ВВЕДЕНИЕ

Сейчас идёт XXI век – век технологий и инноваций. На сегодняшний момент нельзя представить жизнь без компьютеров, различных электронных приборов и техники. Информационные технологии играют важную роль в современном обществе. На данный момент они присутствуют во всех сферах деятельности человека.

Так как живём в мире технологий, огромное количество информации хранится именно в электронном виде. Тяжело представить современный мир без использования различных интернет-источников. Соответственно необходимо создавать высококачественные сайты.

Целью курсовой работы является применение на практике знаний, полученных в процессе изучения курса «Современные технологии интернет-программирования».

Задачами курсовой работы являются: формирование понимания назначения информационных порталов, умения производить обоснованный выбор программного обеспечения для разработки портала, разрабатывать структуру веб-ресурса, дизайн-макет веб-ресурса, разрабатывать модули веб-ресурса.

Для достижения цели курсовой работы необходимо реализовать информационный портал для предметной области «Интернет-магазин».

1 ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

* 1. Классификация веб-ресурсов

На данный момент на просторах Интернета работает огромное количество информационных ресурсов. Так как сайтов огромное количество, необходимо их разделять между собой. Всё разнообразие сайтов можно классифицировать следующим образом:

1. По технике создания: HTML, CMS, конструктор.
2. По характеру решаемых задач: персональные, социальные, коммерческие.
3. По типу представления информации: открытые, с регистрацией, закрытые.
4. По структуре и содержанию: посадочные страницы, интернет-магазины, блоги.
5. По особенностям контента: справочники, видеохостинги, каталоги.

Ниже подробнее рассмотрим каждую категорию информационных порталов.

HTML – страницы, написанные на этом языке, также называют статическими. Исторически они были первыми, и каждая страница верстается вручную. Несмотря на то, что они являются устаревшими, они все еще частично остаются востребованными благодаря минимальной нагрузки на сервер, возможность использования CSS стилей и javascript, помимо этого сайты написанные данным способом совместимы с любым хостингом. Минус заключается в том, что все изменения в меню и других элементах и ​​функциях нельзя производить автоматически. Поэтому HTML подходит для одностраничных проектов, но не для больших проектов.

CMS – в этом случае веб-ресурсы называются динамическими, потому что их страницы состоят из блоков — фрагментов кода, которые используются повторно. Эта итерация позволяет быстро создавать большое количество разделов, не перегружая их тегами и при необходимости вносить изменения только в нужные файлы. Операции выполняются через движок — систему управления контентом.

Конструктор – это платформа (Ucoz, Tilda) с набором готовых шаблонов и проверенных инструментов, которые позволяют даже неспециалисту самостоятельно открыть блог, визитку или другой простой проект. Данный способ создания сайтов является весьма удобным, так как программное обеспечение уже вшито и отлажено, а хостинг подобран.

Интернет-магазины – данные информационные ресурсы продают товары с кратким описанием и, что немаловажно, с доставкой клиенту по городу и домой. Они доступны 24 часа в сутки, известны своим широким охватом и подходят своим владельцам благодаря отсутствию ограничений на размер интерпретации и возможности свободно добавлять/удалять категории или ориентироваться на узкую нишу. Чаще всего каждый зарегистрированный клиент получает личный кабинет с историей заказов, скидками и бонусами.

Landing Page – цель данных ресурсов заключается в том, чтобы затянуть пользователя, вызвать у него интерес, заставить его совершить определенное действие, допустим, пройти курс или приобрести какую-то вещь. На это работают все элементы их логической структуры: кликабельная шапка (заголовок), выгодное и сразу заметное торговое предложение, преимущества клиента, призывная кнопка действия, обратная связь.

Визитки – проекты, состоящие из небольшого количества страниц (от 3 до 7), которые представляют целевую аудиторию компании, отвечают на вопрос: «Кто мы?» и предоставляют людям контакты, прайс-листы, возможности обратной связи. Они быстро разрабатываются (на основе HTML) и представляют компанию во всемирной паутине.

Персональные страницы – это сайты начинающих разработчиков, которые создают один, два и более сайтов в Интернете на одну страницу. Обычно они содержат какую-либо информацию, например: о себе, о друзьях, увлечениях.

Информационные страницы – это один из самых распространенных типов сайтов в Интернете. Предприятия, организации, частные предприниматели или обычные пользователи создают их для предоставления информации широкой аудитории.

Промо-сайты – сайты, посвященные разовым акциям и акциям. Выставка или конференция, праздник или концерт, презентация нового продукта. Освещение подобных мероприятий через специальные ресурсы, называемые «промо-сайтами», является хорошим тоном и показателем высокого уровня их организации и проведения. Такие сайты отличаются очень красивым дизайном. Часто используется сложная анимация. Они как телевизионная реклама. Именно своим дизайном сайт должен удивлять и привлекать посетителей.

Корпоративный сайт создается компанией или организацией. Он предоставляет посетителям максимально полную информацию о продаваемых товарах или услугах и схемах взаимодействия с потенциальными клиентами. Этот сайт позволяет автоматизировать документооборот, обеспечить информационное взаимодействие с удаленными филиалами и расширить рынок сбыта.

Сервисы – сервисные страницы позволяют пользователям обрабатывать различные виды информации: переводить тексты, обрабатывать рисунки и фотографии, анализировать страницы и так далее. Такие сайты могут заменить ряд программ, которые можно установить на персональный компьютер. Существуют крупные порталы, предоставляющие ряд различных услуг.

Социальные сайты – социальные сети созданы для общения пользователей друг с другом. Пользователи ведут свои страницы – выкладывают фото, видео, статьи, мысли. Сайты социальных сетей имеют сходство с форумами, блогами и сайтами-визитками. Самые распространенные примеры соцсетей: Одноклассники, Вконтакте, Instagram, Twitter, Facebook.

