|  |  |
| --- | --- |
|  | **МИНОБРНАУКИ РОССИИ**  **федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Самарский государственный технический университет»**  **(ФГБОУ ВО «СамГТУ»)** |

Кафедра «Вычислительная техника»

Отчет по дисциплине:

Практико-ориентированный проект

Тема: Разработка сайта продуктового магазина

Выполнил:

Студент 4-ИАИТ-9 Зюзин М.А.

Проверили преподаватели:

Ефимушкина Н.В

Самара 2023

Оглавление

[Введение 3](#_Toc131766960)

[1 Типы сайтов и средства их разработки 7](#_Toc131766961)

[1.1 Классификация сайтов 7](#_Toc131766962)

[1.2 Средства разработки сайтов 11](#_Toc131766963)

[1.3 Структуры сайтов 19](#_Toc131766964)

[1.4 Выводы 21](#_Toc131766965)

[2 Проектирование серверной части сайта 22](#_Toc131766966)

[2.1 Анализ предметной области 22](#_Toc131766967)

[2.2 Проектирование базы данных 23](#_Toc131766968)

[2.3 Разработка программного обеспечения сайта 31](#_Toc131766969)

[2.4 Подсистема анализа продаж 31](#_Toc131766970)

[3 Проектирование пользовательской части сайта 31](#_Toc131766971)

[**3.1** **UML – диаграммы** 31](#_Toc131766972)

[4 ЭКОНОМИЧЕСКАЯ ЧАСТЬ 33](#_Toc131766973)

[5 ОХРАНА ТРУДА 33](#_Toc131766974)

[Заключение 33](#_Toc131766975)

[Графический материал 33](#_Toc131766976)

[Список использованных источников 33](#_Toc131766977)

# Введение

В современном мире, невозможно представить компанию, которая не имеет своего сайта. Сайт предоставляет компании очень много преимуществ, таких как:

* Больший охват клиентов;
* Доступность информации о компании;
* Упрощение связи между клиентом и администратором;
* Быстрое обновление информации о товарах, услугах и новостях компании;
* Доступность каталога товаров;
* Репутационная составляющая

Люди по всему земному шару проводят огромное количество времени в интернете. В глобальной сети сайт или интернет-страница могут быть представлены разными способами. Это определяется задачами, для которых он создается. Для упорядочения информации о сайтах применяется их классификация по целому ряду признаков, наиболее распространенными из которых являются следующие:

1. По назначению:
   1. Коммерческие
   2. Некоммерческие;
2. По типу решаемых задач:
   1. Информационные сайты - новостные, со статьями, блоги;
   2. Социальные сайты – социальные сети, форумы, сайты знакомств
   3. Веб-сервисы – почтовые, хостинги, онлайн-инструменты;
3. По охвату пользователей:
   1. Малый охват
   2. Большой охват
4. По реализуемым технологиям:
   1. Статические
   2. Динамические
   3. Смешанные.
5. По размеру:
   1. Одностраничный, многостраничные
6. По структуре:
   1. Линейная
   2. Иерархическая
   3. Паутинная
   4. Гибридная

Любой сайт состоит из пользовательской и серверной части (frond-end и back-end). На страницах сайтов, расположены текст, кнопки, панели, изображение, видео, за это часть отвечает front-end часть. За логику и функционирование сайта отвечает back-end часть, которая скрыта от пользователя.

Реализация Front-end части происходит с помощью таких языков как:

1. HTML
2. CSS
3. Javascript

Для реализации Back-end части, используются языки:

1. PHP
2. C#
3. Ruby
4. Java

Основной Back-end является база данных - набор постоянно хранимой информации или данных, которые хранятся в электронном виде и используются системами программных продуктов. От архитектуры баз данных зависит скорость работы и ресурса-затратность, они бывают следующих видов:

иерархические;

объектные или объектно-ориентированные;

реляционные;

сетевые.[1]

Для управления базами данных, используется СУБД – система управления базами данных. Наиболее популярными СУБД в настоящее время являются:

1. Oracle
2. MySQL,
3. PostgreSQL
4. Microsoft SQL Server
5. Microsoft Access

Процесс проектирования базы данных делится на логические этапы:

1. Анализ предметной области;
2. Инфологическое проектирование;
3. Даталогическое проектирование;
4. Физическое проектирование.

Целью выпускной квалификационной работы является разработка сайта продуктового магазина. Для достижения этой цели, необходимо решить следующие задачи:

1. Определить целевое назначение и функции сайта;
2. Спроектировать структуру сайта и базы данных;
3. Выбрать средства разработки;
4. Разработать [Back-end часть;](http://wikireality.ru/wiki/Back-end)
5. Разработать Front-end часть.

Исходными данными для решения поставленных задач являются:

1. Номенклатура товаров магазина;
2. Информация о каждом товаре;

Результатами должны быть

1. База данных, содержащая информацию о следующих объектах:

* Товарах,
* Услугах,
* Статистике продаж.

1. Приложение, обеспечивающее реализацию следующих функций:

* Предоставление информации о товарах и акциях, проводимых магазином,
* Выполнение онлайн-заказа,
* Сбор статистики продаж по видам товаров и периодам работы.

После тестирования заказчиком сайт предполагается использовать в одной из торговых организаций города Самары.

# Типы сайтов и средства их разработки

## **1.1 Классификация сайтов**

Сайт – это информационная единица в интернете, ресурс из веб-страниц (документов), которые объединены общей темой и связаны друг с другом с помощью ссылок. Он регистрируется на одно юридическое или физическое лицо и обязательно привязан к конкретному домену, являющемуся его адресом. Сайт может состоять как из одной, так и из огромного количества страниц. Каждая страница – это текстовый файл или их набор, написанный на специальном языке разметки или программирования (HTML, PHP, CSS и пр.). После загрузки на компьютер файлы обрабатываются в браузере. В конечном итоге пользователь видит загруженную страницу сайта. В настоящее время существует широкое разнообразие сайтов, которые могут быть разбиты на классы по следующим признакам.

По назначению сайты принято разделять на следующие классы:

1. Коммерческие;
2. Не коммерческие.

