

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**  
( **Н И У « Б е л Г У »** )

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК

КАФЕДРА МАТЕМАТИЧЕСКОГО И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
ИНФОРМАЦИОННЫХ СИСТЕМ

**РАЗРАБОТКА WEB-ПРИЛОЖЕНИЯ ДЛЯ УЧЕТА И КОНТРОЛЯ  
ПРОИЗВОДСТВА МЕБЕЛИ НА ООО «ТРИЛЛИАНТ»**

Выпускная квалификационная работа  
обучающейся по направлению подготовки 02.03.02 Фундаментальная  
информатика и информационные технологии  
очной формы обучения, группы 07001301  
Кичигиной Анастасии Константиновны

Научный руководитель  
к.т.н., доцент  
Михелев В.М.

БЕЛГОРОД 2017

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	3
ГЛАВА 1 ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ АВТОМАТИЗАЦИИ УЧЕТА И КОНТРОЛЯ ПРОИЗВОДСТВА ПРОДУКЦИИ .....	5
1.1 Описание целей, задач и функционала предприятия ООО «Триллиант» .....	5
1.2 Описание систем информатизации предприятия .....	7
1.3 Теоретические основы учета и контроля производства продукции .....	8
1.4 Обусловленность выбора средств разработки и языков программирования	14
ГЛАВА 2 РАЗРАБОТКА WEB-ПРИЛОЖЕНИЯ УЧЕТА И КОНТРОЛЯ ПРОИЗВОДСТВА МЕБЕЛИ.....	17
2.1 Описание архитектуры клиент-сервер.....	17
2.2 Описание использованных в проекте фреймворков.....	19
2.3 Описание базы данных .....	24
2.4 Описание этапов разработки.....	32
2.5 Пользовательский интерфейс и руководство работы с программой.....	40
ГЛАВА 3 АПРОБАЦИЯ И ВНЕДРЕНИЕ WEB-ПРИЛОЖЕНИЯ.....	52
3.1 Тестирование работоспособности приложения .....	52
3.2 Внедрение в производство .....	54
ЗАКЛЮЧЕНИЕ .....	55
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ .....	56
ПРИЛОЖЕНИЯ.....	58

## ВВЕДЕНИЕ

В настоящее время компьютеров, мобильных средств связи, Интернета и высоких технологий все более актуальным становится использование компьютерных систем, Web-приложений для автоматизации бизнеса и работы специалиста.

Предприятия мебельной отрасли давно осознали необходимость внедрения автоматизированных систем управления, учета и контроля. Автоматизация производства мебели позволяет оптимизировать все основные бизнес-процессы управления и повысить их эффективность в среднем на 20-30%.

Разработка качественных автоматизированных программных продуктов помогает структурировать функциональные составляющие организации. Единое информационное пространство и использование компьютерных информационных технологий позволяет отладить и организовать взаимодействие между разными отделами, повысить оперативность их работы. Обмен информацией, передача отчетов, оформление различных документов реализуется намного быстрее и удобнее. Все операции по обработке данных, поиску, выполнению запросов можно совершать дистанционно.

Целью выполнения выпускной квалификационной работы является разработка Web-приложения для учета и контроля производства мебели, направленной на автоматизацию производственных процессов мебельного предприятия.

Для достижения цели проекта поставлены следующие задачи:

- ✓ изучить структуру предприятия ООО «Триллиант», сценарии его работы и бизнес-процессы, сотрудничая с менеджерами, осуществляющими учет и контроль производства продукции;
- ✓ определить функции и задачи, которые возможно и необходимо автоматизировать;

✓ выбрать из всех компьютерных средств наиболее подходящие для создания Web-приложения, которое должно обладать следующими характеристиками и функциями:

1. в базе данных системы должна храниться и обрабатываться исчерпывающая информация о продукции, материалах, сборке, производстве, заказах и т.д.;

2. система должна давать возможность добавления, удаления, изменения данных;

3. осуществлять контроль производства, учет расхода основных и вспомогательных материалов, расчет себестоимости товара;

4. предоставлять возможность поиска, фильтрации, представления в удобном виде всей необходимой пользователю информации;

5. вход в систему должен предоставляться только администратору и менеджеру, зарегистрированному в ней;

6. на действия пользователя программа должна выдавать соответствующие сообщения предупреждающего, поясняющего, запрещающего, информативного характера для удобства работы и предупреждения ошибок и сбоев;

7. разрабатываемая программа должна обеспечивать быстрое и удобное получение информации, иметь простой и понятный интерфейс.