Новостные сайты – функциональность информационного сайта предполагает возможность размещения и редактирования новостей в удобном формате. Эти типы сайтов размещают актуальные новости, ведется постоянная ежедневная работа по обновлению и заполнению информации. С достаточным уровнем трафика, можно размещать на сайте города платные объявления предпринимателей. Трафик на этих сайтах можно приобрести в статьях о достопримечательностях города, справочных адресах, фотографиях. Главный недостаток таких сайтов – высокая конкуренция.

На данном этапе курсовой работы был проведен анализ классификации информационных ресурсов.

* 1. Анализ целевой аудитории информационного портала

Анализ целевой аудитории необходим на всех этапах разработки интернет-проекта. Прежде чем начать, нужно изучить нишу и понять потребности пользователей, а затем обратиться к инструментам веб-аналитики, чтобы корректировать работу.

Важные критерии при изучении целевой аудитории:

1. Заинтересованность в продукте или услуге. Поклонникам здорового образа жизни и спорта бессмысленно пытаться продать домашние торты или пироги.
2. Финансовая способность к покупке. Дорогие автомобили студенты уж точно покупать не будут.
3. Восприимчивость к маркетинговому давлению. Если человек изначально негативно настроен по отношению к бренду, на него не подействует ни один маркетинговый приём.

Детальное изучение своей целевой аудитории позволяет: построить успешную кампанию во всех маркетинговых каналах, привлечь новых покупателей за счет ориентированных на аудиторию акций, скидок, конкурсов, скорректировать ассортимент согласно потребностям потребителей, ускорить развитие проекта.

Для анализа целевой аудитории применяют несколько взаимодополняющих методов:

1. Анализ аудитории сайта и соцсетей.
2. Анализ сезонности и трендов.
3. Анализ спроса с помощью подбора поисковых запросов.
4. Опрос пользователей на сайте или в соцсетях.
5. Анализ упоминаний бренда.
6. Онлайн-исследования с помощью сервисов.
7. Структурирование и визуализация данных.

Используя этот анализ для разработки структуры веб-ресурса, можно учесть стоимость рекламы на просторах Интернета, изменения информации о веб-ресурсе, размер и сложность разрабатываемого сайта, а так же примерное расположение веб-ресурса.

Очевидно, зная предпочтения своих потенциальных клиентов, легче находить с ними общий язык и предлагать актуальные для них решения.

* 1. Техническое задание

Техническое задание (ТЗ) – это завершающий предпроектную стадию документ, который содержит цели и обоснование проектирования, а также определяет основные требования к ИС и исходные данные необходимые при разработке.

Техническое задание составляется по результатам предпроектного обследования объекта автоматизации. Является неотъемлемым инструментом коммуникации между заказчиком и исполнителем.

ТЗ содержит совокупность требований к ПС и может использоваться как критерий проверки и приемки разработанной программы.

В составлении ТЗ участвует не только разработчик, но и заказчик. Задача разработчика в создании технического задания заключается в том, что он должен сформулировать конечную цель и понять пожелания заказчика. Другими словами, исполнитель должен посмотреть на проблему глазами заказчика.

На основании собственноручно утверждённого технического задания, заказчик будет принимать готовую работу. В некоторых случаях ТЗ составляет исполнитель проекта. Техническое задание является одним из основополагающих документов проекта ПС. Качественно составленное техническое задание гарантирует точность выполнения работы.

Техническое задание приведено в приложении А.

2 ВЫБОР ИНСТРУМЕНТАЛЬНЫХ И ПРОГРАММНЫХ СРЕДСТВ РАЗРАБОТКИ ИНФОРМАЦИОННОГО ПОРТАЛА

2.1 Выбор языка и технологии программирования

Существуют несколько языков программирования для создания информационных ресурсов. Ниже представлены языки программирования, которые чаще всего используются при написании веб-ресурсов.

1. Java, разработанная в 1990-х годах и по-прежнему наиболее востребованная, является золотым стандартом в области веб-разработки во всем мире, во всех областях. Она ориентирована на объекты и работает на любой платформе, что делает ее чрезвычайно универсальной. Java представляет собой объектно-ориентированный язык, с помощью которого создаются модули, не интегрированные в код страницы. Эти модули используются, в основном, для разработки серверной части сайта. Разработка на нем очень долгая и дорогая. Его используют в основном для больших проектов со специфическими требованиями.
2. Python – современный язык, разработка на котором проходит довольно быстро и качественно. Используют его для средних и больших проектов. Python чрезвычайно прост в освоении и является динамичным, универсальным языком. Хотя он более популярен как язык на заднем плане, с ним можно делать практически все, что вы хотите.
3. PHP - это скриптовый язык, используемый для быстрого создания динамических веб-страниц. Отличный выбор как для frontend, так и для backend разработчиков. PHP позволяет быстро и легко расширять веб-приложения и запускать веб-сайты с повторяющимися серверными задачами (например, обновлять новостные ленты). Он имеет открытый исходный код и очень популярен в среде начинающих компаний, медиа-агентств и электронной коммерции - таких людей, которые часто нанимают новых веб-разработчиков. Язык PHP используют в основном для простых и средних проектов. Имеется огромное количество библиотек и расширений языка. С выходом последней версии языка, получил действительно мощные возможности.
4. JavaScript - язык интерфейса, используемый для создания и разработки веб-сайтов, настольных приложений и игр. JavaScript работает во всех браузерах и может работать с программами, которые не размещены в Интернете. Он поддерживает как функциональные, так и объектно-ориентированные стили программирования. Компоненты, структуры данных и алгоритмы JavaScript применяются практически к любому другому языку.
5. CSS / HTML - JavaScript с CSS и HTML вместе составляют главную троицу разработки веб-интерфейса. HTML (Hyper Text Mark Up Language) - это язык веб-браузеров - с помощью которых сделаны сайты. С помощью CSS (каскадные таблицы стилей) можно оформить дизайн веб-ресурса и заставить его выглядеть стильно.