Коммерческие - предназначенные для принесения прибыли компании. Наиболее распространенными представителями такого класса являются интернет-магазины или сайт-визитки. Не коммерческие, предназначенные для безвозмездного предоставления информации, например, новостные, сайты со статьями.

По типу решаемых задач сайты принято делить на следующие классы, изображенные на рисунке 1.

* Информационные;
* Социальные;
* Веб-сервисы.

По информации, предоставляемой сайтами, они делятся на:

* Новостные, которые информируют о событиях, произошедших в социальной жизни человека;
* Со статьями, содержащие большое количество текстов, связанных определенной тематикой;
* Для ведения блогов, позволяющие комментировать и участвовать в дискуссиях.

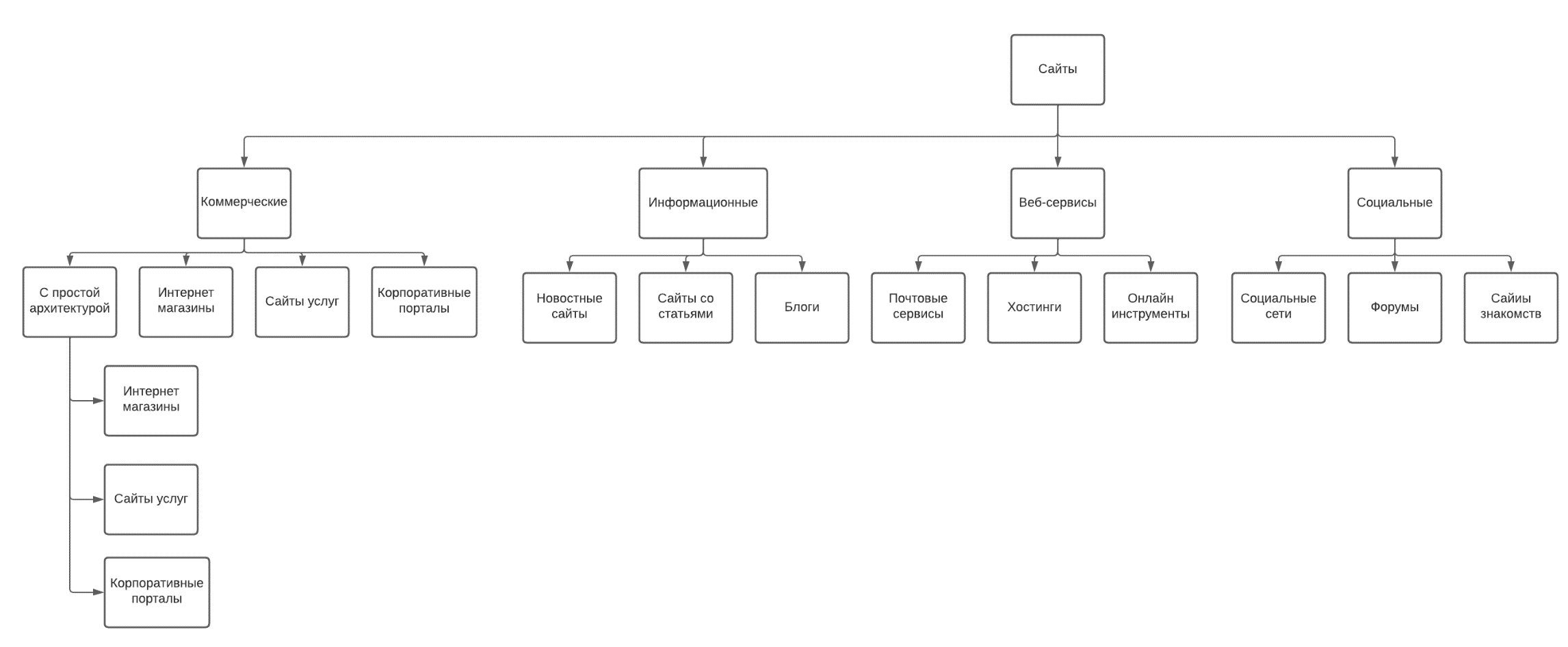


Рисунок 1.1 Классификация сайтов по типу решаемых задач

Среди социальных сайтов выделяют следующие классы:

* Социальные сети, предназначены для общения пользователей;
* Форумы – площадки, созданные для обсуждения определенных тематик;
* Сайты знакомств.

Веб-сервисы подразделяются на следующие классы:

* Почтовые сервисы, обеспечивающие общение между пользователями с помощью текстов или файлов;
* Хостинги – места на серверах, предназначенные для размещения сайтов в интернете;
* Онлайн-инструменты, представляющие пользователям сетевые ресурсы.

По размеру, бывают двух типов:

* Одностраничные – состоящие из одной страницы;
* Многостраничные – содержащие в себе множество страниц.

По реализуемым технологиям сайты делятся на следующие классы:

* Статические, которые имеют постоянную структуру и содержание и поставляются в готовом виде;
* Динамические, содержащие изменяемые страницы. Они генерируются автоматически, и адаптируются к интересам каждого пользователя;
* Смешанные, объединяющие в себе динамические и статические страницы;
* Флеш-сайты – устаревшая технология, созданные на основе программы Adobe Flash;

По охвату пользователей сайты разделяются на две категории:

* Предназначенные для небольшого количества, пользователей. При проектировании таких сайтов, не учитывается возможность появления большого количество пользователь;
* С большим охватом, учитывающие возможность посещения в одно и тоже время большого количества пользователей, что значительно увеличивает нагрузку на аппаратуру.

Коммерческие подразделяются на следующие классы:

* Интернет-магазины – предназначены для продажи товаров или услуг. Имеют сложный функционал, и содержат каталог с описанием продуктов, выбор различных способов оплаты и корзину;
* Сайты услуг – содержат сведения о услугах предоставляемыми определенными компаниями и их контактными данными;
* Корпоративные порталы – ресурсы, предназначенные для предоставления информации о компании, её услугами или товарами.

Таким образом, существует широкое разнообразие сайтов, которые различаются по целому ряду признаков: назначению, типам решаемых задач, охвату пользователей, используемым технологиям и размерам. Наиболее распространенными являются сайты, направленные на получение прибыли – интернет-магазины.