✓ Разработать и создать базу данных;

✓ Разработать программную реализацию Web-приложения и провести ее тестирование.

Для реализации проекта необходимо также глубже изучить алгоритмические структуры, систему команд, функций и процедур выбранного языка программирования и СУБД, показать умение и навыки использования их на практике.

Данная выпускная квалификационная работа состоит из 57 страниц, в ней имеется 32 рисунка и приложение с 4 приведенными листингами программы.

# ГЛАВА 1 ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ АВТОМАТИЗАЦИИ УЧЕТА И КОНТРОЛЯ ПРОИЗВОДСТВА ПРОДУКЦИИ

## 1.1 Описание целей, задач и функционала предприятия ООО «Триллиант»

Общество с ограниченной ответственностью «Триллиант» - это российская, стабильно развивающаяся и расширяющаяся мебельная компания, которая была основана 12 февраля 2003 года.

В настоящее время компания производит широкий ассортимент мебели на любой вкус: кухни, гостиные, шкафы-купе, офисная мебель, спальни, прочая корпусная мебель.

Помимо основной продукции, клиентам предлагаются наборы мебели и модульные системы для гостиной, спальни, а также детские, прихожие, кухни, раскладные столы, журнальные и компьютерные столы, тумбы для обуви и комоды, ТВ-тумбы, стеллажи и полки.

Компания демонстрирует своим покупателям мебель разнообразного фасона и дизайна, широкую гамму цветов – более 60 расцветок. В зависимости от предпочтений, покупатель всегда сможет выбрать мебель различного цвета и дизайна.

Преимуществом компании «Триллиант» на рынке является широкий выбор ассортимента корпусной и мягкой мебели. В компании работают лучшие специалисты по разработке актуальных дизайнов мебели, предлагают разнообразие мебели от классики до модерна. Сохраняя соотношение цена/качество, опережают конкурентов на много шагов вперед. Персонал компании постоянно ведет работу по непрерывному обновлению разнообразных коллекций любой мебели. Ассортимент мебели расширяется, предлагая покупателям выбор товара на любой вкус и цвет.

Мебельная компания «Триллиант» следит за современными трендами мебельного рынка и, учитывая пожелания своих покупателей, ежемесячно создает около трех новых моделей.

За многолетний опыт предприятия сформировался определённый стиль работы, основой которого являются:

- консультация и выполнение замеров помещения для создания идеального решения для изготовления корпусной мебели;
- разработка и сопровождение дизайн-проектов корпусной мебели с учётом пожеланий клиентов, для различных помещений;
- изготовление по разработанным проектам дизайна корпусной мебели на заказ;
- доставка и установка готовой продукции.

Для данного предприятия изготовление и установка корпусной мебели является важнейшим направлением деятельности. Достижение данной цели осуществляется за счёт:

- организации высокого уровня привлечения «Заказчиков», сбыта продукции и оказываемых услуг;
- постоянного роста объёмов продаж продукции и оказываемых услуг;
- наиболее полного удовлетворения существующего спроса в оказании услуг по изготовлению и проектированию корпусной мебели;
- многолетнего опыта и обеспечения высокого доверия «Заказчика».

Фирма ООО «Триллиант» имеет свою собственную структуру предприятия, в состав которой входят такие отделы как:

- складские помещения;
- отдел бухгалтерии;
- распиловочный и сборочный цеха;
- 2 офиса продаж.

Особое внимание стоит уделить принципу управления данным предприятием, который отображен на рис. 1.1.



Рис 1.1. Структура управления предприятием

Изготовление и установка корпусной мебели является основным видом деятельности. Основным материалом для такой мебели служат древесно-стружечная плита (ДСП) или спрессованная древесная пыль (МДФ).

Клиентам также предлагается фурнитура и другие аксессуары, такие как: мойки, вытяжки, смесители, ручки, рейлинговые системы, организаторы и так далее.