Информационный ресурс интернет-магазина разработан на языке Java, исходя из следующих аргументов:

1. Java – высокоуровневый язык, поэтому его легко использовать для чтения, а также для написания и обслуживания.
2. Благодаря разнообразию библиотек разработчики могут реализовать широкий спектр функциональных возможностей.
3. Независимость от платформ позволяет писать код один раз, потом применяя его везде. Java обладает кросс-платформенными возможностями, позволяющими компилировать байт-коды, чтобы запустить приложение на любой другой платформе, работающей с JWM.
4. Поддерживает распределенное программирование, обеспечивающее комфортную работу в удаленном режиме на нескольких компьютерах для повышения производительности и эффективной работы сотрудников.
5. Функция автоматического управления памятью способствует эффективной работе процессора, производительность которого не снижается под воздействием программного обеспечения для очистки памяти.
6. Платформа поддерживает многопоточность. При одновременном запуске потоков они используют одни и те же области памяти. Вы можете быстро переключаться между потоками, функционирование которых не зависит друг от друга, что особенно важно в анимационных приложениях.
7. Одним из требований курсового задания является использование сервлетов, для написания которых используется язык программирования Java. Сервлет - это программный модуль Java на стороне сервера, который обрабатывает клиентские запросы. Сервлеты могут отвечать на любые запросы и обычно используются для расширения приложений, размещенных на веб-серверах. Весь жизненный цикл сервлета контролируется контейнером сервлета, используемым javax.servlet.
8. Java позволяет использовать JSP или Java Server Pages — это технология Java, которая позволяет создавать динамические веб-страницы для Java приложений.

Таким образом, в курсовой работе информационный портал интернет-магазина будет реализован на языке программирования Java.

2.2 Выбор системы управления базами данных

База данных (БД) – информационная модель, которая позволяет хранить структурированную совокупность данных о группе объектов, обладающих одинаковым набором свойств. Другими словами, база данных, как правило, моделирует некоторую предметную область или ее фрагмент.

Удобство хранения и доступа к данным, степень структурирования и целостности информации – основные требования к базам данных.

Система управления базами данных (СУБД) – это программное обеспечение для работы с базами данных.

СУБД позволяет пользователю осуществлять следующие функции:

1. создавать структуру базы данных;
2. обеспечивать це­ло­ст­ность дан­ных;
3. обес­пе­чивать безо­пас­ность, на­дёж­ность хра­не­ния;
4. пре­дос­тав­лять сред­ст­ва для ад­ми­ни­ст­ри­ро­ва­ния БД.
5. ма­ни­пу­ли­ро­вать дан­ны­ми (создание, об­нов­ление, уда­ление и изменение записей).

СУБД поддерживаются независимыми группами разработчиков и распространяются в виде исходных текстов программных модулей. Наиболее популярными СУБД являются SQL Server, MySQL и SQLite:

1. SQL Server - это программа, предназначенная для хранения базы данных и предоставления доступа к этим данным из других программ. SQL Server включает в себя поддержку XML и HTTP, инструменты для повышения производительности и доступности, которые позволяют распределять нагрузку и обеспечивать бесперебойную работу. Для надежного хранения данных используется комплексный подход к обработке данных. SQL server позволяет создавать резервные копии в любое время в течение рабочего дня, не отключая пользователей.
2. MySQL - это наиболее распространенный полноценный сервер СУБД. MySQL - это свободно распространяемая СУБД, которая успешно работает с различными веб-сайтами и веб-приложениями. Несмотря на то, что MySQL не реализует все функции SQL, она предлагает довольно много инструментов для разработки приложений.
3. SQLite – легко встраиваемая в приложение база данных. Так как это система базируется на файлах, то она предоставляет достаточно большой набор инструментов для работы с ней, по сравнению с сетевыми СУБД. При работе с этой СУБД обращения происходят напрямую к файлам (в эти файлах хранятся данные), вместо портов и сокетов в сетевых СУБД. Именно поэтому SQLite очень быстрая, а также мощная благодаря технологиям обслуживающих библиотек.

В данной курсовой работе была выбрана СУБД MySQL, так как помимо универсальности и распространенности СУБД MySQL обладает целым комплексом важных преимуществ перед другими системами. В частности следует отметить такие качества как:

1. Простота в использовании.
2. Обширный функционал.
3. Безопасность.
4. Масштабируемость.
5. Скорость.

Главное достоинство СУБД MySQL - бесплатная лицензия для работы с открытым кодом, а также наличие коммерческих версий с расширенным функционалом.

2.3 Выбор веб-сервера

Понятие «веб-сервер» можно отнести и к аппаратной начинке и к программному обеспечению, а также можно веб-сервер можно рассматривать как обе эти части работающих вместе. С одной стороны веб-сервер хранит файлы сайта, такие как HTML-документы, CSS-стили, Java файлы, а с другой стороны веб-сервер включает в себя несколько компонентов, контролирующие доступ пользователей к размещенным файлам на сервере.

В реализации данной курсовой работы был использован сервер Tomcat. Tomcat – контейнер сервлетов с открытым кодом. Данный сервер реализует спецификацию сервлетов и спецификацию Java Server Pages (JSP). Данный сервер используется при написании программ на языке программирования Java. Данный сервер позволяет запускать веб-приложения, написанные на Java и содержит ряд программ для самоконфигурирования. Стоит заметить, что Tomcat используется в качестве самостоятельного веб-сервера.

Информационный портал рассчитан на взаимодействие клиента с контентом. Если поступает запрос от клиента, то он обрабатывается и пользователя отправляется ответ на его запрос. Именно для этого и используется веб-сервер Tomcat. Фактически Tomcat представляет собой Java-приложение, которое отвечает за открытие порта для взаимодействия с пользователем, помимо этого Tomcat отвечает за настройку сессий, количество запросов от пользователя и за многие другие операции, производящихся на информационном портале. Для корректной работы у Tomcat есть компоненты, которые отвечают за определённые функции.