## **1.2 Средства разработки сайтов**

Общепринятая структура сайта включает в себя две основные составляющие:

1. Пользовательскую (frond-end).
2. Серверную (back-end).

Основной задачей первой является привлечение и удержание пользователей с помощью соответствующего дизайна. Она обеспечивает компактное и доступное представление информации. Для разработки пользовательской части применяется программирование на специализированных языках. При этом решаются следующие задачи:

1. Проектирование дизайна;
2. Верстка;
3. Связь с back-end частью.

Используется несколько типов средств для создания дизайна:

1. Готовые компоненты интерфейса, которые потом конвертируются в код.
2. Языки программирования высокого уровня.
3. Специальные конструкторы сайтов.

Первый и третий способы не обеспечивают возможности тонкой настройки компонентов, а второй позволяет создавать сайты любого типа с любым функционалом. Для решения поставленных в работе задач целесообразно использовать именно этот способ.

Основными языками программирования для веб-разработки являются:

1. HTML - стандартизированный язык гипертекстовой разметки документов для просмотра [страниц](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0) в [браузере](https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80). Веб-браузеры получают HTML документ с сервера по протоколам [HTTP](https://ru.wikipedia.org/wiki/HTTP)/[HTTPS](https://ru.wikipedia.org/wiki/HTTPS) или открывают с локального диска, этот документ реализует пользовательский интерфейс. [1]
2. CSS - [формальный язык](https://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) описания внешнего вида документа ([веб-страницы](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0)), написанного с использованием [языка разметки](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8). Он может применяться к любым [XML-документам](https://ru.wikipedia.org/wiki/XML), например, к [SVG](https://ru.wikipedia.org/wiki/SVG) или [XUL](https://ru.wikipedia.org/wiki/XUL) . [2]
3. JavaScript поддерживает [объектно-ориентированный](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), [императивный](https://ru.wikipedia.org/wiki/%D0%98%D0%BC%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) и [функциональный](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) стили. Является реализацией спецификации [ECMAScript](https://ru.wikipedia.org/wiki/ECMAScript). JavaScript обычно используется как встраиваемый язык для программного доступа к объектам [приложений](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0). Наиболее широкое применение он находит в [браузерах](https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80) как язык сценариев для придания [интерактивности](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%B0%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D1%81%D1%82%D1%8C) [веб-страницам](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0) [3]

Для CSS и JavaScript существует множество фреймворков, наиболее популярным является Bootstrap 4, это - открытый и бесплатный HTML, CSS и JS фреймворк, который используется веб-разработчиками для быстрой вёрстки адаптивных дизайнов сайтов и веб-приложений. Он является самым популярным. [4]

Back-end включает в себя внутреннюю часть сайта и сервера. Основные процессы, которые реализует back-end:

1. Получение данных с клиентской стороны;
2. Обработка их сервером;
3. Возвращение обработанных данных обратно пользователю.

Наиболее популярной архитектурой распределенных систем, таких как World Wide Web, в настоящее время является REST (Representational state transfer). Каждая единица информации в ней однозначно определяется глобальным идентификатором, таким как URL. Каждая URL, в свою очередь, имеет строго заданный формат. [5] Для отправки и принятия URL-запросов, в большинстве сайтов используется протокол HTTP.

Этот протокол лежит в основе обмена данными в Интернете и позволяет получать различные ресурсы, например, HTML-документы. HTTP является протоколом клиент-серверного взаимодействия, и инициирует запросов к серверу самим получателем, обычно веб-браузером. Полученный документ может состоять из различных частей, например, отдельно полученного текста, описания структуры документа, изображений, видеофайлов, скриптов и многого другого. [6]

Наиболее популярными языками для разработки серверной части сайтов являются следующие.

1. PHP — распространённый язык программирования общего назначения с открытым исходным кодом. Он специально сконструирован для веб-разработок и его код может внедряться непосредственно в HTML. [7]
2. С# и ASP.NET. C# - один из наиболее и востребованных языков программирования. Фреймворк ASP.NET разработан для C# и платформы .NET. Он позволяет создавать веб приложения. [8]
3. Ruby и Ruby on Rails, динамический, рефлективный, интерпретируемый высокоуровневый язык программирования. Он обеспечивает независимую от операционной системы реализацией многопоточности, динамическую типизацию, сборщик мусора и многие другие возможности. [9].
4. Java — универсальный объектно-ориентированный язык, который часто используется для веб-разработки. Программы на Java транслируются в байт-код, который затем выполняется виртуальной машиной Java. [10]. Реализация back-end логики на языке Java, обеспечивает множество преимуществ по сравнению с другими языками, а фреймворк Spring, расширяет его возможности, особенно в веб разработке.

Основная информация, необходимая для работы сайта, в крупных проектах хранится в базах данных. Для управления этими базами используются СУБД:

База данных – набор постоянно хранимой информации или данных, которые хранятся в электронном виде и используются системами программных продуктов. Для управления базами данных, используется СУБД – система управления базами данных. СУБД и хранящиеся данные, для краткости, называются базы данных. Существует огромное количество разновидностей баз данных, различающихся по различным [критериям](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D1%82%D0%B5%D1%80%D0%B8%D0%B9). Например, в «Энциклопедии технологий баз данных»[[2]](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85#cite_note-KogMR-7), по материалам которой написан данный раздел, определяются свыше 50 видов БД.

В классификацию по [модели данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) обычно включают:

* [иерархические](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85);
* [объектные](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) или [объектно-ориентированные](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85);
* [реляционные](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85);
* [сетевые](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).[1]

В основе иерархической базы данных лежит древовидная структура. Такой вид хранения данных подразумевает комплекс объектов, расположенных по принципу их подчинения – от общего к частному. Такая структура одна из самых распространенных, ввиду ее целостностью и простоты. Однако, для работы с таким типом баз данных, необходимо большое количество основной и дисковой памяти. Пример иерархической базы данных представлен на рисунке 1.2.

