**Министерство образования Иркутской области**

Государственное бюджетное профессиональное

образовательное учреждение Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

|  |  |
| --- | --- |
| КР.09.02.07.22.201.10 ПЗ |  |

ИНФОРМАЦИОННАЯ СИСТЕМА «СКЛАД»

|  |  |  |
| --- | --- | --- |
| Председатель ВЦК: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Кудрявцева) |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Кудрявцева) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Д.В. Козлов) |

Иркутск 2022

**Содержание**

[Введение 3](#_Toc122649719)

[1. Описание предметной области ИС 4](#_Toc122649720)

[2. Анализ инструментов, используемых в разработке 5](#_Toc122649721)

[3. Техническое задание 14](#_Toc122649722)

[4. Проектирование ИС 15](#_Toc122649726)

[4.1. Структурная схема ИС 15](#_Toc122649727)

[4.2. Функциональная схема ИС 17](#_Toc122649728)

[4.3. Проектирование базы данных 19](#_Toc122649729)

[4.4. Проектирование интерфейса 23](#_Toc122649734)

[5. Разработка ИС 24](#_Toc122649735)

[5.1. Разработка интерфейса ИС 24](#_Toc122649736)

[5.2 Разработка базы данных 31](#_Toc122649737)

[5.3 Разработка ИС 32](#_Toc122649738)

[6. Технологическая документация ИС 36](#_Toc122649744)

[6.1. Руководство пользователя ИС 36](#_Toc122649745)

[Заключение 38](#_Toc122649746)

[Список используемых материалов 39](#_Toc122649747)

[Приложение А – Техническое задание 40](#_Toc122649748)

[3 Требования к системе в целом 42](#_Toc122649749)

[3.1. Требования к структуре и функционированию приложения 42](#_Toc122649750)

[Приложение Б – Листинг User.php 45](#_Toc122649751)

Введение

Склад – это сложное техническое сооружение, предназначенное для приемки, размещения, накопления, хранения, переработки, отпуска и доставки продукции потребителям.

Большая часть складов проектируется для хранения сырья до выполнения операций и готовой продукции до её дистрибьюции. В меньшей степени здесь хранят незавершенное производство, расходуемые материалы и запасные части.

Для более удобного и быстрого поиска товаров, необходимо создать информационную систему, с помощью которой можно будет ускорить и облегчить работу на складе.

Целью курсовой работы является создание информационной системы «Склад».

Для достижения конечного результата, а именно создание информационной системы «Склад», необходимо решить следующие задачи:

* разработать техническое задание;
* проанализировать инструментальные средства разработки;
* спроектировать базу данных;
* спроектировать информационную систему;
* спроектировать интерфейс;
* разработать базу данных;
* разработать информационную систему;
* разработать интерфейс;
* продемонстрировать программный продукт;
* создать руководство пользователя по информационной системе.

1. Описание предметной области ИС

В качестве предметной области выбран «Склад», который предназначен для осуществления процедуры хранения продукции для сторонних лиц, заинтересованных в этом. Склад должен выполнять следующие функции: прием, учет, хранение и отгрузка готовой продукции, приемка готовой продукции, рассортировка, определение потребности в транспортных средствах, механизированных погрузочных средствах, таре и рабочей силе для отгрузки продукции, согласование планов и условий поставок продукции с основного производства и по договорам со сторонними организациями, организация приемки продукции сторонними организациями, координация деятельности по закупке и продаже продукции с наличием свободных складских площадей, подготовка отчетов об объемах продукции.

В ходе работы склада, он нуждается в инвентаризации, которая включает в себя такие стадии как: ответственный работник по переучету продукции, в сопровождении книги переучета, отправляется на склад и вручную осматривает и переписывает данные о товаре и его количестве; после этого данные сверяются в книге учета товаров, лицами ответственными за документы отчетности на складе и составляется соответствующий отчет, по данным переучета продукции.

Информационная система предназначена для сотрудников, и позволяет работать с товарами – добавлять, изменять и удалять их.

1. Анализ инструментов, используемых в разработке программного продукта

Грамотно выбранные инструменты для создания программного продукта, заранее определяют его результат.

Проектировать структуру информационной системы удобно через phpMyAdmin и Draw.io, а дизайн – через сервис Figma. Информационная система будет серверной, для работников. Для реализации отлично подойдут следующие инструменты: HTML5, CSS3 и JS, включая AJAX.

phpMyAdmin — [веб-приложение](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5) с открытым кодом, написанное на языке [PHP](https://ru.wikipedia.org/wiki/PHP) и представляющее собой веб-интерфейс для администрирования СУБД MySQL. PhpMyAdmin позволяет через браузер и не только осуществлять администрирование сервера MySQL, запускать команды SQL и просматривать содержимое таблиц и баз данных. В проекте используется для создания ER-диаграммы БД

Draw.io — это бесплатное кроссплатформенное программное обеспечение для рисования графиков с открытым исходным кодом, разработанное на HTML5 и JavaScript. Его интерфейс можно использовать для создания таких диаграмм, как блок-схемы, каркасы, UML-диаграммы, организационные диаграммы и сетевые диаграммы. В проекте используется для создания прототипа страниц.

Figma – графический редактор для создания прототипов сайтов и приложений. Над проектом одновременно могут работать несколько человек, так как можно выдать доступ на редактирование или комментирование любому. В Figma обычно создают прототипы сайтов и приложений, иллюстрации, векторную графику, рисуют элементы интерфейса. Ещё здесь создают макеты сайтов для тильды: есть возможность импортировать дизайн. В проекте используется для наглядного дизайна проекта.

HTML – язык разметки гипертекста. Язык разметки дает браузеру необходимые инструкции о том, как отображать тексты и другие элементы страницы на мониторе. Язык HTML интерпретируется браузерами и отображается в виде документа, в удобной для человека форме.

CSS – каскадные таблицы стилей, которые используются для определения стилей (правил) оформления документов — включая дизайн, вёрстку и вариации макета для различных устройств и размеров экрана.

JavaScript – это мультипарадигменный язык программирования, который обычно применяется в качестве встраиваемого инструмента для программного доступа к различным объектам приложений. С точки зрения веб-разработки, без знаний этой технологии невозможно заниматься созданием современных интерактивных сайтов. Язык JS – это то, что «оживляет» разметку страниц (HTML) и пользовательский функционал (CMS) сайтов. С помощью этого языка реализуется возможность реакции страницы или отдельных ее элементов на действия посетителя. Сегодня JavaScript является базовым языком программирования для браузеров. Он полностью совместим с операционными системами Windows, Linux, Mac OS, а также всеми популярными мобильными платформами.

AJAX, или, более длинно, Asynchronous Javascript And Xml - технология для взаимодействия с сервером без перезагрузки страниц. За счет этого уменьшается время отклика и веб-приложение по интерактивности больше напоминает десктоп.

Веб-приложение будет содержать в себе информацию – её необходимо хранить, изменять, структурировать и использовать. Это реализуется благодаря базе данных. Были рассмотрены следующие варианты реализации СУБД:

1. MySQL;

2. MySQL Workbench;

3. PostgreSQL.

MySQL — одна из наиболее используемых систем управления базами данных. MySQL управляет реляционными базами данных, то есть такими, в которых таблицы связаны между собой. MySQL работает по принципу клиент-сервер. Компьютер пользователя (клиент) отправляет запрос. Сервер баз данных его обрабатывает и предоставляет ответ. Именно поэтому часто можно услышать понятие MySQL-сервер. Это сервер, на котором хранится база данных. Система MySQL написана на языках программирования C и C++. Для работы MySQL используется язык структурированных запросов SQL.

[MySQL Workbench](https://www.mysql.com/products/workbench/) — универсальный инструмент для работы и управления базами данных. С его помощью можно создавать и редактировать таблицы и другие объекты, управлять доступом пользователей и полноценно администрировать БД. Workbench может пригодиться разработчикам, администраторам БД и архитекторам систем.

Одна из главных особенностей Workbench — визуальное моделирование. Оно позволяет увидеть все таблицы и связи между ними в наглядном виде. Это намного упрощает проектирование базы данных.

PostgreSQL — это объектно-реляционная система управления базами данных (ORDBMS), наиболее развитая из открытых [СУБД](https://blog.skillfactory.ru/glossary/subd/) в мире. Имеет открытый исходный код и является альтернативой коммерческим базам данных. СУБД позволяет гибко управлять базами данных. С ее помощью можно создавать, модифицировать или удалять записи, отправлять транзакцию — набор из нескольких последовательных запросов на особом языке запросов SQL.

Для наглядности сравнения вариантов реализации базы данных была составлена таблица 1.

Таблица 1 – Сравнение средств реализации базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| Название СУБД | MySQL | MariaDB | PostgreSQL |
| Большое кол-во типов данных | + | + | + |
| Популярность | + | + | - |
| Отказоустойчивость | - | + | + |
| Не требует удаленного сервера | - | - | - |
| Простота использования | - | + | - |
| Портативность | - | + | - |

Таким образом, в качестве базы данных для будущего продукта была выбрана MariaDB, так как она предоставляет весь необходимый функционал для разработки продукта, также она имеет больше достоинств по сравнению с другими БД и при этом она проста в изучении и овладении.

Для взаимосвязи баз данных и северной части продукта необходимо использовать серверный язык. Для реализации этого были рассмотрены два языка программирования – Python и Php.

Python — это активно развивающийся скриптовый язык, который используют для решения большого объема самых разноплановых проблем и задач. Python пригодится в создании компьютерных и мобильных приложений, его применяют в работе с большим объемом информации, при разработке web-сайтов и других разнообразных проектов, используют в машинном обучении. Данный язык программирования используют крупные известные корпорации, такие как Spotify и Амазон (например, для анализа данных и создания алгоритма рекомендаций), YouTube и даже Walt Disney. Таким образом, Python нашел свое место в различных областях — с его помощью можно решить множество задач разной сложности. Синтаксис ядра языка минималистичен, за счёт чего на практике редко возникает необходимость обращаться к документации. Сам же язык известен как интерпретируемый и используется в том числе для написания скриптов. Недостатками языка являются зачастую более низкая скорость работы и более высокое потребление памяти написанных на нём программ по сравнению с аналогичным кодом, написанным на компилируемых языках, таких как C или C++. Недостатками языка являются использование языка в серверной разработке, только благодаря фреймворку, а также зачастую более низкая скорость работы и более высокое потребление памяти написанных на нём программ по сравнению с аналогичным кодом, написанным на компилируемых языках, таких как С или C++. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений, высокоуровневые структуры данных. Поддерживается разбиение программ на модули, которые, в свою очередь, могут объединяться в пакеты.

PHP — это распространённый язык программирования общего назначения с открытым исходным кодом. PHP специально сконструирован для веб-разработок и его код может внедряться непосредственно в HTML.