Для разработчиков информационных порталов контейнеры сервлетов играют огромную роль, так как с помощью них решается огромный спектр задач. Для программирования на языке программирования Java Tomcat имеет следующие преимущества:

1. Tomcat выполняет роль полноценного веб-сервера, именно поэтому развертывание приложений происходит гораздо быстрее. Помимо этого, Tomcat выступает в роли классического контейнера сервлетов, которые можно встроить в более крупный веб-сервер.
2. Tomcat позволяет реализовать обмен данными между пользователем и сервлетом. Берет на себя функции авторизации пользователей и создаёт сессии для каждого из них.
3. Tomcat предоставляет удобное пользование Java Server Pages (JSP).

3 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОГО ПОРТАЛА

3.1 Разработка структуры информационного портала

Интернет магазин фитнес добавок состоит из следующих страниц: магазин, главная, поиск, корзина, страница авторизации и регистрации пользователя.

Структура веб-ресурса представляет собой схему, которая содержит перечень всех разделов, страниц, подразделов, функциональных блоков, которые используются в качестве отдельных элементов, а также структура веб-ресурса содержит взаимосвязи между ранее упомянутых элементов ресурса. Другими словами, структура веб-ресурса – то, как сайт организован и как связаны между собой его страницы.

Структуры веб-ресурсов можно разделить на четыре следующих основных вида типы:

1. Линейная структура. Данная структура подразумевает собой наличие главной страницы и нескольких внутренних страниц. Используется для простых сайтов, таких как: портфолио, презентационных ресурсов и визиток. Данная структура представляется в виде цепочки.
2. Линейно-разветвленная. Данная структура является разновидностью линейной структуры. Основным отличием от линейной структуры является то, что линейно-разветвленная структура веб-ресурса представляет собой цепочку переходов, иногда с правом выбора. Подойдет также для небольших сайтов — в основном визиток.
3. Древовидная. Данная структура подразумевает наличие трёх и более уровней вложения, каждая ветка в такой структуре может иметь различную глубину вложенности. Для каждого направления существует своя ветка, а для каждого товара или услуги – отдельное ответвление. Данная структура универсальная, самая распространенная и оптимальная для многостраничных сайтов.
4. Сетевая. Данная структура представлена в виде сети, создаётся по принципу древовидной. Главное отличие сетевой структуры от древовидной заключается в том, что все проектируется так, чтобы посетитель мог перескочить со страницы глубокого уровня вложенности, на любой другой. Является сложной в реализации, так как можно легло упустить какие-либо детали и тем самым пользователям будет неудобно пользоваться данным веб-ресурсом.

Интернет-магазин, разрабатываемый в ходе курсовой, работы имеет древовидную структуру. Данная схема изображена на рисунке 1.

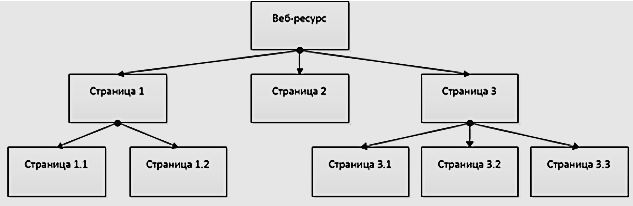


Рисунок 1 – Древовидная структура разрабатываемого веб-ресурса

3.2 Разработка дизайн-макета информационного портала

Для более удобной реализации информационного портала принято использовать макет. Анализируя макет, можно сделать выводы о том, как должны быть размещены блоки с контентом на веб-ресурсе, где должны находиться изображения и так далее. Макет для разрабатываемого информационного портала представлен на рисунке 2.

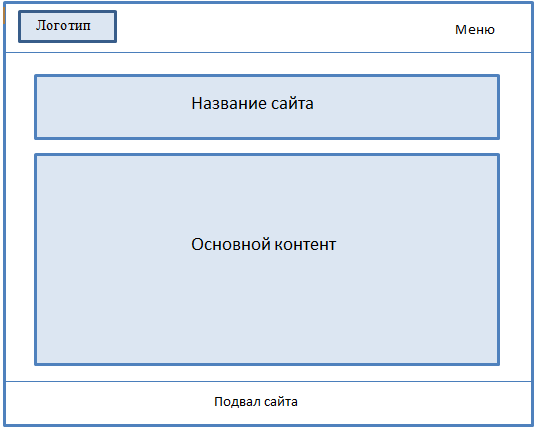


Рисунок 2 – Макет информационного портала

Анализируя макет, видно, что каждая страница сайта имеет три основных компонента: шапка, в которой располагается меню сайта, основной контент и подвал (низ страницы).

Шапка и подвал расположены соответственно сверху и снизу страницы, вытянуты по всей ширине. В правой части шапки располагается меню, которое включает в себя переходы на разные страницы и кнопку входа на сайт, в правой части шапки размещается название магазина, которое служит логотипом. В подвале сайта содержится раздел «Доска объявлений».

Основной контент главной страницы информационного портала содержит информацию и изображения товаров. Каждый товар, представленный на главной странице сайта, имеет кнопку «Подробнее», при нажатии на которую открывается страница с подробным описанием товара, также имеется возможность выбрать определённое количество товара и «положить» понравившейся товар в корзину.

Страница «Корзина» даёт пользователям возможность выбрать необходимое количество товара и заказать его.

После того как пользователь выбрал интересующий его товар, его перенаправляет на страницу авторизации, если пользователь не зашел на сайт, а также там размещена кнопка «Зарегистрироваться», если нет учётной записи.

Один из важных элементов при авторизации и регистрации является капча, которая располагается на этих страницах.