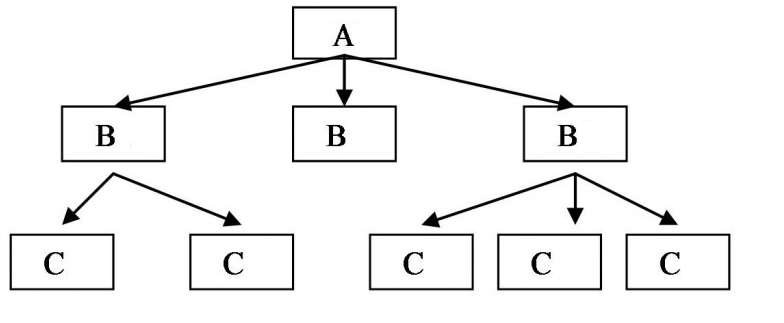


Рисунок 1.2 Иерархическая структура базы данных.

[Объектные](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) или [объектно-ориентированные](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) базы данных, хранят в себе информацию в виде объектов, как в объектно-ориентированных языках программирования. Преимущества такой архитектуры в отсутствии несоответствия модели в приложении и базе данных, а также не требуется поддерживать модель данных на стороне СУБД. Однако, к такому вид базы данных, прибегают, когда требуется высокопроизводительная обработка данных со сложной структурой.

Базы данных с реляционной структурой совмещают в себе преимущества иерархических и объектно-ориентированных структур, благодаря чему, является универсальной архитектурой, подходящей под большинство задач. Пример такой архитектуре изображен на рисунке 1.3



Рисунок 1.3 Реляционная структура базы данных.

Наиболее популярными СУБД в настоящее время являются:

1. Oracle – популярный у разработчиков продукт, который способен обрабатывать данные больших объемов, поддерживает язык SQL;
2. MySQL, который использует стандартную форму SQL, имеет интуитивно понятный интерфейс и позволяет хранить большое число записей в таблицах. Он обладает высокой скоростью, поддерживает большинство ОС и гарантирует безопасность данных;
3. PostgreSQL – масштабируемая, реляционная база данных, работающая на большинстве ОС, она имеет большое количество функций;
4. Microsoft SQL Server – популярная СУБД, которая применяется только для ОС Windows, не имеет графического интерфейса, но так же поддерживает SQL запросы;
5. Microsoft Access - NoSQL СУБД, которая сочетает в себе реляционное ядро базы данных Microsoft Jet и имеет простейший интерфейс.

Наиболее перспективной для разработки веб-сайтов представляется PostgreSQL.

Процесс проектирования БД, начинается с неформального, словесного описания предметной области, а заканчивается на формализованном описании объектов предметной области в специализированных терминах. Результатом таких действий, является построение БД отвечающей требованиям предметной области. Хорошо спроектированные базы данных значительно упрощают работу с ней. Процесс проектирования базы данных делится на логические этапы:

* Анализ предметной области. Анализ предметной области разделяется на два подхода – функциональный и предметные подходы. Функциональный подход – подход, который используется при наличии представления о задачах и способах их решения, для которых создается база данных. Выделяется минимальный объем необходимый для успешного старта работ. Дальнейшее усложнение структуры базы данных осуществляется на более поздних этапах разработке – по мере требования. Предметный подход – подход, который используется, когда представления о назначении четко не фиксируется, и в дальнейшем может кардинально поменяться. В таком случае, в описание предметной области ложатся базовые задачи, характерные для текущей предметной области.
* Инфологическое проектирование - формализованное описание объектов предметной области в терминах определенной семантической модели.
* Даталогическое проектирование - разработке схемы БД, т.е. совокупности схем отношений, которые адекватно моделируют объекты предметной области и семантические связи между объектами. Основой анализа корректности схемы являются функциональные зависимости между атрибутами БД.
* Физическое проектирование

Для приложений веб-сайтов наиболее популярными являются следующие IDE (Интегрированную среду разработки) на Java:

1. IntelliJ IDEA – это интеллектуальная среда, учитывающая контекст. Она предназначена для разработки разнообразных приложений на Java и других языках JVM. Кроме того, IntelliJ IDEA Ultimate помогает в разработке веб-приложений: она предлагает эффективные встроенные инструменты, поддержку JavaScript и связанных с ним технологий, а также расширенную поддержку таких популярных фреймворков, как Spring и Spring Boot. Бесплатные плагины позволяют дополнительно расширить возможности IntelliJ IDEA и использовать ее для работы с другими языками программирования, в том числе Go, Python, SQL, Ruby и PHP. [11].
2. Eclipse – [свободная](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) [интегрированная среда разработки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8) [модульных](https://ru.wikipedia.org/wiki/Plug-in_Development_Environment) [кроссплатформенных](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D1%8C) приложений. Развивается и поддерживается [Eclipse Foundation](https://ru.wikipedia.org/wiki/Eclipse_Foundation). [12].

Одним из главных преимуществ IntelliJ IDEA перед Eclipse, является ultimate версия, которая поставляется на коммерческой основе, с возможностью получения бесплатных учебных лицензий и имеет встроенные инструменты для разработки веб-приложений

Таким образом, разработка сайта разделяется две основные составляющие: пользовательскую и серверную. Для их разработки используются различные средства. Так, пользовательскую часть целесообразно реализовать на JavaScript с фреймворком Bootstrap 4. Серверная часть должна иметь наиболее популярную архитектуру распределенных систем REST. В качестве среды разработки предлагается использовать IntelliJ IDEA для Java с фреймворком Spring Web MVC. Наиболее перспективной для разработки веб-сайтов признана СУБД PostgreSQL, с использованием реляционной структурой базы данных.

## **1.3 Структуры сайтов**

Структура сайта — это логическая связка страниц, расположение конкретных элементов дизайна, которые должны следовать стандартам разработки. В настоящее время наибольшее распространение получили следующие структуры:

1. Линейная
2. Иерархическая
3. Паутинная
4. Гибридная

Линейная структура – простая структура, в которой все связи между страницами – последовательная. Общий вид такой структуры представлен на рисунке 2.1. Такая схема применяется в сайтах презентациях и портфолио. В таких видах структур, страницы расставляются цепочкой, звенья которой взаимосвязаны, однако из-за этой особенности, подобная схема не удобна для продвижения, потому что реклама распространяется только на главную страницу. Основной плюс сайтов такого вида состоит в простоте их разработки.