Язык PHP обладает рядом неоспоримых преимуществ:

* Высокая скорость работы и, соответственно, общая производительность ресурсов.
* Бюджетность, экономичность. Найти специалиста не представляется проблемой, стоимость написания программ на php не высока.
* Простота освоения, простой синтаксис.
* Отличная совместимость и переносимость — php-коды работают одинаково хорошо с разными платформами.
* Набор текста кода и его редактирование можно осуществлять в любом текстовом или html-редакторе.

Для наглядности сравнения языков программирования была составлена таблица 2.

Таблица 2 – Сравнение языков программирования для разработки программного продукта

|  |  |  |
| --- | --- | --- |
| Название языка программирования | Php | Python |
| Наличие библиотек | + | + |
| Инструменты для работы с БД | + | + |
| Объектно-ориентированные возможности | + | + |
| Лёгкий понятный синтаксис | + | - |
| Более активное сообщество | + | - |
| Более лёгкая простая модульность | + | - |

Таким образом, php будет более лучшим вариантом, ведь он более компактный и простой в освоении язык. Он вобрал все лучшие особенности таких популярных языков, как C, Java и Perl, а также его сообщество более активное.

Для разработки программного продукта рассмотрены следующие инструментальные средства разработки программных продуктов:

1. PHPStorm.

2. Netbeans.

3. Atom.

PhpStorm — коммерческая кросс-платформенная [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) обеспечивает первоклассную поддержку PHP 5.3–8.1, мгновенно предотвращает ошибки, предоставляет точное автодополнение и безопасные рефакторинги, а также возможность редактирования кода на HTML, CSS и JavaScript.

IDE обеспечивает умное автодополнение, подсветку синтаксиса, расширенные настройки форматирования, проверку ошибок на лету, сворачивание фрагментов кода, поддержку языковых комбинаций и много других возможностей для продуктивной разработки. Автоматические рефакторинги позаботятся о порядке в вашем коде и помогут легко и безопасно выполнять глобальные настройки проекта.

Сотни встроенных инспекций проверяют код прямо в режиме редактирования, а также позволяют проанализировать весь проект на наличие ошибок и проблем в структуре кода. Для большинства инспекций доступны быстрые исправления, которые позволяют откорректировать код мгновенно. Нажмите Alt+Enter, чтобы просмотреть список возможных исправлений для любой инспекции. PhpStorm помогает быстро перемещаться по коду и экономить время при работе над большими проектами. В один клик можно перейти к определению любого метода, функции или переменной, или найти их использования в проекте.

NetBeans IDE — свободная интегрированная среда разработки приложений (IDE) на языках программирования Java, Python, PHP, JavaScript, C, C++, Ада и ряда других. Проект NetBeans IDE поддерживается и спонсируется компанией Oracle, однако разработка NetBeans ведётся независимым сообществом разработчиков-энтузиастов (NetBeans Community) и компанией NetBeans Org. Последние версии NetBeans IDE поддерживают рефакторинг, профилирование, выделение синтаксических конструкций цветом, автодополнение набираемых конструкций на лету и множество предопределённых шаблонов кода.

Atom (в прошлом Atomicity) — бесплатный текстовый редактор с открытым исходным кодом для Linux, macOS, Windows с поддержкой плагинов, написанных на JavaScript, и встраиваемых под управлением Git. Большинство плагинов имеют статус свободного программного обеспечения, разрабатываются и поддерживаются сообществом.

Сравнение IDE для разработки программного продукта наглядно представлено в таблице 3.

Таблица 3 – Сравнение IDE для разработки программного продукта

|  |  |  |  |
| --- | --- | --- | --- |
| Название IDE | PHPStorm | NetBeans | Atom |
| Распространяется бесплатно | +  - | + | + |
| Автоматическое сохранение | + | - | + |
| Подсказки по коду | + | + | + |
| Интеграция с системой контроля версия (GIT) | + | + | - |
| Возможность расширения функционала библиотеками | + | + | + |
| Заточен под PHP-разработку | + | + | + |
| Поддержка CSS/HTML/JS | + | + | + |
| Комфортное использование на слабых ПК | + | - | + |

Таким образом, после рассмотрения вариантов средств разработок, было принято решение использовать PHPStorm. PHPStorm намного функциональнее, чем приведённые выше Atom и NetBeans. В PHPStorm имеется автоматическое сохранение, которое не позволит случайно потерять все наработки, а также он более удобен в работе.

1.2 Обоснование выбора

Для создания программного продукта было решено использовать

средства:

1) Для создания структурных схем, контекстной и диаграмм декомпозиции использовались CASE-средства – Draw.io.

2) Для наглядного составления структуры базы данных использовался инструмент для визуального проектирования баз данных, их редактирования и полного администрирования – MySQL Workbench.

3) Для разработки дизайна web-приложения использовался онлайн-сервис для разработки дизайна и прототипа сайта – Figma.

4) На этапе разработки программного продукта использовались инструменты и среды для разработки: PHP, Laravel, HTML5, CSS3, JavaScript и AJAX, а также редактор кода PHPStorm.

Основные преимущества веб-технологий:

1. Распределенность (пользователь может работать с системой из любого места, связанного с WEB-сервером по сети, находясь в любой точке земного шара);
2. Переносимость (Web-клиенты (браузеры) существуют для любых  
   платформ, от настольных компьютеров до сотовых телефонов. Web-сервера используются для большинства платформ, а Web-приложения обычно пишутся на переносимых языках);
3. Привычность интерфейса (почти каждый пользователь компьютера хотя бы раз запускал браузер и работал в нем);
4. Простота установки и обслуживания (новую версию web-приложения не надо устанавливать на все компьютеры – достаточно установить на сервер) ;
5. Простота интерфейса - Пользователи не любят гигантских окошек с сотнями полей ввода, а программисты и разработчики интерфейсов почему-то любят. Web как раз не поощряет сложный интерфейс, скорее поощряет простой;
6. Простота программирования интерфейса - Создавать HTML из  
   шаблонов куда проще и удобнее, чем создавать графические  
   приложения визуальными редакторами интерфейсов;
7. Простота программирования в целом - 3х-уровневая архитектура:  
   база данных – логика приложения (сервер) – логика представления  
   (клиент, браузер).
8. Техническое задание

Одним из этапов разработки курсового проекта является техническое задание, разработанное в соответствии с ГОСТ на основе стандарта ISO/IEC/IEEE 29148:2011. Техническое задание состоит из следующих пунктов:

1. Общие сведения.
2. Назначение и цели создания системы.
3. Требования к системе в целом.

3.1 Требования к структуре и функционированию системы

3.2 Требования к надежности

3.3 Требования к безопасности

3.4 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

1. Требования к документированию.
2. Состав и содержание работ по созданию системы.

Техническое задание представлено в приложении А.

4. Проектирование ИС
   1. Структурная схема ИС

Проектирование информационной системы началось с построения диаграммы вариантов использования. На рисунке 1 представлена диаграмма прецедентов Uses CASE. Она содержит 2 актёра, которые могут выполнять суммарно 9 функций, часть из которых может делать только один актёр, а часть – только другой актёр.

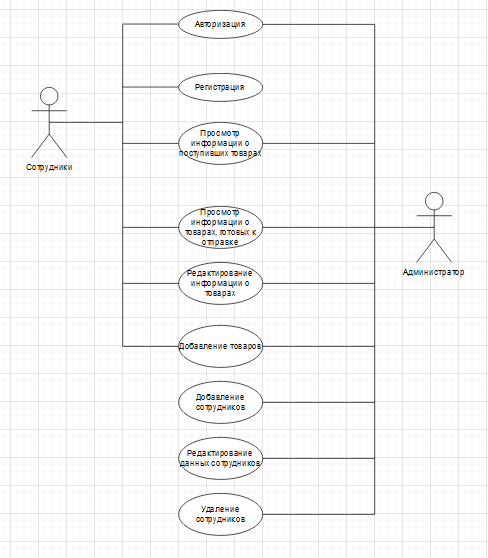


Рисунок 1 - Диаграмма прецедентов

На рисунке 2 представлена диаграмма деятельности. Она содержит 2 роли. Процесс начинается с администратора, который добавляет сотрудников, редактирует или удаляет их. Далее сотрудник или администратор добавляет товар, если есть какая-то ошибка, то изменяет или же удаляет товар.

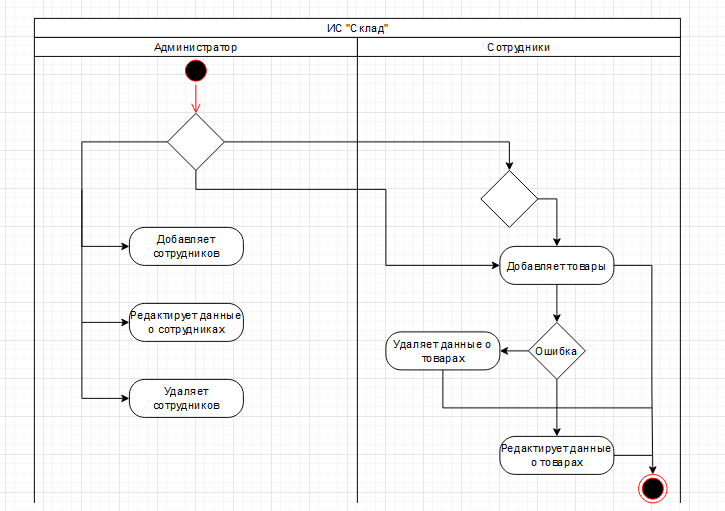


Рисунок 2 – Диаграмма деятельности

На рисунке 3 представлена диаграмма компонентов. В общей сложности 4 элементов, которые служат для просмотра, редактирования и удаления информации о прибывшем товаре, а также товаре готовом к отгрузке.