Выбор правильной цветовой гаммы для веб-ресурса так же важен, как и качество контента, размещенного на сайте. Для дизайна разрабатываемого информационного портала была выбрана серо-красная цветовая гамма.

Серый цвет не напрягает глаз пользователя, а красные элементы наоборот привлекают внимание к важным деталям на сайте.

Таким образом, посетители сайта смогут расслабиться при поиске необходимого товара, а красный цвет не даст упустить важную информацию.

3.3 Разработка модулей информационного портала

Главное преимущество интернет магазина – это, в первую очередь, возможность просмотра и выбора необходимо товара, так как это можно сделать даже дома, совсем не обязательно идти для этого в магазин.

Для осуществления заказа понравившегося товара необходима регистрация пользователя на данном сайте. На разрабатываемом информационном портале организована регистрация и авторизация пользователей с помощью кнопок «Войти» и «Зарегистрироваться», расположенных в правой части шапки страницы сайта. Страницы авторизации и регистрации представлены на рисунках 3 и 4.

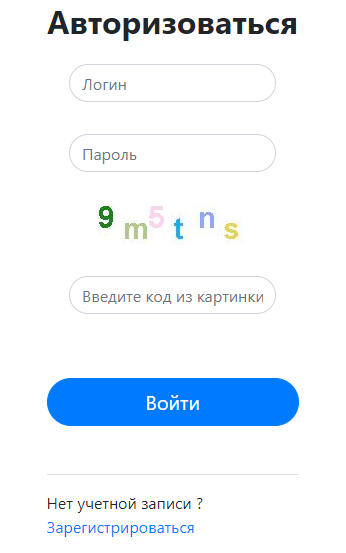


Рисунок 3 – Страница авторизации пользователя

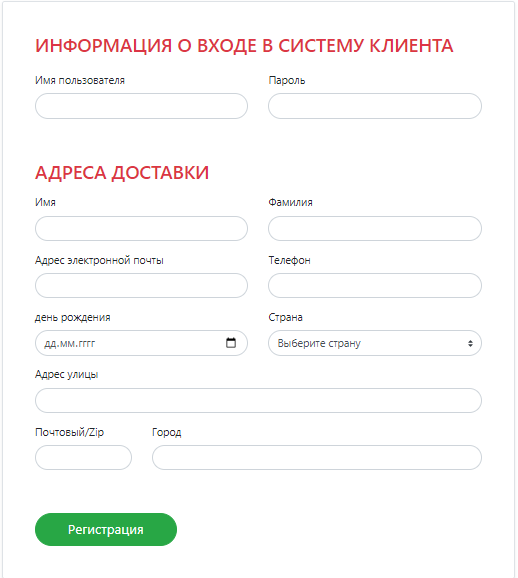


Рисунок 4 – Страница регистрации пользователя

Сервлет, используемый для регистрации представлен ниже:

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

// validate request parameters and get errors messages

Map<String, String> errors = Validation.validateSignUpForm(req);

if(errors.size() == 0) { // all parameters are valid

// get DB connection from servlet context

Connection connection = (Connection)getServletContext().getAttribute("connection");

// check weather the username is still available

String username = req.getParameter(Fields.USERNAME.toString());

UserDao userDao = new UserDao();

if(userDao.isUsernameAvailable(connection, username)) {

// get parameters request

String password = req.getParameter(Fields.PASSWORD.toString());

String firstName = req.getParameter(Fields.FIRSTNAME.toString());

String lastName = req.getParameter(Fields.LASTNAME.toString());

String email = req.getParameter(Fields.EMAIL.toString());

String phone = req.getParameter(Fields.PHONE.toString());

String birthdate = req.getParameter(Fields.BIRTHDATE.toString());

String country = req.getParameter(Fields.COUNTRY.toString());

String street = req.getParameter(Fields.STREET.toString());

String zipCode = req.getParameter(Fields.ZIPCODE.toString());

String city = req.getParameter(Fields.CITY.toString());

// create beans

User user = new User(username, password);

Customer customer = new Customer(firstName, lastName, email, phone, Utils.formatDate(birthdate));

Address address = new Address(street, zipCode, city, country, Address.Status.PRINCIPALE.toString());

// register new customer credentials

user = userDao.registerUser(connection, user);

System.out.println("Register username successfully!");

// register new customer billing informations

CustomerDao customerDao = new CustomerDao();

customer = customerDao.registerCustomer(connection, customer, user.getId());

// register new customer address

AddressDao addressDao = new AddressDao();

address = addressDao.registerAddress(connection, address, customer.getId());

// verify that user has checkout

HttpSession session = req.getSession();

String checkout = (String)session.getAttribute("checkout");

if(checkout != null) {

System.out.println("Checkout has been visted in register");

// then send user to checkout his cart

session.setAttribute("username", user.getUsername());

req.getRequestDispatcher("checkout").forward(req, resp);

return;

}

// redirect the user to the login.jsp to sign-in with success message

req.setAttribute("isRegistered", true);

req.getRequestDispatcher("jsp/login.jsp").forward(req, resp);

return;

}else {

errors.put(Fields.USERNAME.toString(), "Username: " + username + " is not available!");

}

}

req.setAttribute("errors", errors);

req.getRequestDispatcher("jsp/register.jsp").forward(req, resp);

}

}

Большую часть сайта занимает основной контент – список товаров, представленный на рисунке 5.