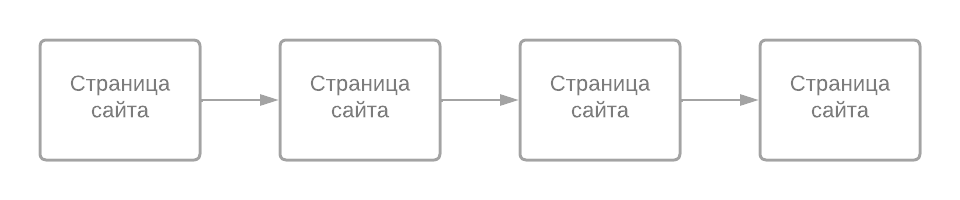


Рисунок 2.1 - Линейная структура сайта

Иерархия – структура, в которой связи между страницами выполнены в форме дерева. Она приведена на рисунке 2.2. При такой структуре, каждому разделу отводится отдельная ветвь.

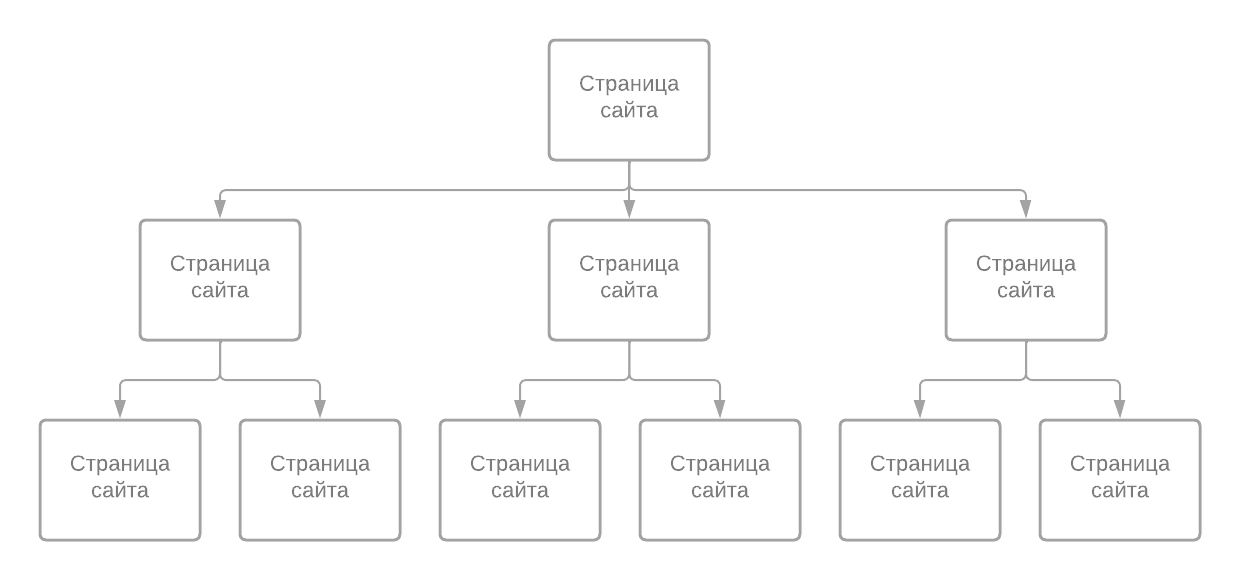


Рисунок 2.2 - Иерархичная структура сайта

Паутина – структура, в которой каждая страница связана со всеми остальными, как показано на рисунке 2.3. Такая схема сайта зачастую применяется на информационных порталах, ввиду сложности выстраивания логических связей.

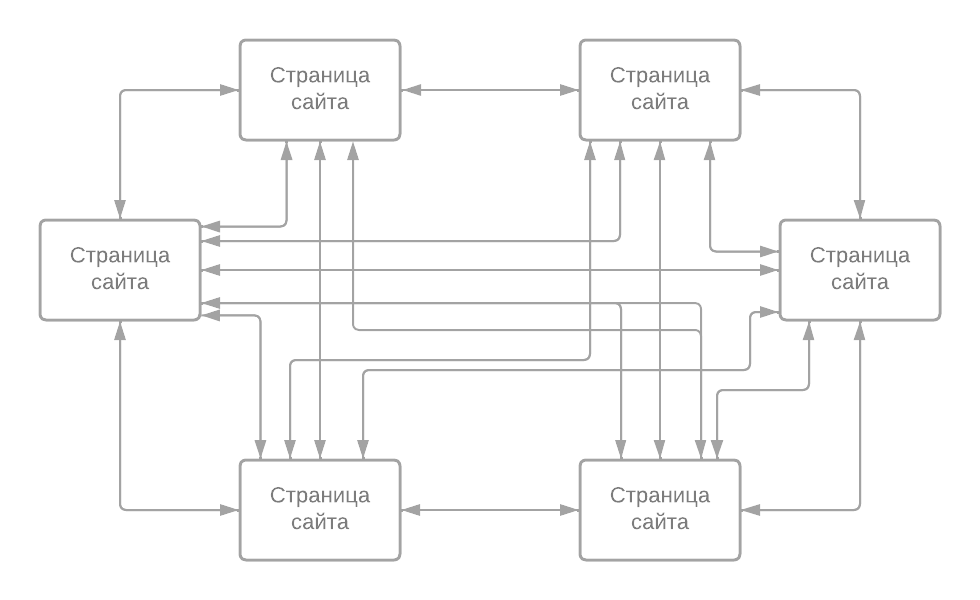


Рисунок 2.3 - Паутинная структура сайта

Приведенные структуры являются базовыми и редко используются в чистом виде. В современных сайтах обычно применяют их комбинацию. Такой вид структуры называется гибридной, из-за ее универсальности, она применяется при проектировании большинства сайтов, хотя проектирование такой схемы трудоемкий процесс.

Таким образом, гибридная структура сайта, позволит реализовать сайт любой сложности, с любыми возможными связями страниц, что сделает гарантирует комфортное взаимодействия пользователей и сайта.

## **1.4 Выводы**

Существует широкое разнообразие сайтов, которые различаются по целому ряду признаков: назначению, типам решаемых задач, охвату пользователей, используемым технологиям и размерам. Наиболее распространенными являются сайты, обеспечивающие продажу товаров или предоставление услуг – интернет-магазины.