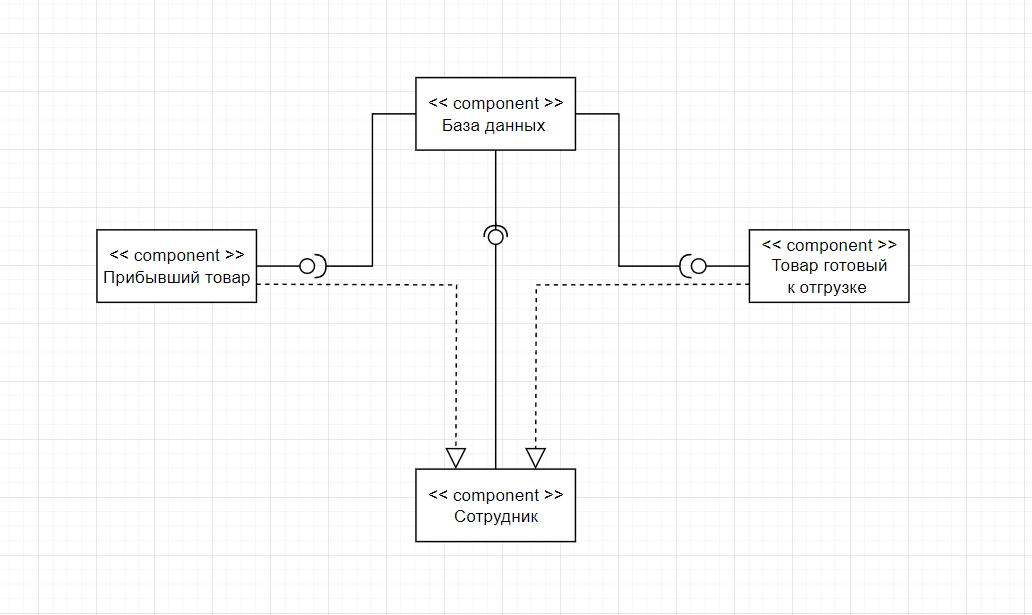


Рисунок 3 – Диаграмма компонентов

На рисунке 4 представлена диаграмма развёртывания. Она показывает, что, чтобы пользоваться программным продуктом, необходим web-сервер, на котором размещаются ИС и БД, а также для администратора и сотрудников необходим ПК с браузером и выходом в интернет.

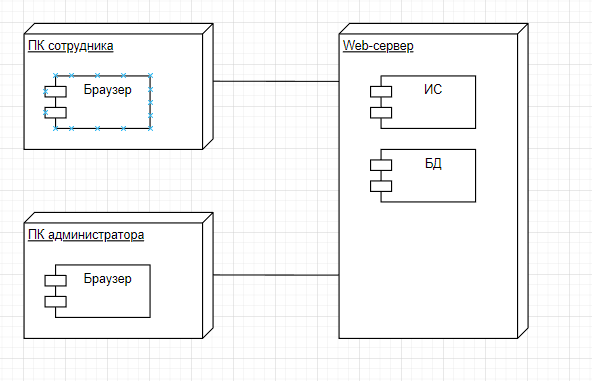


Рисунок 4 – Диаграмма развёртывания

1. 1. Функциональная схема ИС

После исследования предметной области была построена контекстная диаграмма. Она позволяет демонстрировать в целом деятельность ИС «Склад». Контекстная диаграмма – это важнейшее представление, обеспечивающее высочайший уровень понимания людей и систем, с которыми будет взаимодействовать ваша будущая система. На рисунке 5 представлена контекстная диаграмма.

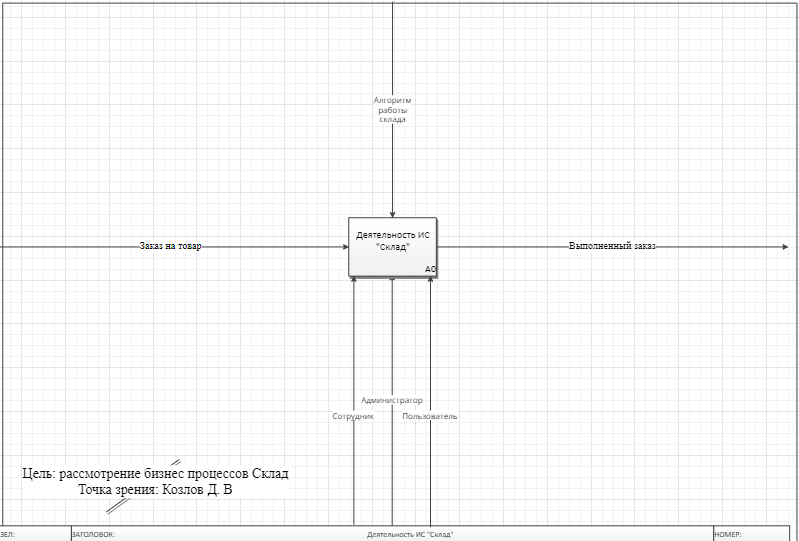


Рисунок 5 – Контекстная диаграмма IDEF0

На рисунке 6 представлена диаграмма декомпозиций (А1), отображающая деятельность ИС более подробно предыдущей.

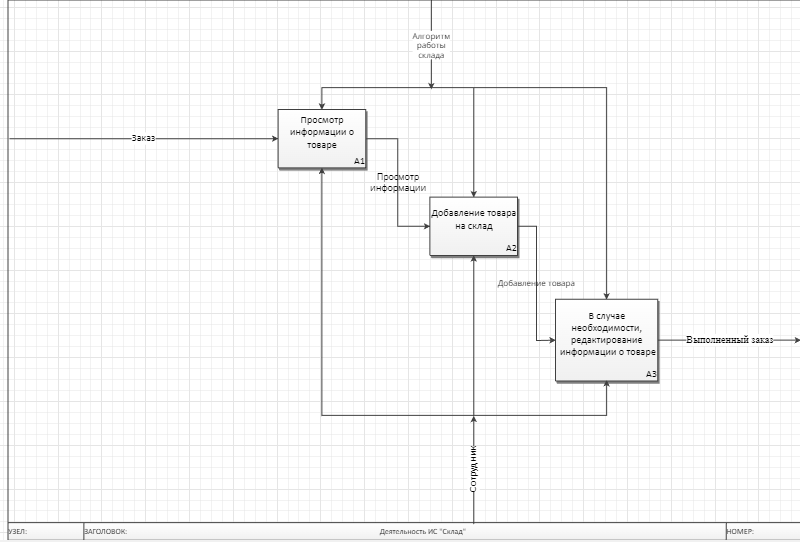


Рисунок 6 – Диаграмма декомпозиций А1

На рисунке 7 представлена диаграмма классов. Она содержит 9 классов, среди них база данных, html-страница, информационная сеть «Склад», пользователь, сотрудник, администратор, прибывший товар, товар, готовый к отгрузке и просто товар.



Рисунок 7 – Диаграмма классов

На рисунке 8 представлена диаграмма потоков данных. В центре всего сотрудник, который может взаимодействовать с товарами, для соответствующих действий в ИС предусмотрены соответствующие БД.

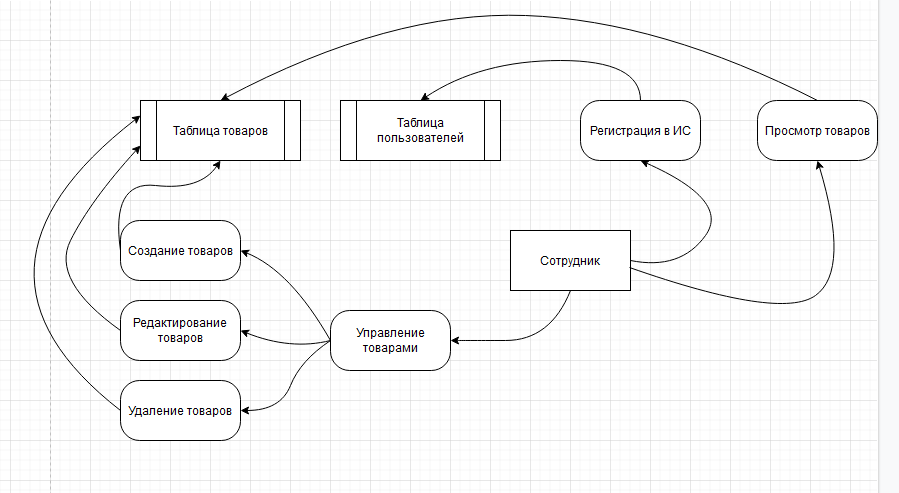


Рисунок 8 - Диаграмма потоков данных

* 1. Проектирование базы данных

Проектирование базы данных начинается с концептуального проектирование базы данных.

Концептуальное проектирование – построение семантической модели предметной области, то есть информационной модели наиболее высокого уровня абстракции. Такая модель создаётся без ориентации на какую-либо конкретную СУБД и модель данных. На рисунке 9 представлена инфологическая модель базы данных.



Рисунок 9 – Инфологическая модель базы данных

На инфологической модели базы данных выше схематично отображены сущности системы, их атрибуты и связи между ними. Так, в прямоугольник отображены сущности, такие как: Сотрудник, Товары, Администратор. В овалах отображены атрибуты сущностей, например, ФИО, E-mail, ID и другие. Ромбами изображены связи между сущностями, например, сотрудник добавляет товары.

Далее происходит преобразование концептуальной модели в логическую модель, по формальным правилам. Таким образом, логическое (даталогическое) проектирование — создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных.

На даталогической модели базы данных (рисунок 10) отображены сущности приложения, а также первичные и внешние ключи, связывающие сущности между собой.

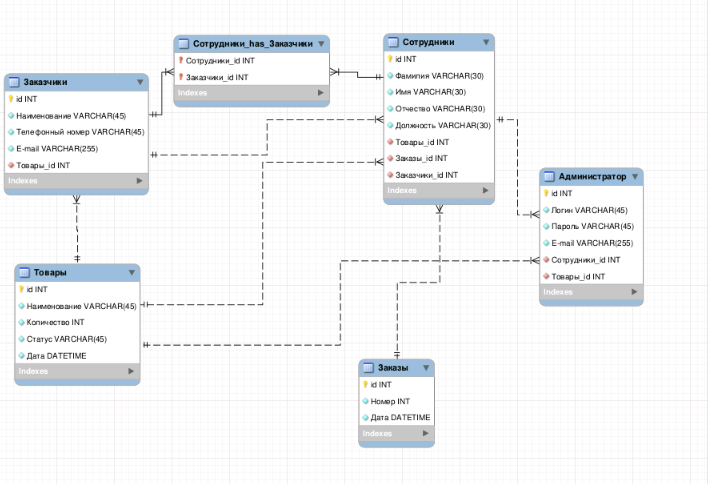


Рисунок 10 – Даталогическая модель базы данных

На рисунке 11 представлена ER-диаграмма базы данных. Она содержит 6 таблиц для полного функционирования и качественной сортировки информации.