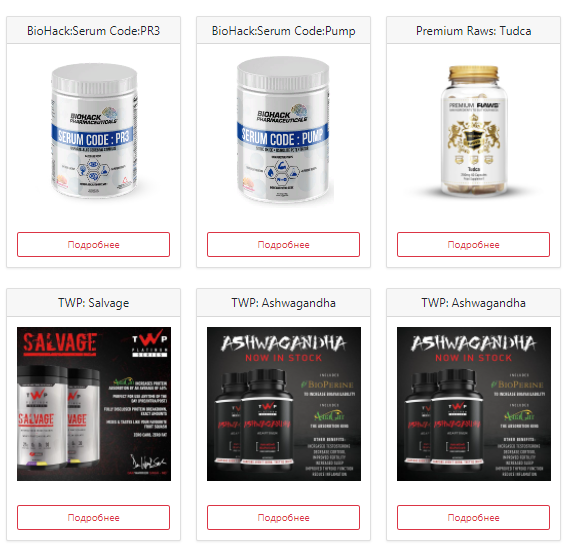


Рисунок 5 – Каталог товаров

Ниже представлен скрипт реализации данного раздела (для примера вставлен код одного товара):

<div class="card mb-4 shadow-sm" style="max-width:350px;">

<div class="card-header">

<h4 class="my-0 font-weight-normal">BioHack:Serum Code:Pump</h4>

</div>

<div class="card-body">

<img class="img-fluid" alt="BioHack:Serum Code:Pump" src="image/biohack-serum-code-pumk.png" style="max-width:308px; max-height:308px;">

<form action="product" id="productForm" method="get">

<input type="hidden" name="id" id="productId" value="2"/>

<button class="btn btn-lg btn-block btn-outline-danger mt-5">Подробнее</button>

</form>

</div>

</div>

Каждый товар имеет кнопку «подробнее», после нажатия, на которую посетителя перекидывает на страницу, на которой описан товар. Также на этой странице имеется возможность добавить интересующий товар в корзину. На рисунке 6 представлена подробная информация о товаре.



Рисунок 6 – Информация о товаре

Ниже представлен скрипт реализации данного раздела:

<div class="my-5">

<a class="float-left text-danger py-5" href="shop">Посмотреть все продукты</a>

</div>

<div class="row">

<div class="col-6">

<img class="img-fluid" alt="BioHack:Serum Code:Pump" src="image/biohack-serum-code-pumk.png">

</div>

<div class="col-6">

<h2 class="blog-post-title text-uppercase mb-3">BioHack:Serum Code:Pump</h2>

<h3 class="blog-post-title text-danger mb-3">USD

39.99</h3>

<form id="itemDescriptionForm" action="cart" method="post" onsubmit="return validateProductQuantityForm()">

<div>

<label class="text-uppercase" for="quantity">Количество</label>

<input type="text" class="form-control bg-light" id="quantity" name="quantity" style="max-width:180px;">

<span class="text-danger font-weight-bold" id="quantityError"></span>

<input type="hidden" name="id" value="2"/>

<input type="hidden" name="name" value="BioHack:Serum Code:Pump"/>

<input type="hidden" name="price" value="39.99"/>

<input type="hidden" name="description" value="Что такое BioHack Pharmaceuticals Serum Code насос?

Сывороточный кодовый насос-это не насос, предварительно работающий насосом, так же высоко дозированный, как и стимулятор BioHack PR3.Основная цель добавки - улучшить мышечные насосы во время тренировки и помочь доставке питательных веществ в ваши мышцы, чтобы дать вам этот дополнительный толчок, который вам действительно нужен."/>

<input type="hidden" name="imgPath" value="biohack-serum-code-pumk.png"/>

</div>

<button class="btn btn-danger btn-md text-white text-uppercase my-5 p-2">Далее</button>

</form>

Сервлет использованный для данного этапа проектирования:

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

// get product id from request

String id = req.getParameter("id");

System.out.println("Id to fetch: " + id);

// get connection to database

Connection connection = (Connection)getServletContext().getAttribute("connection");

// get the "id" product

ProductDao productDao = new ProductDao();

try {

// check id validity

Product product = productDao.findById(connection, Integer.parseInt(id));

if(product == null) { // the id does not exist

req.getRequestDispatcher("").forward(req, resp);

return;

}

// pass the control to product.jsp

req.setAttribute("product", product);

req.getRequestDispatcher("jsp/product.jsp").forward(req, resp);

}catch (NumberFormatException e) { // the id is not valid

req.getRequestDispatcher("").forward(req, resp);

return;

}

}

После того как пользователь выбрал товар, его необходимо положить в корзину. На странице «Корзина» пользователь может выбрать необходимое количество товара перед его заказом. На рисунке 7 представлена страница «Корзина».

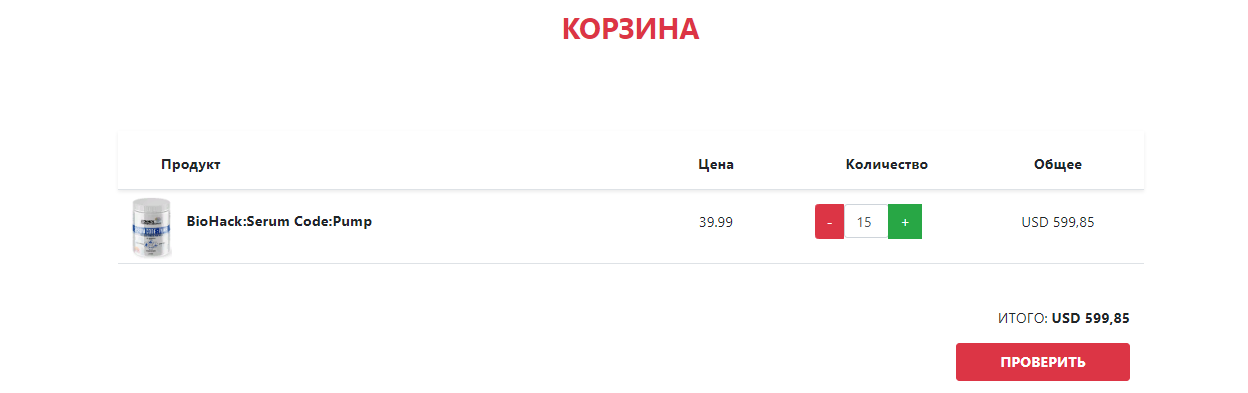


Рисунок 7 – Страница «Корзина»

