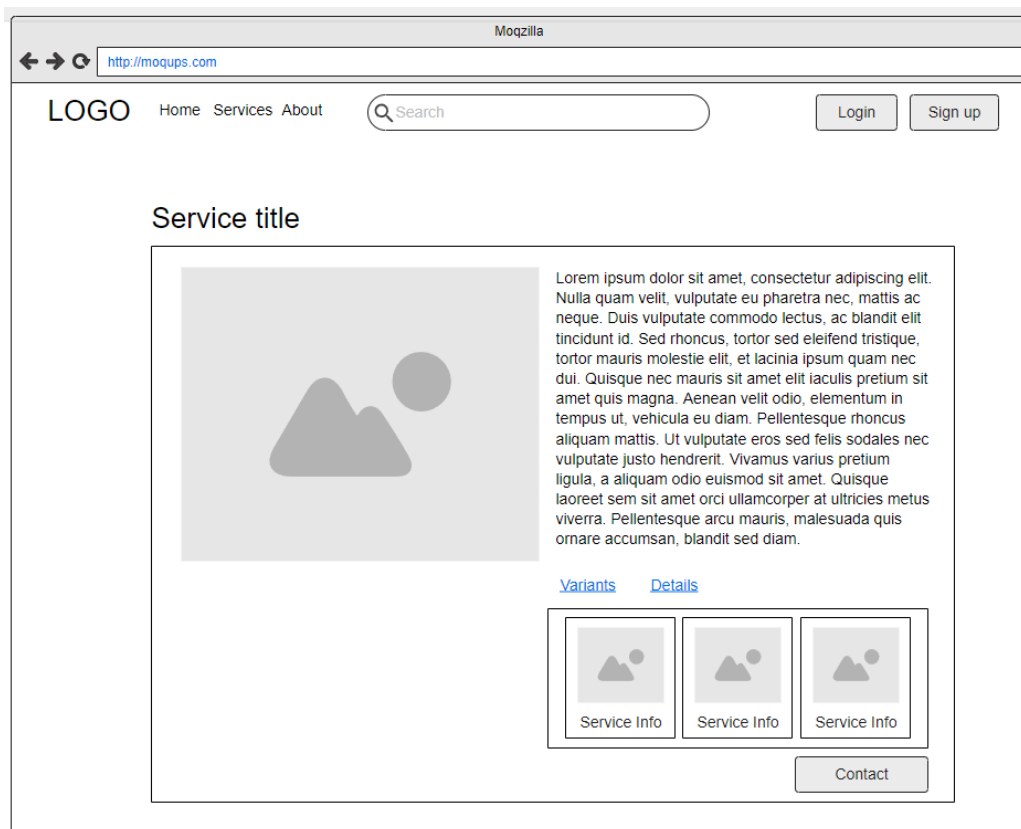


Рис. 4 – Отображение деталей у варианты услуги

Разработанная система позволяет:

- организовывать быстрое взаимодействие представителя услуг (продавца) и клиента (покупателя);
- оптимизировать взаимодействие между участниками системы;
- организовать наибольшее количество людей проживающих на ограниченной территории в товарно-денежном обмене, с учетом минимальных затрат участников.

Использование выбранных технологий на основе результатов проектирования позволит осуществить разработку задуманного программного продукта, отвечающего поставленным требованиям с возможностью дальнейшего расширения.

Литература

1. Angular и TypeScript. Сайтостроение для профессионалов. – СПб.: Питер, 2018. – 464 с.
2. Бушуева Л. И. Роль Интернет-услуг в практической маркетинговой деятельности [Электронный ресурс] / Л. И. Бушуева // Маркетинг в России и за рубежом, 2001. – № 4. URL: <http://www.cfin.ru/press/marketing/2001-4/07.shtml> (дата обращения: 08.05.2022).
3. Spring Boot [Электронный ресурс]. URL: https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm (дата обращения: 08.05.2022).