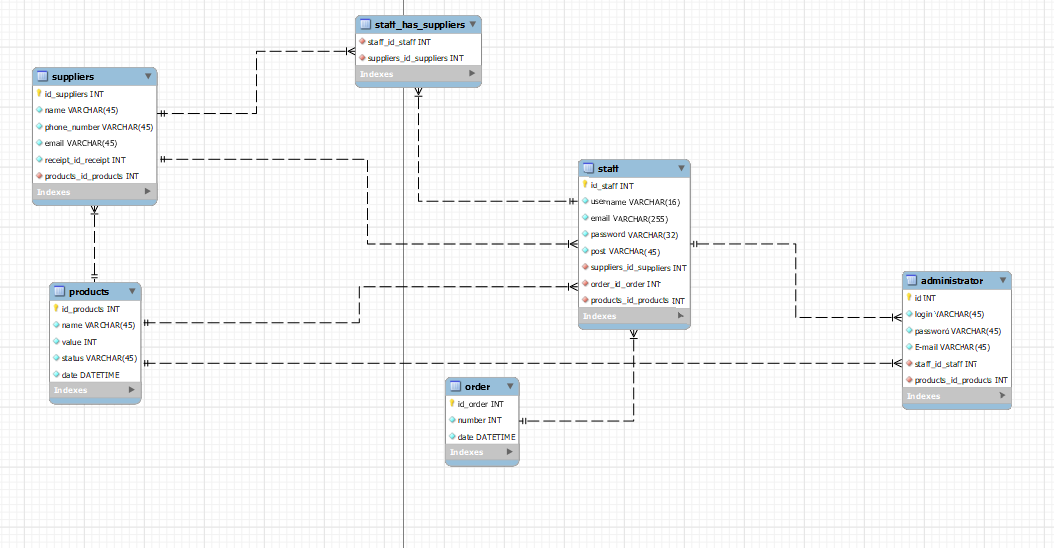


Рисунок 11 – ER-модель базы данных

Таблица 4 – Таблица staff

|  |  |  |
| --- | --- | --- |
| Таблица | Тип данных | Описание |
| id | INT | ID |
| login | VARCHAR(16) | Логин |
| email | VARCHAR(255) | Электронная почта |
| password | VARCHAR(32) | Пароль |
| post | VARCHAR(45) | Должность |
| suppliers\_id\_suppliers | INT | Вторичный ключ |
| order\_id\_order | INT | Вторичный ключ |
| products\_id\_products | INT | Вторичный ключ |

Таблица 5 – Таблица administrator

|  |  |  |
| --- | --- | --- |
| Таблица | Тип данных | Описание |
| id | INT | ID |
| login | VARCHAR(45) | Логин |
| email | VARCHAR(255) | Электронная почта |
| password | VARCHAR(45) | Пароль |
| staff\_id\_staff | INT | Вторичный ключ |
| products\_id\_products | INT | Вторичный ключ |

Таблица 6 – Таблица order

|  |  |  |
| --- | --- | --- |
| Таблица | Тип данных | Описание |
| id | INT | ID |
| number | INT | Номер заказа |
| date | DATETIME | Дата |
| password | VARCHAR(45) | Пароль |
| staff\_id\_staff | INT | Вторичный ключ |
| products\_id\_products | INT | Вторичный ключ |

Таблица 7 – Таблица products

|  |  |  |
| --- | --- | --- |
| Таблица | Тип данных | Описание |
| id | INT | ID |
| name | VARCHAR(45) | Наименование |
| value | INT | Количество |
| status | VARCHAR(45) | Статус товара |
| date | DATETIME | Дата |

Таблица 8 – Таблица suppliers

|  |  |  |
| --- | --- | --- |
| Таблица | Тип данных | Описание |
| id | INT | ID |
| name | VARCHAR(45) | Наименование |
| phone\_number | INT | Телефонный номер |
| email | VARCHAR(45) | Электронная почта |
| receipt\_id\_receipt | INT | Вторичный ключ |
| products\_id\_products | INT | Вторичный ключ |

Таблица 9 – Таблица staff\_has\_suppliers

|  |  |  |
| --- | --- | --- |
| Таблица | Тип данных | Описание |
| staff\_id\_staff | INT | Вторичный ключ |
| suppliers\_id\_suppliers | INT | Вторичный ключ |

Для разработки даталогической и ER-модели был использован инструмент MySQL Workbench.

После завершения разработки прототипа базы данных информационной системы «Склад» получиться готовая схема, по которой будет разрабатываться база данных.

1. 4. Проектирование интерфейса

В результате проектирование интерфейса будущей информационной системы были спроектированы прототипы трёх страниц: главная страница (рисунок 12), страница авторизации (рисунок 13), страница с товарами (рисунок 14). Благодаря созданию прототипов, разработка программного продукта будет значительно облегчена, за счёт наглядных примеров будущих страниц ИС.

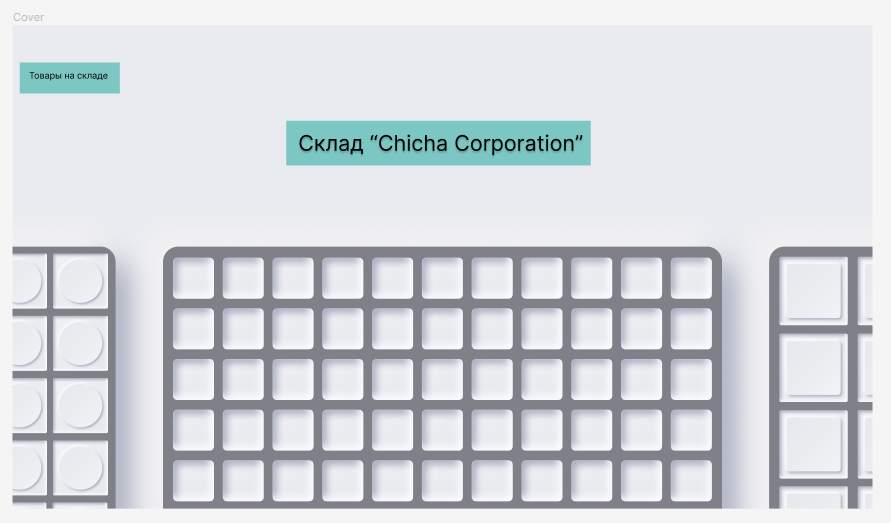


Рисунок 12 – Главная страница

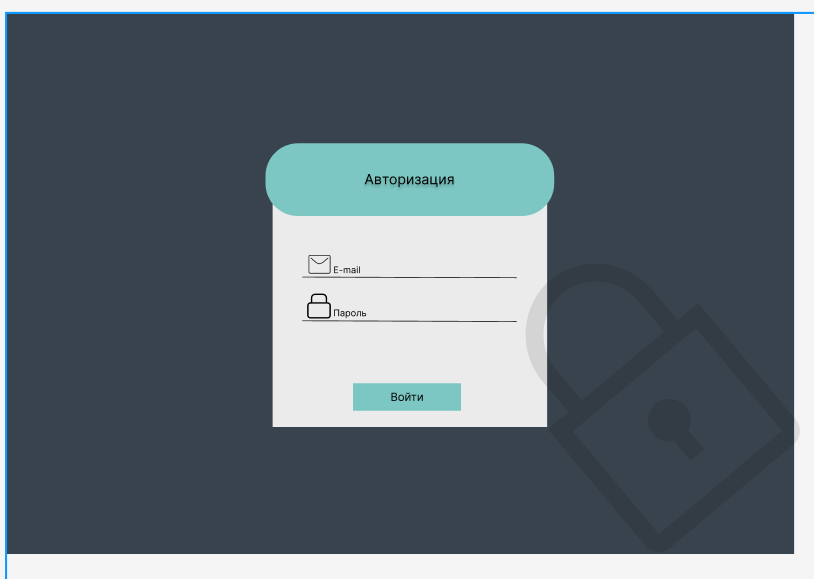


Рисунок 13 – Страница авторизации

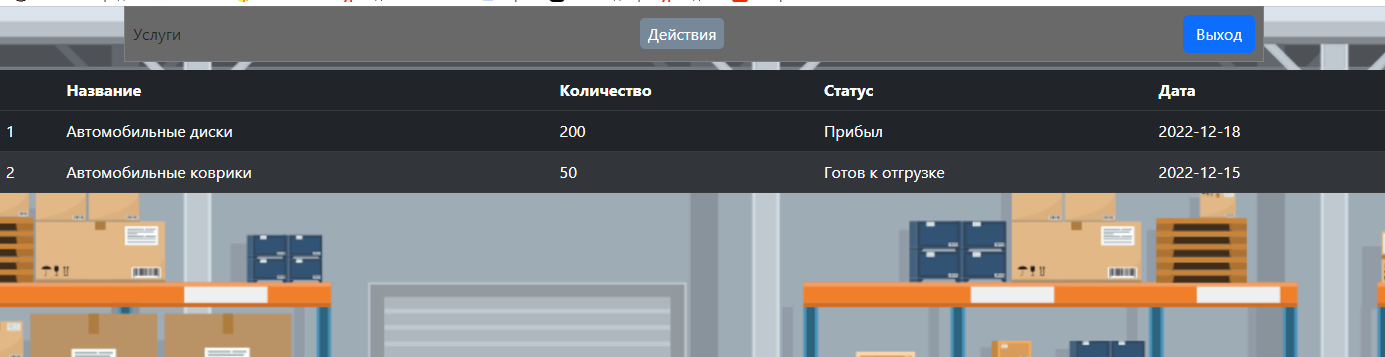


Рисунок 14 – Страница с товарами

1. Разработка ИС
   1. Разработка интерфейса ИС

В информационной системе «Склад» были поставлены и выполнены следующие задачи:

для сотрудников:

* авторизация;
* добавление товаров;
* удаление товаров;
* редактирование товаров.

для администратора:

* создание сотрудников;
* удаление сотрудников;
* редактирование сотрудников;
* добавление новых товаров;
* добавление товаров;
* удаление товаров;
* редактирование товаров.

Для серверной операционной системы, на которой будет располагаться программный код ИС, выдвигаются следующие минимальные требования:

* процессор: Intel Pentium 4 или более поздней версии;
* ОЗУ 512 Мб DDR3;
* 15 ГБ на HDD;
* поддержка БД MySQL;
* операционная система Windows;
* доступ к сети Интернет.

Требования к клиентской части ИС, необходимые для пользования информационной системой:

* процессор Intel Pentium 4 или более поздней версии;
* ОЗУ 2 ГБ.
* доступ к сети Интернет;
* HDD 1ГБ;
* Браузер;
* встроенный видеоадаптер;
* клавиатура, мышь;
* операционная система: Windows.

Общий размер дисковой памяти, занимаемой информационной системой, составляет 2,60 МБ (рисунок 15).



Рисунок 16 – Объем дисковой памяти ИС

Объем потребляемой ОЗУ составляет 30 668К (что примерно равно 30Мб) на одну вкладку в браузере Яндекс (рисунок 15).



Рисунок 16 – Потребляемая ОЗУ в Яндекс браузере

На рисунках 17, 18 показан html код для основной страницы, которую видят все пользователи не зависимо от того, зарегистрированы ли они или нет, а также прошли ли авторизацию и результат кода.