Ниже представлен скрипт реализации данной страницы:

<section>

<div class = "container">

<h2 class="text-center text-danger text-uppercase my-5 py-5 font-weight-bold">корзина</h2>

<div class = "row border-bottom shadow-sm pt-4 pb-0 mb-2 font-weight-bold">

<div class="col-md-6 pl-5">

<p>Продукт</p>

</div>

<div class="col-md-2">

<p class="text-center">Цена</p>

</div>

<div class="col-md-2">

<p class="text-center"> Количество</p>

</div>

<div class="col-md-2">

<p class="text-center">Общее</p>

</div>

</div>

<div id ="shoppingCart" class="mb-5">

<div class = "row border-bottom mb-1 pb-1">

<div class="col-md-6">

<div class="d-flex">

<img class="img-fluid" alt="BioHack:Serum Code:Pump" src="image/biohack-serum-code-pumk.png" style="max-width:68.8px; max-height:68.8px;">

<div class = "d-flex align-items-center ml-3">

<p class="text-center font-weight-bold">BioHack:Serum Code:Pump</p>

</div>

</div>

</div>

<div class="col-md-2">

<div>

<p class="text-sm-center pt-3">39.99</p>

</div>

</div>

<div class="col-md-2">

<form id="form2" action="cart" method="post" class="form-inline mt-2">

<div class="btn-group" role="group">

<button type="button" id="2" class="quantityDown btn btn-danger">-</button>

<input type="text" class="form-control" name="quantity" value="15" style="max-width:50px;"/>

<input type="hidden" class="form-control" name="id" value="2" style="max-width:50px;"/>

<button type="button" id="2" class="quantityUp btn btn-success">+</button>

<input type="hidden" name="\_method" value="put" />

</div>

</form>

</div>

<div class="col-md-2">

<div>

<p class="text-sm-center pt-3">USD

599,85

</p>

</div>

</div>

</div>

</div>

<div class ="clearfix">

<div class ="float-right">

<p class="muted text-uppercase text-center"> Итого:

<span class="font-weight-bold">

USD

599,85

</span>

</p>

</div>

</div>

<div class="clearfix">

<div class = "float-right">

<a href="checkout" class="btn btn-danger text-uppercase text-white px-5 py-2 font-weight-bold">Проверить</a>

</div>

</div>

Таким образом, в курсовой работе был реализован интернет магазин фитнес добавок.

ЗАКЛЮЧЕНИЕ

Таким образом, результатом выполнения курсовой работы является разработанный «Интернет-магазин Фитнес добавки».

Помимо практической части - разработки информационного портала на языке программирования Java, было рассмотрено следующее: классификация веб-ресурсов, был проведён анализ целевой аудитории и разработано техническое задание.

В ходе выполнения курсовой работы были закреплены знания и навыки, полученные в процессе изучения дисциплины «Современные технологии интернет-программирования», получены практические навыки по реализации веб-ресурсов с помощью различных инструментальных и программных средств.

Использование разработанного интернет-магазина в первую очередь позволит пользователю значительно облегчить процесс выбора необходимого товара, так как он может, не выходя из дома, посмотреть и заказать товара, который ему нужен.

Все цели и задачи, которые поставлены на начальном этапе проектирования курсовой работы были успешно достигнуты.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Какие бывают виды веб-сайтов [Электронный ресурс] – URL: https://std17.com/blog/dlya-novichkov/kakie-byvayut-vidy-sajtov-klassifikacziya-i-primery-osnovnyh-raznovidnostej-internet-resursov/ (Дата обращения: 01.03.2022).
2. Виды, типы и разновидности сайтов [Электронный ресурс] – URL: <https://www.miit.ru/content/Приложение%204.pdf?id_wm=710843> (Дата обращения: 01.03.2022).
3. Анализ целевой аудитории интернет проекта [Электронный ресурс] URL: <https://www.uplab.ru/blog/how-to-perform-target-audience-check-list/> (Дата обращения: 02.03.2022).
4. Как составить ТЗ: подробная инструкция по созданию технического задания: [Электронный ресурс] URL: https://fb.ru/article/266231/strahovayakompaniya---eto-chto-takoe-struktura-i-funktsii. (Дата обращения 04.03.2022)
5. Плюсы и минусы сайтов на java [Электронный ресурс] URL: <https://landing.ua/blog/sajty-na-java-plyusy-i-minusy.html> (Дата обращения: 10.03.2022).
6. Система управления базами данных mysql [Электронный ресурс] URL: <https://depix.ru/articles/sistema_upravleniya_bazami_dannyh_mysql> (Дата обращения: 15.03.2022).
7. Шустова, Л. И. Базы данных: учебник / Л.И. Шустова, О.В. Тараканов. - М.: НИЦ ИНФРА-М, 2016. - 336 с.
8. Горев А., Ахаян Р., Макашарников С. Эффективная работа с СУБД. – СПб.: Питер, 2017. – 704с.
9. Tomcat в Java [Электронный ресурс] URL: <https://javarush.ru/groups/posts/tomcat-v-java> (Дата обращения: 16.03.2022).
10. Сервлеты. Введение в сервлеты [Электронный ресурс] URL: <https://metanit.com/java/javaee/4.1.php> (Дата обращения: 05.04.2022).
11. Васильев В.В. Практикум по Web-технологиям [Электронный ресурс] / В.В. Васильев, Н.В. Сороколетова, Л.В. Хливненко – М.: Форум, 2009. – 416 с. (Электронно-библиотечная система Znanium.com) (Дата обращения: 05.04.2022).
12. Хеффельфингер Д. Java EE 6 и сервер приложений GlassFish 3 [Электронный ресурс] / Д. Хеффельфингер – М.: ДМК Пресс, 2013. – 416 с. (Электронно-библиотечная система Znanium.com) (Дата обращения: 05.04.2022).