Разработка сайта разделяется две основные составляющие: пользовательскую и серверную. Для их разработки используются различные средства. Так, пользовательскую часть целесообразно реализовать на JavaScript с фреймворком Bootstrap 4. Серверная часть должна иметь наиболее популярную архитектуру распределенных систем REST. В качестве среды разработки предлагается использовать IntelliJ IDEA для Java с фреймворком Spring Web MVC. Наиболее перспективной для разработки веб-сайтов признана PostgreSQL. Для обеспечения наибольшего комфорта пользователей, структура сайта должна быть гибридной.

# Проектирование серверной части сайта

## **Анализ предметной области**

Основной задачей магазине, является перевод посетителей в покупателей. Для ее решения необходимо проанализировать основные признаки и психологию покупателя, по данным портала [R](https://www.rbc.ru/technology_and_media/10/10/2021/61618e229a7947975cf67c04)BC, на 2021 год, типичным портретом покупателя продуктового интернет-магазина является [13]:

1. Женщина (по статистике, количество женщин, совершающих покупки в интернете, преобладает над количеством мужчин);
2. Приблизительный возраст потенциального покупателя составляет 25-34 года (на эту возрастную категорию пришлось 26,6% покупателей);
3. Средний доход, составляет меньше 20 тыс. руб.;
4. Место проживания покупателей – населенные пункты с числом жителей меньше 100 тыс. (на покупки из таких городов приходится 35,8%);
5. Большинство покупателей – замужем (63,3%) и не имеют детей (40,2%);
6. Они, посещают интернет-магазины с мобильных устройств, в промежутке от 6 до 10 часов вечера.

Проанализировав этот портрет покупателя, можно сделать вывод о том, что магазин должен предлагать продукты первой необходимости, а также выдерживать большой трафик в вечерние часы.

По данным [Роспотребнадзора](http://04.rospotrebnadzor.ru/index.php/press-center/healthy-lifestyle/13599-17112020.html), в перечень продуктов первой необходимости входят следующие товары [14]:

* Зеленые овощи;
* Яблоки;
* Рис;
* Бобовые;
* Орехи;
* Рыба;
* Кисломолочные продукты;
* Крупы;
* Мясные продукты: свинина, говядина, баранина, куриное мясо;
* Куриные яйца;
* Изделия из пшеничной муки;
* Чай;
* Макаронные изделия;
* Картофель;
* Репчатый лук;
* Белокочанная капуста;
* Морковь.

А следующие продукты, не относятся к категории первой необходимости, поэтому они могут отсутствовать в номенклатуре товаров:

* Мюсли;
* Фитнес-батончики;
* Йогурты с наполнителями;
* Обезжиренные молочные продукты;
* Соки и нектары в коробках;
* Соевые продукты.

## **Проектирование базы данных**

Основной составляющей серверной части является база данных, в которой должна храниться информация о товарах, покупателях и статистике покупок. Она должна содержать следующие сущности:

1. Пользователь, у которого есть электронный адрес, имя, пароль и указатель на уровень доступа к интернет-магазину.
2. Продукт, у которого есть название, описание, цена и изображение.
3. Заказ, в котором хранятся сведения о адресе, общей цены заказа, статусе, продуктах, которые в него входят, их количестве, их цене
4. Корзина, в которой хранится информация о товарах, которые добавил каждый пользователь в свою корзину

Для проектирования базы данных построена модель сущность связь, которая приведена на рисунке 2.1.