Рисунок 17 – Часть HTML кода основной страницы

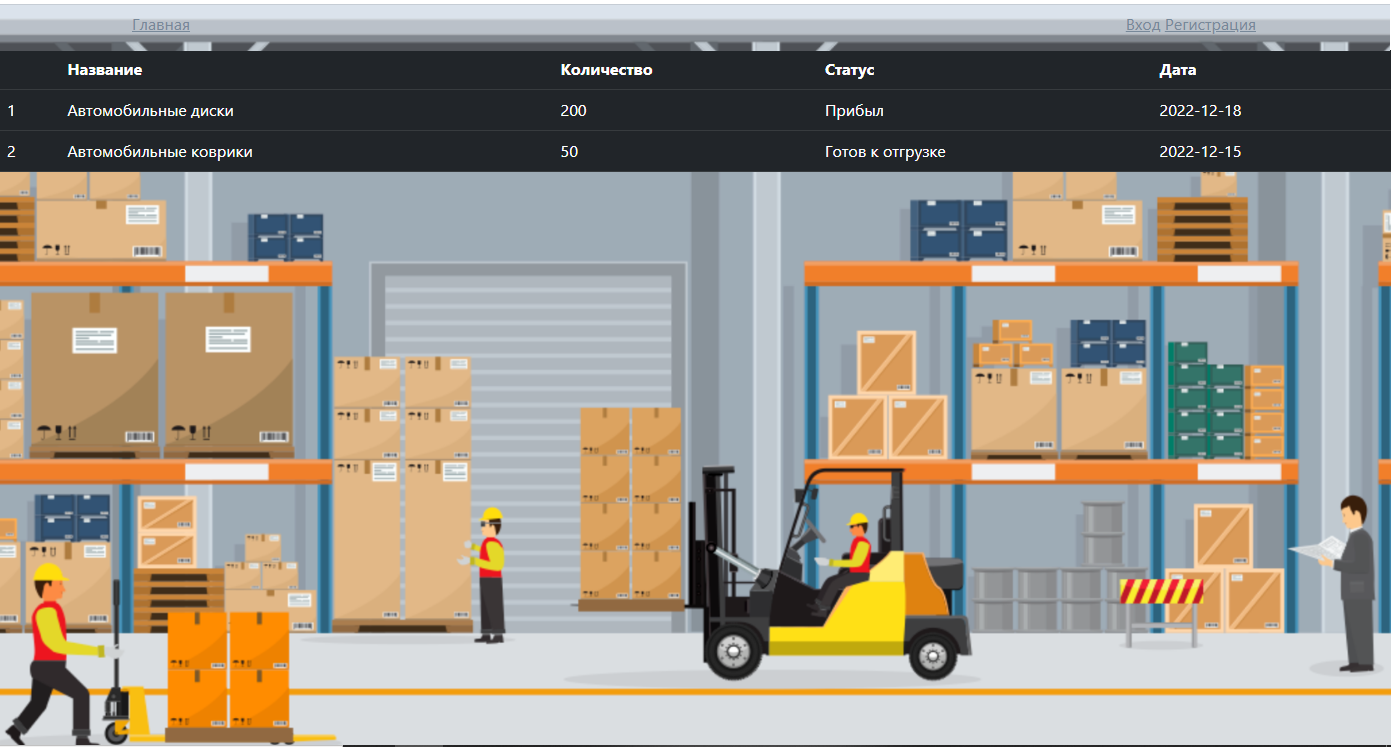


Рисунок 18 – Результат HTML кода основной страницы

На рисунках 19, 20 показан html код для регистрации новых сотрудников, для последующего вноса в базу данных и дальнейшего прохождения авторизации, зарегистрированы ли они или нет, а также прошли ли авторизацию и результат кода.



Рисунок 19 – Часть HTML кода страницы регистрации



Рисунок 20 – Результат html кода страницы регистрации

На рисунках 21, 22 изображен html код страницы авторизации пользователя, для последующего использования информационной системы, и результат кода.

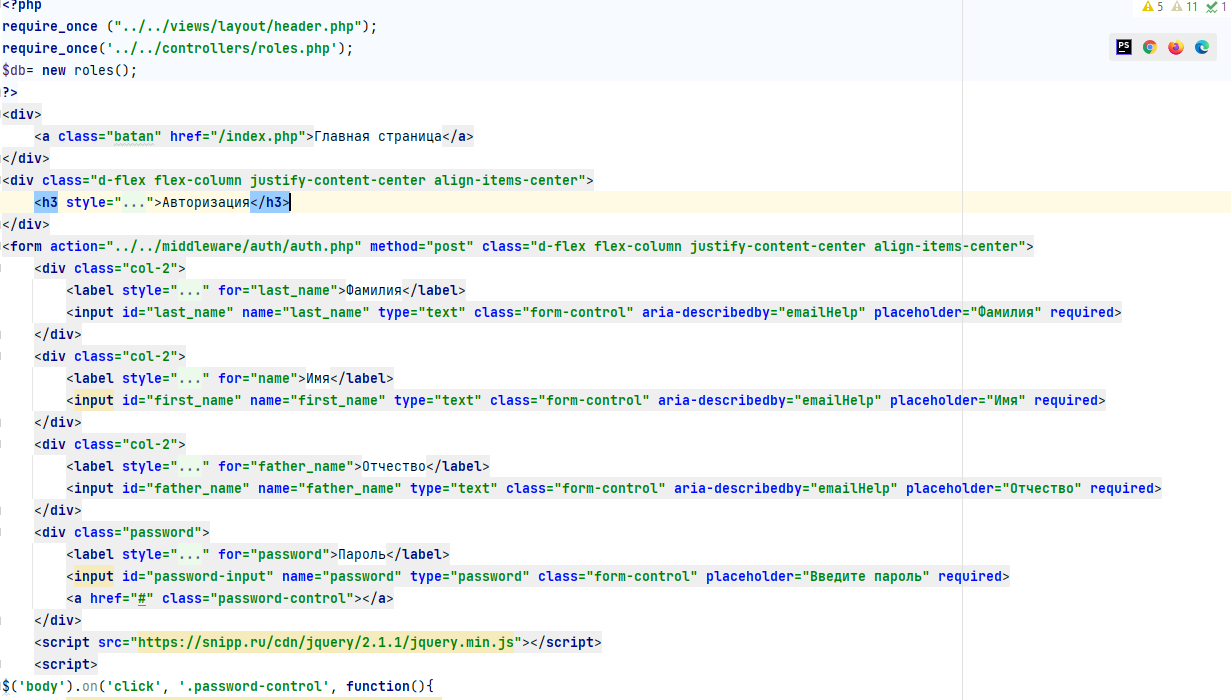


Рисунок 21 – Часть HTML кода страницы авторизации

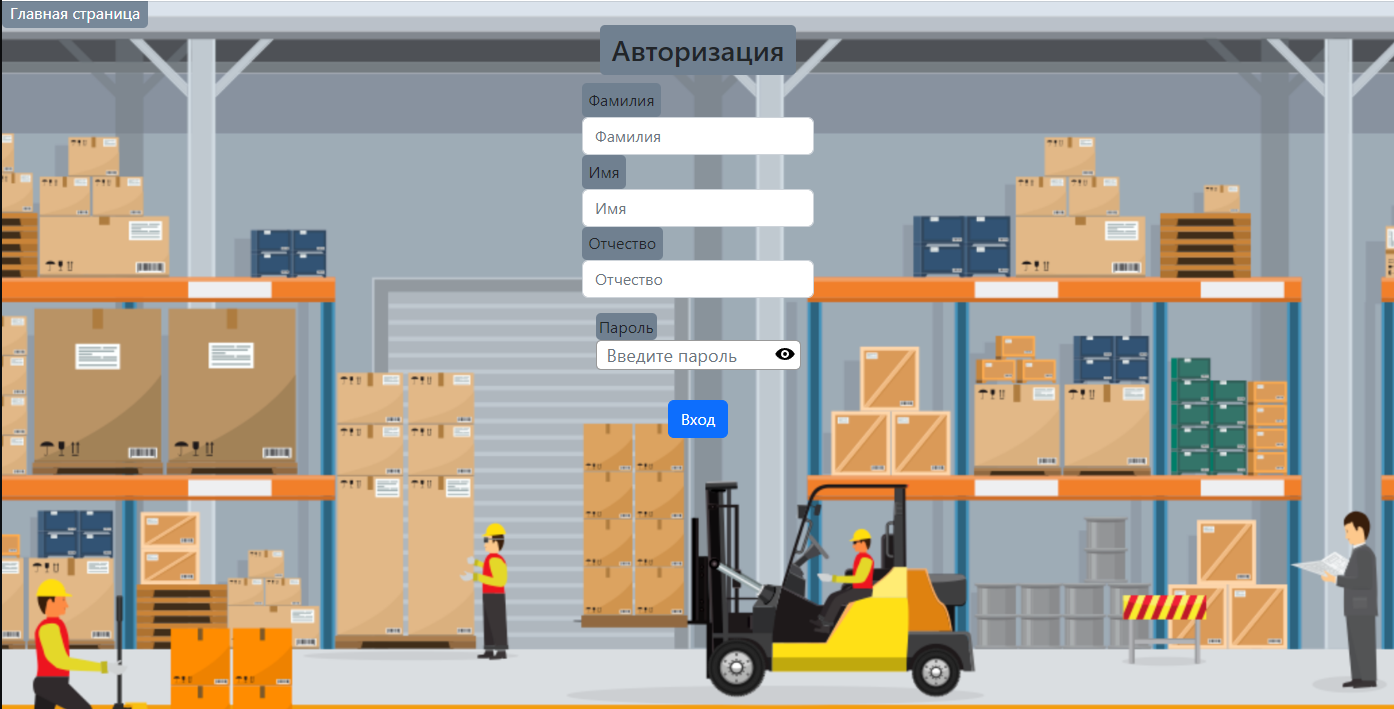


Рисунок 22 – Результат html кода страницы авторизации

Далее было создано 2 личных кабинета общего вида: для администратора и сотрудника. На рисунках 23, 24 изображен html код личного кабинета администратора, с помощью которого администратор может выполнять все действия в ИС.