Приложение А

**УТВЕРЖДЕНО**

**Веб-ресурс «Разработка информационного портала «Интернет магазин фитнес добавок»**

**Техническое задание**

|  |  |
| --- | --- |
| **Подпись и дата** |  |
| **Инв. №дубл.** |  |
| **Взам. инв. №** |  |
| **Подпись и дата** |  |
| **Инв. № подл.** |  |

Листов 7

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

«Разработка информационного портала «Интернет магазин фитнес добавок»»

СОДЕРЖАНИЕ

|  |  |
| --- | --- |
| 1 Наименование и область применения | 39 |
| 2 Основание для разработки | 39 |
| 2.1 Организация, утвердившая документ | 39 |
| 2.2 Тема разработки | 39 |
| 3 Назначение разработки | 40 |
| 4 Технические требования | 40 |
| 4.1 Требования к функциональным характеристикам | 40 |
| 4.2 Требования к надежности | 40 |
| 4.3 Требования к эксплуатации | 41 |
| 4.4 Требования к составу и параметрам технических средств | 41 |
| 4.5 Требования к информационной и программной совместимости | 41 |
| 5 Технико-экономические характеристики | 42 |
| 6 Стадии и этапы разработки | 42 |
| 7 Порядок контроля и приемки | 42 |
|  |  |
|  |  |

1 НАИМЕНОВАНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

Веб-ресурс «Интернет магазин фитнес добавок».

Данный веб-ресурс предназначен для облегчения клиентам совершения покупок. Интернет магазин доступен пользователям 24 часа в сутки, что позволяет не упустить ни одного клиента. Главная задача данного веб-ресурса заключается в представлении клиентам интересующих товаров. Главным преимуществом веб-ресурса является то, что, будучи полноценным сайтом, содержит полную информации о представленных товарах, позволяет пользователям зарегистрироваться на сайте для совершения покупки.

2 ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ

Основанием для разработки веб-ресурса служит актуальность использования интернет-магазинов в современном мире.

Разработка ведется на основании задания курсовой работы «Разработка информационного портала организации».

Задание утверждено Балаковским инженерно-технологическим институтом филиалом НИЯУ МИФИ.

2.1 Организация, утвердившая документ

Балаковский инженерно–технологический институт – филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ», 2022 год.

2.2 Тема разработки

Разработка информационного портала «Интернет магазин фитнес товаров».

1. НАЗНАЧЕНИЕ РАЗРАБОТКИ

Информационный портал представляет собой веб-ресурс, который позволяет совершать просмотр и онлайн-покупки товаров.

1. ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К ПРОГРАММНОМУ ПРОДУКТУ

4.1 Требования к функциональным характеристикам

Веб-ресурс должен обеспечить регистрацию и авторизацию клиентов интернет-магазина, а также просмотр, добавление в корзину и заказ выбранных товаров.

Входные данные: персональные данные клиента – имя, фамилия, адрес электронной почты, телефон, дата рождения, страна, адрес.

Выходные данные: формирование заказа клиента.

* 1. Требования к надежности

Надежное (устойчивое) функционирование веб-ресурса должно быть обеспечено выполнением Заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

1. организацией бесперебойного питания технических средств;
2. организацией доступа в Интернет;
3. использованием лицензионного программного обеспечения;
4. регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;
5. регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

В состав технических средств должен входить IВМ-совместимый персональный компьютер (ПЭВМ) с минимальными системными требованиями не ниже:

1. процессор Pentium-2.0Hz, не менее;
2. оперативную память объемом, 1Гигабайт, не менее;
3. HDD, 5 Гигабайт, не менее;
4. операционную систему Windows XP/Vista/7 или старше.
   1. Условия эксплуатации

Условия эксплуатации программы совпадают с условиями эксплуатации ПЭВМ IBM PC и совместимых с ними ПК. Программа должная быть рассчитана на непрофессионального пользователя.

* 1. Требования к составу и параметрам технических средств

Необходимо наличие IBM PC – совместимого ПК с графическим

адаптером VGA или любого другого способного обеспечить вывод

изображения на экран. Необходимое дисковое пространство – не менее 100

Мб, объем свободной оперативной памяти – не менее 256 Мб.

1. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

Ориентировочная экономическая эффективность не рассчитываются. Аналогия не проводится ввиду уникальности предъявляемых требований к разработке веб-ресурса. Основным преимуществом данного программного обеспечения над аналогами является абсолютная оригинальность и неповторимость.

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание | Строки | Исполнитель | Отчёт |
| Курсовая работа | 10 недель | Коба Н.А.  Студент группы ИФСТ-31 | Демонстрация курсовой работы |

1. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание | Сроки | Исполнители | Отчёт |
| Анализ требований к проекту | C 01.03.22 до 6.03.22 | Менеджер | Документация с требованиями к проекту |
| Проектирование | C 07.03.22 до 14.03.22 | Проектировщик | Разработка структуры информационного портала |
| Реализация | C 15.03.22 до 20.04.22 | Программист | Веб-ресурс |
| Тестирование продукта | C 20.04.22 до 22.04.22 | Тестировщик | Проверенный веб-ресурс |
| Внедрение, поддержка | C 22.04.22 до 01.05.22 | Программист | Документ о завершении работ |

1. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

Приемо-сдаточные испытания веб-ресурса должны проводиться согласно разработанной Исполнителем и согласованной Заказчиком Программы и методик испытаний.

Ход проведения приемо-сдаточных испытаний Заказчик и Исполнитель документируют в Протоколе проведения испытаний.

На основании Протокола проведения испытаний Исполнитель совместно с Заказчиком подписывает Акт приемки-сдачи программы в эксплуатацию.