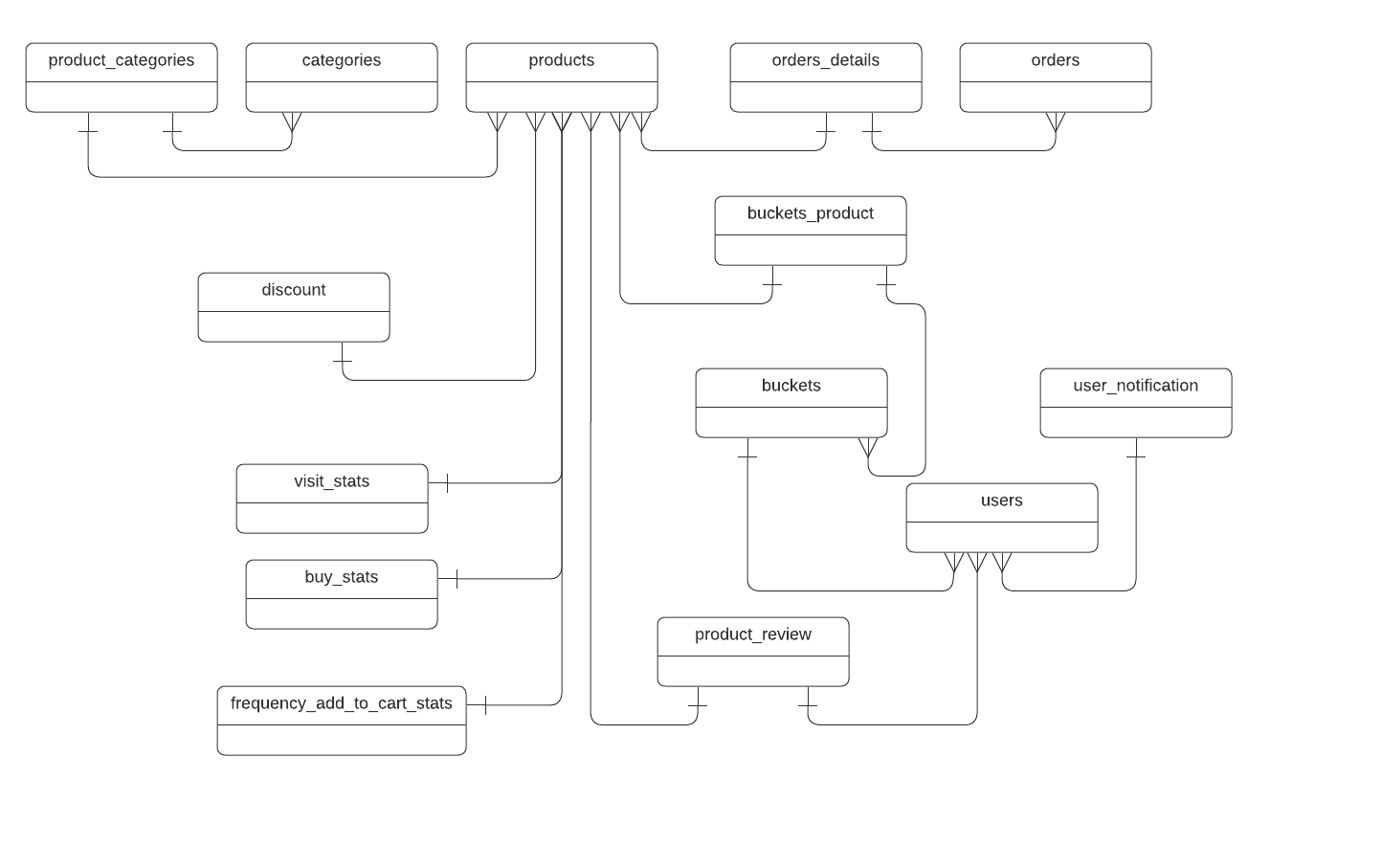


Рисунок 2.1 Er-диаграмма

По рис 2.1 построена даталогическая модель базы данных, в ней используются следующие таблицы:

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Activation\_code | код активации, для подтверждения владения пользователем указанной при регистрации электронный почты | Varchar (255) |
| Email | Электронный адрес пользователя, который в дальнейшем, будет использоваться для отправки уведомлений | Varchar (255) |
| Name | Имя пользователя, который у него будет в профиле, а также, будет отображаться в отзывах на продукты | Varchar (255) |
| Password | Пароль, под которым пользователь будет заходить в свой аккаунт | Varchar (255) |
| Role | Роль пользователя на сайте, в зависимости от которой, он будет иметь различный уровень доступа | Varchar (255) |

Таблица 2.1 User\_notification

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Message | Сообщение, содержащее реакцию на действие пользователя | Varchar (255) |
| url | Ссылка, если реакция подразумевает ее наличие | Varchar (255) |
| url\_text | Текст, по клику на который будет осуществлен переход на новую страницу по ссылке из поля url | Varchar (255) |
| User\_id | уникальный идентификатор пользователя, которому адресовано сообщение | Varchar (255) |

Таблица 2.2 User\_notification

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Description | Описание товара | Varchar (255) |
| Image | Url, который указывает на местоположение картинки | Varchar (255) |
| Price | Цена продукта | double |
| Title | Название продукта | Varchar (255) |

Таблица 2.3 Products

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Discount\_price | Цена продукта со скидкой | double |
| Product\_id | Уникальный идентификатор продукта | Varchar (255) |

Таблица 2.4 Discount

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Review | Текстовое описание продукта, на которое пользователь оставил отзыв | Varchar (255) |
| Stars | Количество звезд, которые поставил пользователь товару | Integer |
| Product\_id | Уникальный идентификатор продукта, на который пользователь оставил отзыв | BigInt |
| User\_id | Уникальный идентификатор пользователя, который оставил отзыв | BigInt |

Таблица 2.5 Product\_review

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| title | Наименование категории | Varchar (255) |

Таблица 2.6 Categories

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Product\_id | Уникальный идентификатор продукта | BigInt |
| Category\_id | Уникальный идентификатор категории | BigInt |

Таблица 2.7 Products\_Categories

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Address | Адрес, куда будет осуществлена доставка заказа | Varchar (255) |
| Created | дата создания заказа | timestamp |
| Status | статус заказа | Varchar (255) |
| Sum | общая сумма заказа | Double |
| Updated | дата обновления статуса заказа | Timestamp |
| User\_id | уникальный идентификатор пользователя, который совершил заказ | BigInt |

Таблица 2.8 Orders

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Amount | Количество продукта, в единицах | integer |
| Price | Цена с учетом количества | Double |
| Product\_id | Уникальный идентификатор продукта | BigInt |
| Order\_details\_id | Уникальный идентификатор заказа | BigInt |

Таблица 2.9 OrderDetails

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| User\_id | Уникальный идентификатор пользователя, которому принадлежит корзина | BigInt |

Таблица 2.10 Buckets

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Bucket\_id | Уникальный идентификатор корзины | BigInt |
| Product\_id | Уникальный идентификатор продукта, который находится в корзине | BigInt |

Таблица 2.11 Buckets\_product

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Created | Дата создания | Timestamp |
| Product\_id | Уникальный идентификатор продукта | BigInt |

Таблица 2.12 Visit\_stats

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Amount |  | Integer |
| Created | Дата создания | Timestamp |
| Product\_id | Уникальный идентификатор продукта | BigInt |

Таблица 2.13 Buy\_stats

|  |  |  |
| --- | --- | --- |
| Наименование | Содержание | Тип |
| Id | Уникальный идентификатор | BigInt |
| Created | Дата создания | Timestamp |
| Product\_id | Уникальный идентификатор продукта | BigInt |

Таблица 2.14 Frequency\_add\_to\_cart\_stats

## **2.3 Разработка программного обеспечения сайта**

Руководство пользователя

Привести экранные формы сайта и особенности навигации по нему.

## **2.4 Подсистема анализа продаж**

Описать, какие показатели являются важными для работы магазина: объем продаж в денежном выражении, в натуральных единицах, спрос по временам года, месяцам и дням и пр. Привести и обработать статистику.

# Проектирование пользовательской части сайта

## **UML – диаграммы**

При проектировке сайта, в первую очередь необходимо обратить внимание на анализ допустимых действий пользователя на страницах. С помощью такого анализа, можно увидеть точки входа и выхода с портала, сформировать поведенческие паттерны пользователей на сайте, выявить неудобные аспекты в дизайне или логистике сайта, с целью дальнейшего улучшения. В современных архитектурах любых веб-сервисов, используется понятие «Роль», которая назначается каждому пользователю, зарегистрировавшемуся на портале. Допустимые действия пользователей, с учетом их ролей, представлены на рисунке 3.1.