Рисунок 23 – Часть html код личного кабинета администратора

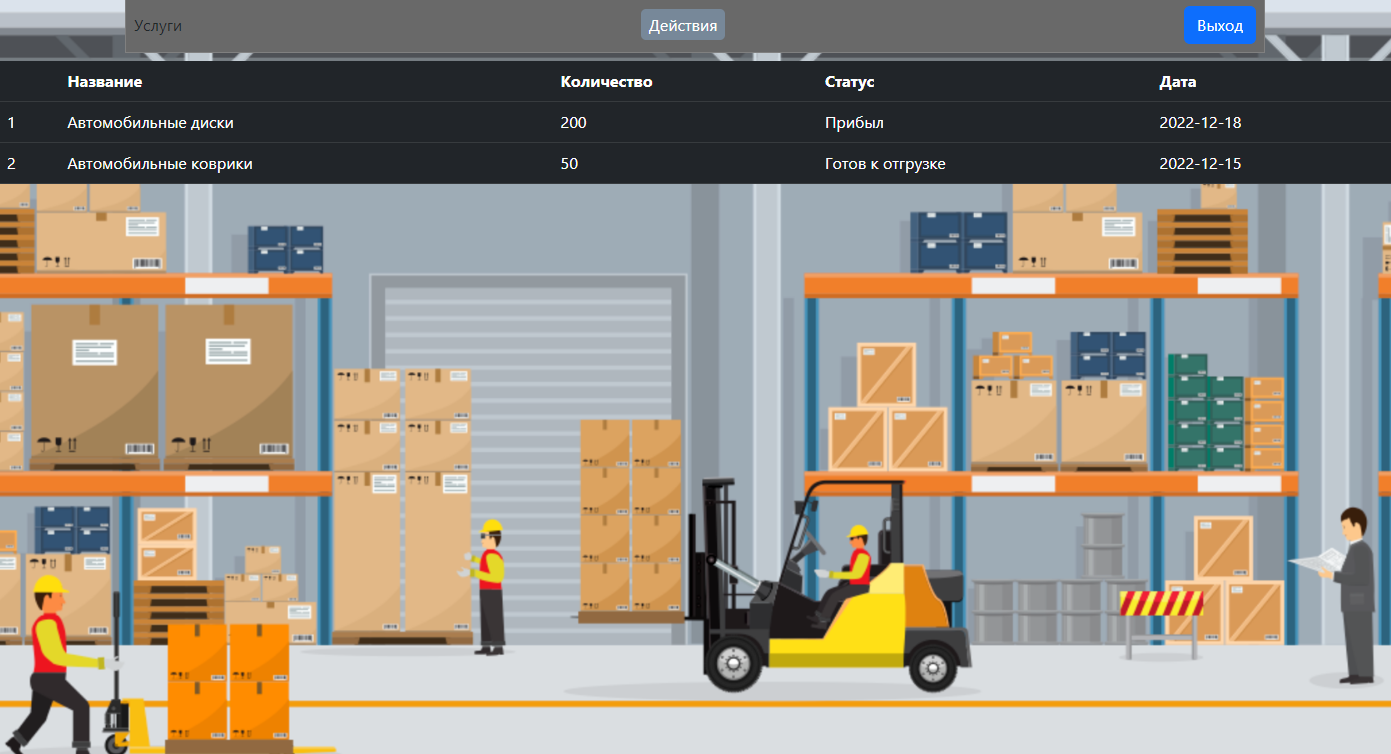


Рисунок 24 – Итоговой вид личного кабинета администратора

На рисунках 25, 26 изображен html код личного кабинета сотрудника, с помощью которого сотрудник может выполнять действия над товарами, прибывшими на склад.



Рисунок 25 – Часть html кода личного кабинета сотрудника

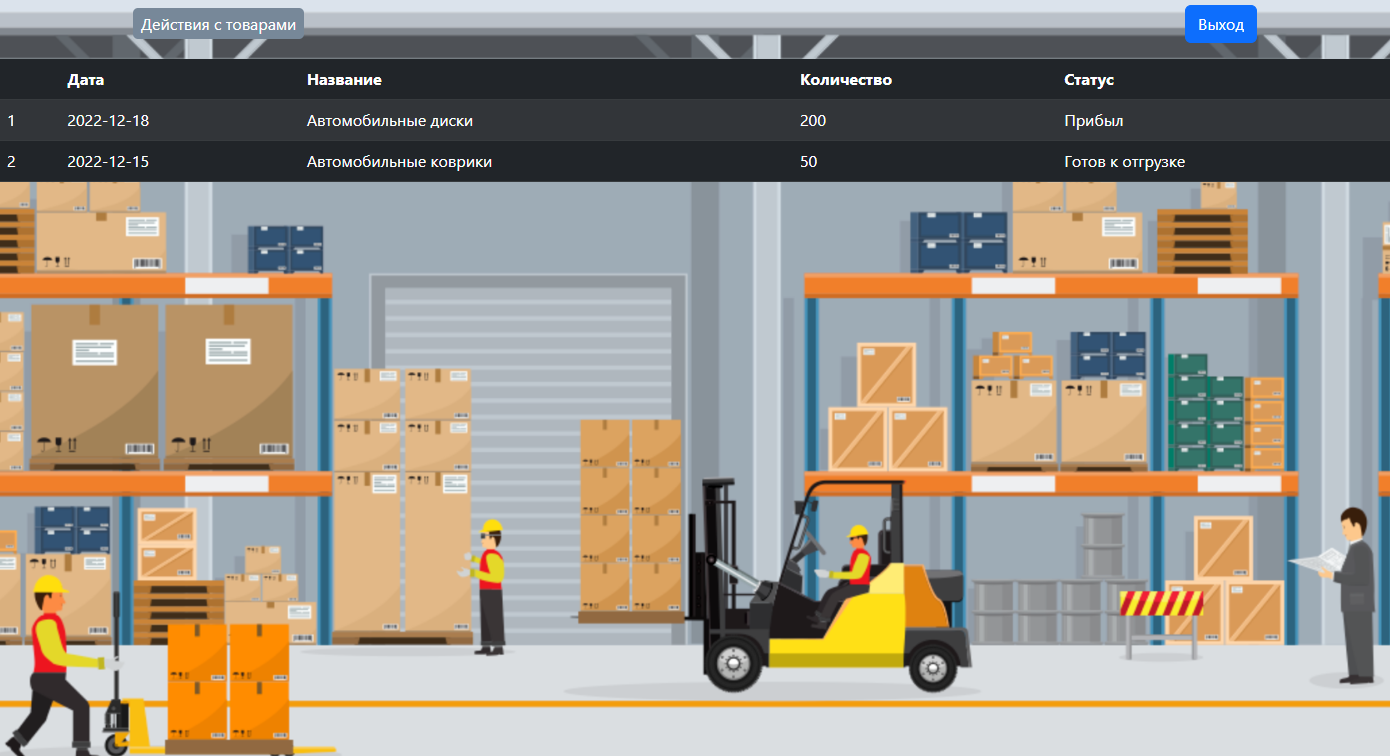


Рисунок 26 – Результат html кода личного кабинета сотрудника

5.2 Разработка базы данных

После того, как разработан интерфейс, необходимо где-то хранить данные, которые будут заноситься пользователями ИС. Для этого необходимо разработать базу данных (далее – БД).

Разрабатывалась БД информационной системы «Склад» во встроенном менеджере баз данных PHPStorm.

БД состоит из 6 таблиц: users, suppliers, staff, products, order, administrator

Структура таблиц соответствует ER-модели из рисунка 11 и таблицам 4-8.

Таблица users на рисунке 27.

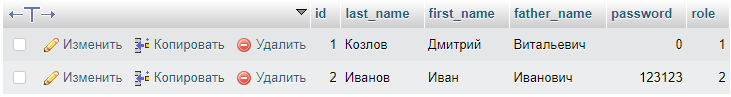


Рисунок 27 – Таблица users

Таблица staff на рисунке 28.



Рисунок 28 – Таблица staff

Таблица products на рисунке 29.

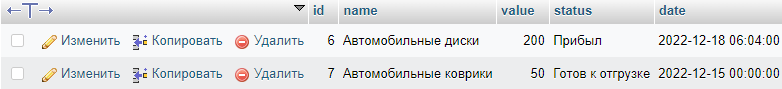


Рисунок 29 – Таблица products

Таблица order на рисунке 30.



Рисунок 30 – Таблица officiants

Таблица suppliers на рисунке 31.



Рисунок 31 – Таблица suppliers

Таблица administrator на рисунке 32.



Рисунок 32 – Таблица administrator

5.3 Разработка ИС

Во время разработки ИС за основу была взят шаблон программирования MVC (рисунок 33).

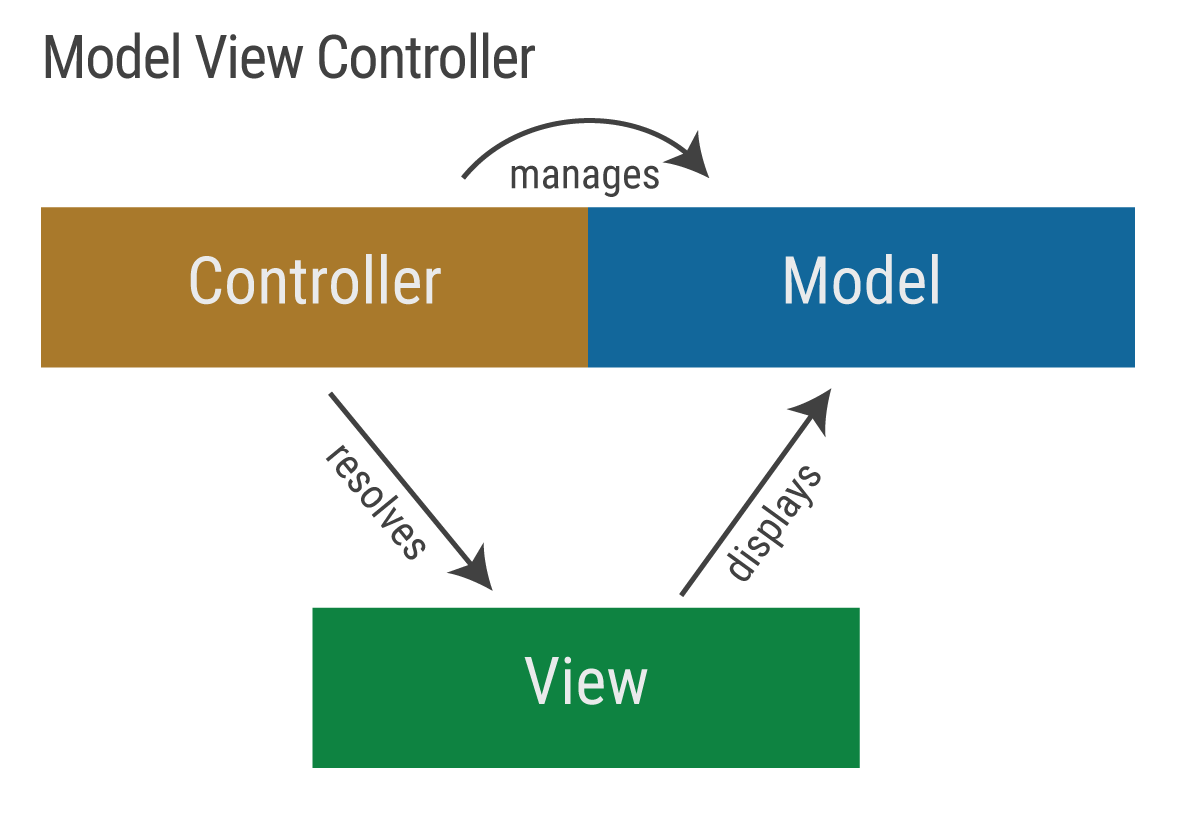


Рисунок 33 – Иллюстрация MVC

MVC - это сокращение названия шаблона программирования Model-View-Controller. Этот шаблон используется для отделения данных от логики и представления. Допустим, есть некий пользователь, он будет моделью, так как содержит данные. Чтобы показать данные, создаём форму, в которую передаём этого пользователя. Это будет вид. Форма может содержать разные кнопки для изменения данных, то есть модели. Чтобы управлять кнопками, к форме привязывают контроллер, который ждёт нажатия кнопок и меняет модель. Таким образом достигается чёткое разделение обязанностей: модель хранит данные, вид показывает, а контроллер изменяет. При таком подходе одна модель может иметь много разных видов, а они, в свою очередь, контроллеров.