## 1.2 Описание систем информатизации предприятия

Структура информатизации предприятия ООО «Триллиант»:

- индивидуальные рабочие станции пользователей;
- особые каналы связи для каждого офиса, созданные для получения необходимой информации;
- средства коммуникаций, позволяющие функционировать всем каналам связи [1].

Локальные и любые другие сети отсутствуют между рабочими станциями, так как это нецелесообразно.

В каждой рабочей станции имеется набор основного и специального программного обеспечения. Офисные станции так же дополнительно оборудованы специализированным программным обеспечением.

Приведем для примера некоторые программные продукты, используемые на предприятии:

- PRO100 – служит для создания дизайн-интерьеров, работа с фотореалистичными изображениями, проектировки корпусной мебели;
- Астра раскрой – удобное программное обеспечение, служащее для создания оптимальных карт раскроя листовых материалов;
- Skype – служит для связи пользователей между офисами;
- TeamViewer – программное обеспечение, служащее для предоставления удаленного доступа к компьютерам с возможностью устранения неполадок, поддержания рабочих станций в качественном для работы состоянии;
- Woody – программа для проектировщиков, которые планируют изготавливать мебель на высоком уровне.

Проанализировав наличие имеющегося информационного обеспечения, можно сделать вывод, что на предприятии нет единой автоматизированной системы, позволяющей объединить все бизнес-процессы, связанные с учетом и контролем производства продукции. Следовательно, разрабатываемое в проекте Web-приложение будет дополнительным подспорьем в организации и развитии его бизнеса [1].

### **1.3 Теоретические основы учета и контроля производства продукции**

Рыночные отношения в наше время намного усложняют процесс управления предприятием. Огромное количество фирм и организаций

предоставлены сами себе в своей хозяйственной и финансовой деятельности [3].

Учет и контроль являются основными функциями управления производственной деятельностью предприятия, которая напрямую зависит от уровня информационного обеспечения отделов учета и контроля.

Бухгалтерский учет делится на финансовый и управленческий. Задачей управленческого учета является составление отчетов для целей периодического планирования и контроля, для принятия решений в нестандартных ситуациях.

Управленческий учет представляет собой систему учета, планирования, контроля, анализа информации о затратах и результатах хозяйственной деятельности, необходимой управленческому персоналу для управления деятельностью организации.

Управленческий учет становится связующим звеном между учетным процессом и управлением предприятием.

Производственная деятельность организации в целом и ее отдельных структурных подразделений (центров ответственности) выступает предметом управленческого учета.

Объектами являются затраты и результаты хозяйственной деятельности предприятия и его центров ответственности, внутреннее ценообразование и внутренняя отчетность [3].

Основное содержание управленческого учета составляет калькулирование и учет затрат на производство. Функцией производственного учета в наши дни является наблюдение за затратами производства, анализ причины перерасхода по сравнению с предыдущими периодами, прогнозами, стандартами и выявление возможных резервов снижения затрат. Он включает учет издержек по видам, по местам их возникновения [4].

Наиболее важными целями управленческого учета являются:

- оказание управляющим информационной помощи в принятии оперативных управленческих решений;
- контроль, планирование и прогнозирование экономической эффективности деятельности предприятия;
- обеспечение базы для ценообразования;
- выбор наиболее эффективных путей развития предприятия.

Взаимосвязь финансового и управленческого учета заключается в следующем:

- информация двух видов учета используется для принятия решений;
- однократное ведение первичной информации не только для управленческого учета, но и для составления финансовой отчетности;
- общепринятые принципы финансового учета применяются в управленческом учете [4]:
  - единые объекты учета;
  - единый подход к выбору целей и задач учета;
  - использование общих методов в каждом из видов учета (документация, инвентаризация, оценка и калькуляция, группировка объектов учета, отчетность подразделений).

Управленческий учет охватывает все виды информации, необходимой для управления в пределах организации. Частью общей сферы управленческого учета является производственный учет, под которым обычно понимают учет издержек производства и анализ данных об экономии или перерасходе по сравнению с данными за предыдущие периоды, прогнозами и стандартами. Основная цель управленческого учета — обеспечение информацией менеджеров, ответственных за достижение конкретных производственных показателей. Процесс подготовки такой информации может существенно отличаться от того, что используется в финансовом учете.