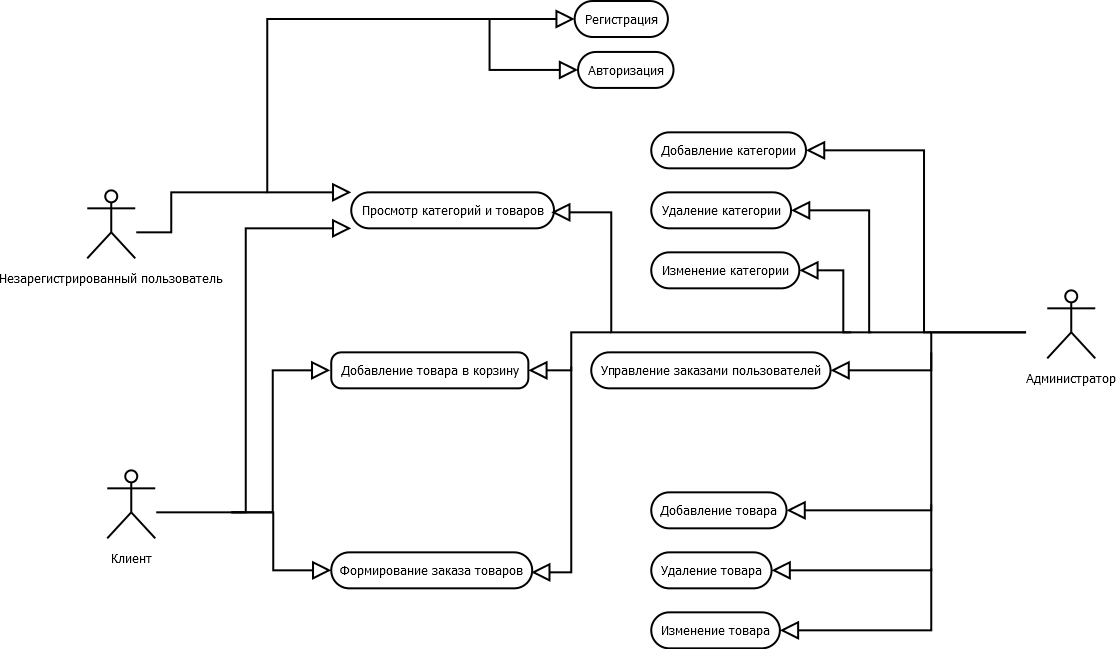


Рисунок 3.1 UML диаграмма

Пользователи, которые не прошли авторизацию имеют самый низкий уровень доступа. В контексте интернет-магазина, они могут посещать только главную страницу, страницу с товарами и окно регистрации.

Следующая роль – клиент. Эта роль присваивается основной массе пользователей, и имеет доступ к личной корзине, возможности добавления продукта в корзину, возможности формирования заказа, а также отслеживания личных заказов.

Роль, которой доступен наибольший функционал сайта – администратор. Администратору разрешается добавлять, удалять или редактировать карточки с товарами, получать информацию о заказах всех пользователей, а также изменять их статусы.

# ЭКОНОМИЧЕСКАЯ ЧАСТЬ

# ОХРАНА ТРУДА

Вопросы:

1. Правильно ли оформлена нумерация рисунков и таблиц (шрифт)?
2. Какие подпункты должны находиться в 3 пункте?
3. Нужно комментировать таблицы из базы данных (2.1-2.14)
4. Что нужно писать в пункте 2.3
5. Красная строка должна быть везде, даже в такой ситуации:

…по материалам которой написан данный раздел, определяются свыше 50 видов БД.

В классификацию по [модели данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) обычно включают:

* [иерархические](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85);

…

1. Новая глава должна начинаться с новой страницы?
2. Глава 3.1, должна называться UML-диаграммы?

Обсудить

1. Как именно нужно описывать работу в пунктах разработка программного обеспечения\пользовательской части сайта.
2. Наполнение 3 главы

# Заключение

«В ВКР разработаны структура и программное обеспечение сайта продуктового магазина, занимающегося торговлей Мясом и мясными продуктами, а также ….. Сайт является корпоративным и имеет иерархическую структуру. Он содержит следующие основные подсистемы (перечислить). Сайт реализован с помощью следующих средств ….» Перечислить основные характеристики сайта (функции, объемы используемых ресурсов и пр.).

# Графический материал

Слайды к докладу по ВКР – обсудим, когда будет написана записка.

# Список использованных источников

|  |  |
| --- | --- |
| [1] | «3,» [В Интернете]. Available: https://ru.wikipedia.org/wiki/HTML. |
| [2] | «2,» [В Интернете]. Available: https://ru.wikipedia.org/wiki/CSS. |
| [3] | «1,» [В Интернете]. Available: https://ru.wikipedia.org/wiki/JavaScript. |
| [4] | «15,» [В Интернете]. Available: https://itchief.ru/bootstrap/introduction. |
| [5] | «4,» [В Интернете]. Available: https://habr.com/ru/post/38730/. |
| [6] | «5,» [В Интернете]. Available: https://developer.mozilla.org/ru/docs/Web/HTTP/Overview. |
| [7] | «7,» [В Интернете]. Available: https://www.php.net/manual/ru/intro-whatis.php. |
| [8] | «8,» [В Интернете]. Available: https://timeweb.com/ru/community/articles/chto-takoe-csharp. |
| [9] | «9,» [В Интернете]. Available: https://ru.wikipedia.org/wiki/Ruby. |
| [10] | «10,» [В Интернете]. Available: https://web-creator.ru/articles/java. |
| [11] | «13,» [В Интернете]. Available: https://www.jetbrains.com/ru-ru/idea/features/. |
| [12] | «14,» [В Интернете]. Available: https://ru.wikipedia.org/wiki/Eclipse. |
| [13] | «11,» [В Интернете]. Available: https://www.rbc.ru/technology\_and\_media/10/10/2021/61618e229a7947975cf67c04. |
| [14] | «10,» [В Интернете]. Available: http://04.rospotrebnadzor.ru/index.php/press-center/healthy-lifestyle/13599-17112020.html. |
| [15] | «11,» [В Интернете]. Available: https://habr.com/ru/post/490586/. |
| [16] | «12,» [В Интернете]. Available: https://www.nic.ru/help/chto-takoe-subd\_8580.html. |
| [17] | «6,» [В Интернете]. Available: https://ru.wikipedia.org/wiki/JSON. |