Для того, чтобы ИС работала, был создан файл db.php (рисунок 34), в котором осуществляется подключение к БД через PDO (PHP Data Object). В подключении указывается адрес хостинга, а также название БД, кодировка текста, имя пользователя и пароль (который по умолчанию является пустым).



Рисунок 34 – Подключение к базе данных

Затем были реализованы функции авторизации, регистрации и выхода из учётной записи с помощью моделей auth.php, registration.php и logout.php (рисунки 35, 36, 37). Также были созданы представления авторизации (auth.php) и регистрации (registration.php). (рисунок 38, 39) Полная версия программного кода контроллера находится в приложении Б.



Рисунок 35 – Модель auth.php



Рисунок 36 – Модель registration.php

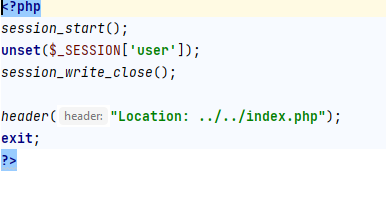


Рисунок 37 – Модель logout.php

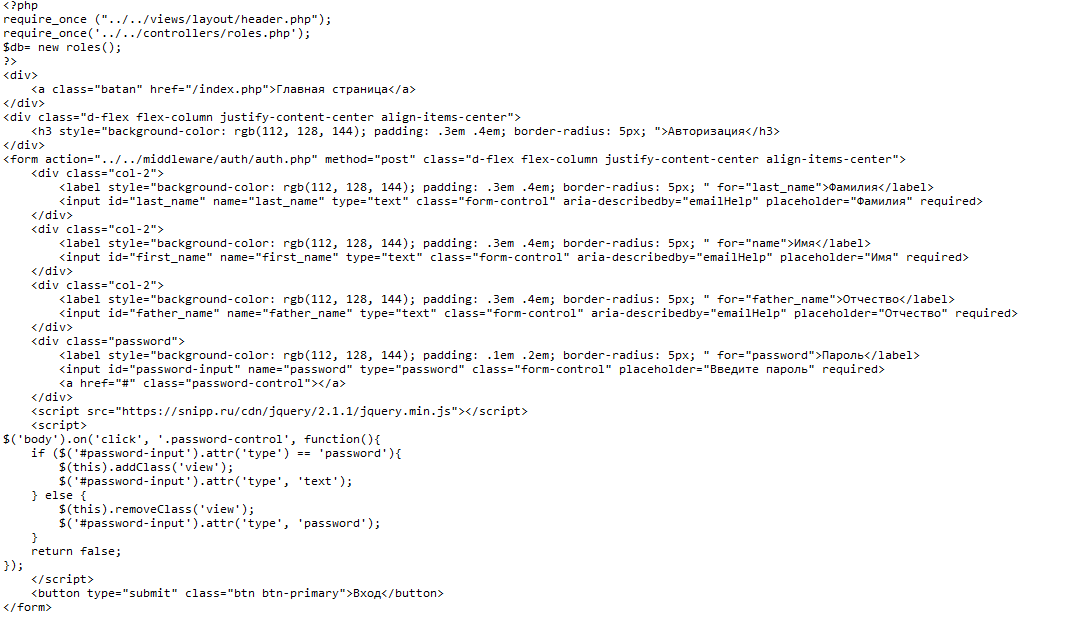


Рисунок 38 – Представление auth.php

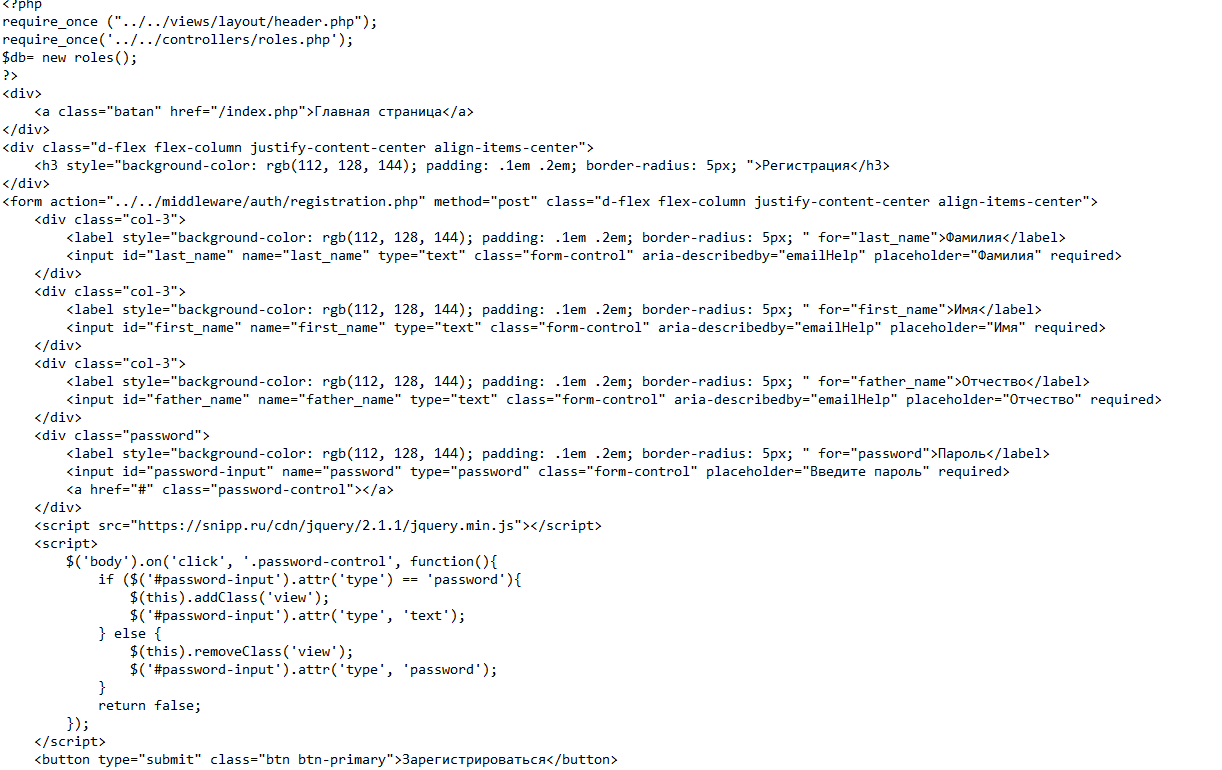


Рисунок 39 – Представление registration.php

6. Технологическая документация ИС
   1. Руководство пользователя ИС

Для работы в ИС необходимо иметь XAAMP для запуска локального веб-сервера. А для того чтобы войти в веб-сервер нужно запустить файл start.bat, скопировать ссылку http://localhost:8080, и вставить её в поисковую строку браузера. После этого откроется главная страница системы.

В данном руководстве пользователя рассмотрен пример личного кабинета сотрудника. На рисунке 27 можно видеть список товаров, поступивших на склад, а также 2 кнопки: «Выход» и «Действия с товарами» для взаимодействия с товарами.

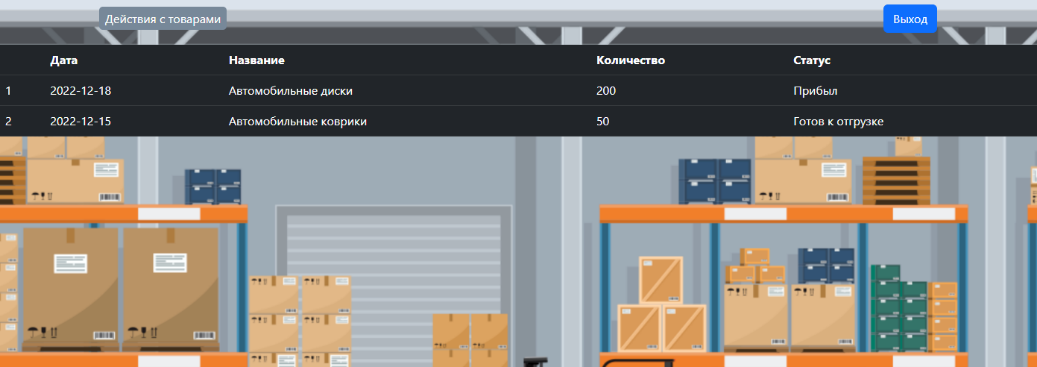


Рисунок 27 – Личный кабинет пользователя

При нажатии на кнопку «Действия с товарами» открывается страница, на которой можно выбрать либо «Удаление товара», либо «Добавление товара», при нажатии на «Добавление товара» открывается соответствующее меню. Это можно увидеть на рисунке 28.

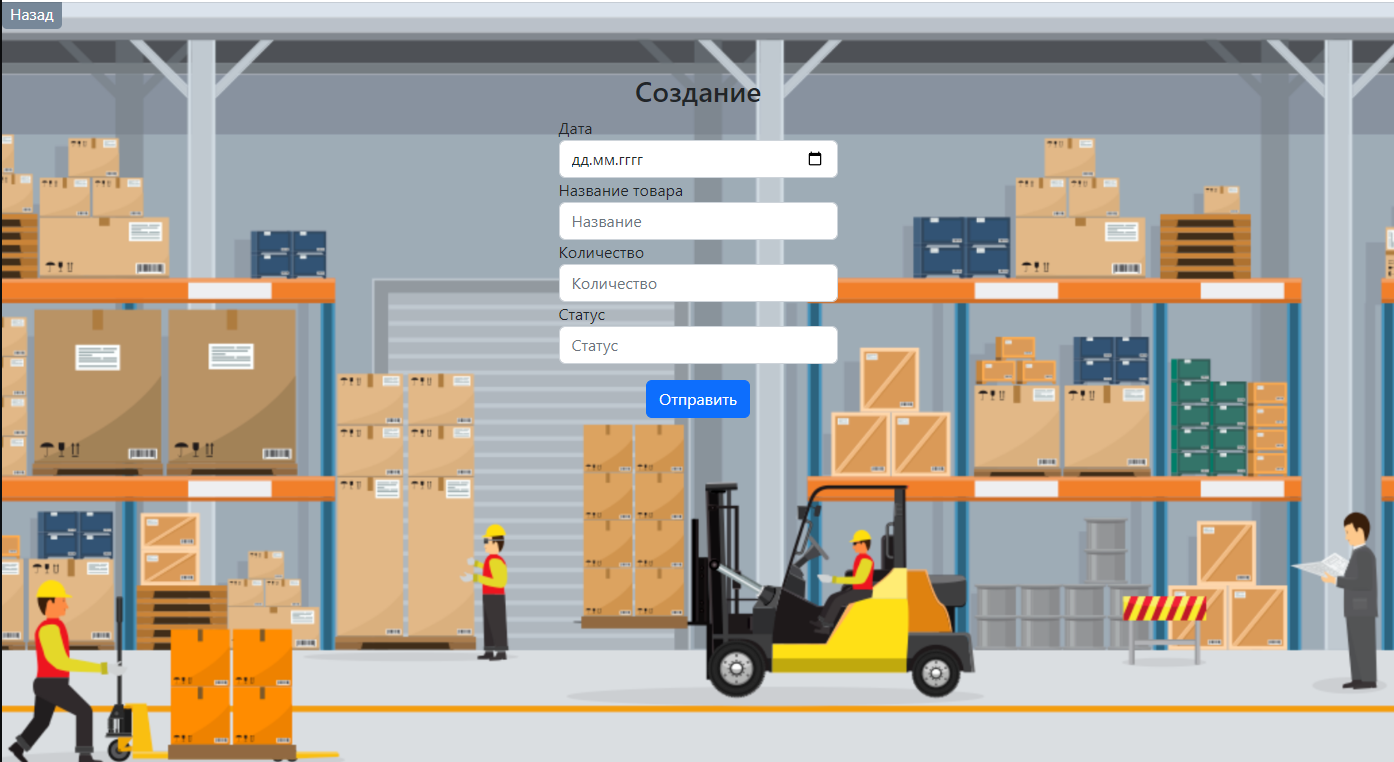


Рисунок 28 – Создание товара

При нажатии на кнопку «Удаление товара», открывается соответствующая вкладка. Представлено на рисунке 29.



Рисунок 29 – Удаление товара

Заключение

В ходе выполнения данного курсовой работы была разработана информационная система «Склад» – система в которой сотрудники могут легко взаимодействовать с товарами, что заметно облегчает их работу. Был определен и реализован следующий функционал информационной системы:

для сотрудников:

* авторизация;
* добавление товаров;
* удаление товаров;
* редактирование товаров.

для администратора:

* создание сотрудников;
* удаление сотрудников;
* редактирование сотрудников;
* добавление новых товаров;
* добавление товаров;
* удаление товаров;
* редактирование товаров.

Были рассмотрены возможные реализации ИС с использованием разных технологий и языков программирования, но в результате анализа инструментальных средств разработки выбор остановился на языке программирования PHP.

Был разработан браузерный программный продукт, с широким функционалом, а также понятным и лёгким интерфейсом.

В дальнейшем ИС может развиваться путём расширения функционала и совершенствования интерфейса, тем самым набирая большее сообщество пользователей, а также и актуальность программного продукта.

Все поставленные цели и задачи курсового проекта были успешно выполнены.

Список используемых материалов

1. Colorscheme – Цвета HTML. Таблица из 147 имён цветов для HTML и CSS – URL: https://colorscheme.ru/html-colors.html (дата обращения: 15.10.2022). – Текст: электронный.
2. Htmlbook – Самоучитель HTML4 – URL: http://htmlbook.ru/samhtml (дата обращения: 15.10.2022). – Текст: электронный.
3. HTML5 BOOK – Основы CSS – URL: https://html5book.ru/osnovy-css/ (дата обращения: 20.10.2022). – Текст: электронный.
4. Itnan – MariaDB в сравнении с MySQL в 2022 году – URL: https://itnan.ru/post.php?c=1&p=662870 (дата обращения: 24.10.2022). – Текст: электронный.
5. JetBrains – Возможности PhpStorm – URL: https://www.jetbrains.com/ru-ru/phpstorm/features/ (дата обращения: 20.10.2022). – Текст: электронный.
6. MDN – Основы HTML – URL: https://developer.mozilla.org/ru/docs/Learn/Getting\_started\_with\_the\_web/HTML\_basic (дата обращения: 22.09.2022). – Текст: электронный.
7. Oracleplsql – URL: https://oracleplsql.ru/mariadb-manual.html (дата обращения: 24.10.2022). – Текст: электронный.
8. Php.net – Что такое PHP? – URL: https://www.php.net/manual/ru/intro-whatis.php (дата обращения: 24.09.2022). – Текст: электронный.
9. StudFiles – Инфологическое моделирование. Er-модель. – URL: https://studfile.net/preview/6862142/page:11/ (дата обращения: 28.09.2022). – Текст: электронный.
10. zaLinux – Изучение MySQL/ MariaDB – URL: https://zalinux.ru/?p=760 (дата обращения: 24.09.2022). – Текст: электронный.

Приложение А ­­– Техническое задание

**Министерство образования Иркутской области**

Государственное бюджетное профессиональное

образовательное учреждение Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

|  |  |
| --- | --- |
|  |  |

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

ИНФОРМАЦИОННАЯ СИСТЕМА

«СКЛАД»

|  |  |  |
| --- | --- | --- |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Кудрявцева) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Д.В. Козлов) |

Иркутск 2022

**1 Общие сведения**

Наименование работы: информационная система «Склад».

Исполнитель: студент иркутского авиационного техникума, группы ИС-20-1, Козлов Д.В

Разработка информационной системы проходит в рамках курсовой работы.

Сроки разработки информационной системы с 21.09.2022 по 06.12.2022 года.

**2 Цели и назначение создания автоматизированной системы**

Целью курсовой работы является создание информационной системы «Склад», которое включает в себя возможности внесение в списки новых заказов и редактирования старых, отслеживания товаров.

В данной информационной системе реализуется следующий функционал:

для сотрудников:

* авторизация;
* добавление товаров;
* удаление товаров;
* редактирование товаров.

для администратора:

* создание сотрудников;
* удаление сотрудников;
* редактирование сотрудников;
* добавление новых товаров;
* добавление товаров;
* удаление товаров;
* редактирование товаров.

3 Требования к системе в целом

3.1. Требования к структуре и функционированию приложения

Функции веб-приложения:

1. Окно «Авторизации»:
   1. авторизация пользователей.
2. Окно «Регистрация»:
   1. регистрация пользователей.
3. Окно «Действия с товарами»:
   1. Информация о товарах;
      1. добавление наименований;
      2. удаление наименований;
      3. редактирование данных.
   2. Окно «Сотрудники»:
      1. добавление сотрудников;
      2. удаление сотрудников;
      3. редактирование данных.
   3. **. Требования к надежности**

Для обеспечения надежности необходимо проверять корректность получаемых данных и реализовать валидность полей. Входные данные поступают в виде значений c клавиатуры. Эти значения отображаются в отдельных полях таблицы.

* 1. **Требования к безопасности**

Для обеспечения безопасности в информационной системе, необходимо реализовать разграничение прав доступа.

* 1. **Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы**

Минимальные системные требования для сервера:

1. Процессор: Intel Pentium 4 2.0Ghz / AMD XP 2200+;
2. Оперативная память: 512 Мб;
3. Жёсткий диск: 150мб;
4. Операционная система: Windows 7/8/10.
5. Версия MySQL 5.0 и выше;

Минимальные системные требования для рабочей станции:

1. Процессор: Intel Pentium 4 2.0Ghz / AMD XP 2200+;
2. Оперативная память: 512 Мб;
3. Жёсткий диск: 150мб;
4. Операционная система: Windows 7/8/10.
5. **Требования к документированию**

Основным документом, регламентирующими использование информационной системы является руководство пользователя.

Основным документом, регламентирующими разработку информационной системы является техническое задание.

1. **Состав и содержание работ по созданию системы**

В таблице 1 представлены плановые сроки начала и окончания работы по созданию веб-приложения.

Таблица 1 – Этапы выполнения разработки

|  |  |  |
| --- | --- | --- |
| Наименование этапов разработки | Срок | |
| Начало | Окончание |
| Предпроектное исследование предметной области (выбор темы, постановка цели, задач, описание области применения, исследование предметной области) | 15.09.22 | 21.09.22 |
| Разработка технического задания (выбор архитектуры программного обеспечения, выбор типа пользовательского интерфейса, выбор языка и среды программирования) | 21.09.22 | 26.09.22 |
| Проектирование программного обеспечения.  (разработка структурной и функциональной схемы ПО, проектирование базы данных (инфологическое, ER-модель, физическая модель) | 26.09.22 | 25.10.22 |
| Разработка (программирование) и отладка программного продукта | 25.10.22 | 28.11.22 |
| Составление программной документации (написание руководства пользователя) | 28.11.22 | 06.12.22 |

Приложение Б – Листинг User.php

<?php

require ('db.php');

class Staff extends DB

{

public function get(){

return $this->DBAll('SELECT \* from staff');

}

public function deleteStaff($request){

$req=json\_decode($request);

return $this->transaction(

'DELETE from staff where id='.$req->id,

'Сотрудник удален');

}

public function createStaff($request){

$req = json\_decode($request);

$last\_name = $req->last\_name;

$first\_name = $req->first\_name;

$father\_name = $req->father\_name;

$post = $req->post;

$connect = $this->connect();

try{

$connect->beginTransaction();

$connect->exec("INSERT INTO staff (last\_name,first\_name,father\_name,post) values ('{$last\_name}','{$first\_name}','{$father\_name}','{$post}')");

$connect->commit();

return json\_encode([

'message'=>'Сотрудник добавлен'

]);

}catch (PDOException $e){

$connect->rollBack();

return json\_encode([

'message'=>$e->getMessage()

]);

}

}

public function updateStaff($request){

$req = json\_decode($request);

$id = $req->id;

$last\_name = $req->last\_name;

$first\_name = $req->first\_name;

$father\_name = $req->father\_name;

$post = $req->post;

$connect = $this->connect();

try{

$connect->beginTransaction();

$connect->exec("UPDATE staff SET last\_name='{$last\_name}', first\_name='{$first\_name}', father\_name='{$father\_name}', post='{$post}'

WHERE id={$id} ");

$connect->commit();

return json\_encode([

'message'=>'Сотрудник обновлён'

]);

}catch (PDOException $e){

$connect->rollBack();

return json\_encode([

'message'=>$e->getMessage()

]);

}

}

}