Изучение особенностей управленческого учета позволяет сделать вывод о том, что он служит для:

- предоставления необходимой информации администрации для управления производством и принятия решений на перспективу;
- исчисления фактической себестоимости продукции (работ и услуг) и отклонений от установленных норм, стандартов, смет;
- определения финансовых результатов по реализованным изделиям или их группам, новым технологическим решениям, центрам ответственности и другим позициям.

Внутренний контроль выступает составной частью системы корпоративного управления и осуществляется как непосредственно руководством и другими должностными лицами хозяйствующего субъекта, так и его специальными контрольными службами или привлеченными специализированными организациями. В свою очередь, внутренний контроль может осуществляться собственником и исполнительным органом путем создания специальной службы внутреннего контроля либо непосредственно управленческим персоналом. В последнем случае такой контроль не редко называют управленческим, рассматривая его по аналогии с управленческим учетом одним из средств решения управленческих задач [4].

Общественный контроль осуществляется органами самодеятельности и представляет собой самый массовый вид экономического контроля.

По сфере применения экономического контроля различают финансовый и специализированный контроль.

Финансовый контроль охватывает финансовую сферу как совокупность экономических отношений, связанных с образованием и использованием фондов денежных средств.

Специализированный контроль охватывает не финансовую, а специализированную сферу экономических отношений, связанных с технологическими и иными специфическими аспектами экономического производства.

По времени проведения экономического контроля различают предварительный, текущий и последующий контроль.

Предварительный контроль осуществляется до начала совершения хозяйственных операций, подвергающихся контролю, и направлен на предупреждение незаконного и нерационального использования экономических ресурсов.

Текущий контроль проводится непосредственно в процессе совершения хозяйственных операций и призван оперативно устранять недостатки, выявлять и закреплять положительные тенденции в финансово-хозяйственной деятельности хозяйствующих субъектов.

Последующий контроль осуществляется после совершения хозяйственных операций и направлен на выявление уже допущенных недостатков или примененного положительного опыта. Благодаря последующему контролю обеспечивается объективная оценка деятельности хозяйствующих субъектов и отдельных должностных лиц.

По источникам контрольных данных выделяют документальный и фактический контроль.

Документальным называется контроль, осуществляемый по документальным данным. Специфика источников документального контроля состоит в том, что они могут быть как достоверными, так и недостоверными, полными и неполными, что, безусловно, сказывается и на выборе технических приемов его осуществления, которые связаны, главным образом, с проверкой документов.

Фактическим считается контроль, осуществляемый по данным осмотра в натуре проверяемых объектов. Достоинство фактического контроля состоит в том, что он обеспечивает безусловную достоверность контрольных данных. Однако из-за отсутствия необходимой информации во всей полноте он не в состоянии воссоздать всю картину хозяйственных процессов и объектов, поэтому его следует сочетать с документальным контролем.

По охвату проверяемых объектов выделяют сплошной и выборочный контроль [4].

Сплошной контроль означает проверку объектов (хозяйственных операций) в сплошном порядке и характеризуется большой трудоемкостью, в связи с чем он используется по отдельным участкам финансово-хозяйственной деятельности хозяйствующих субъектов (обычно при обнаружении фактов недостач, растрат и хищений, а также в профилактических целях, например, при ревизии кассовых операций, финансовых вложений, отдельных видов материально-производственных запасов, основных средств и др.)

Выборочный контроль ограничивается проверкой выборочного круга проверяемых объектов (хозяйственных операций) по существующим методам их отбора, на основании чего по выборочной совокупности формируются выводы по результатам проведенного контроля. При надлежащей организации контрольной выборки выборочный контроль может оказаться достаточным и эффективным для надежной оценки фактического положения дел на проверенных участках финансово-хозяйственной деятельности хозяйствующего субъекта.

Для принятия оптимальных управленческих и финансовых решений необходимо знать свои затраты и в первую очередь разбираться в информации о производственных расходах. Анализ издержек помогает выяснить их эффективность, установить, не будут ли они чрезмерными, проверить качественные показатели работы, правильно установить цены, регулировать и контролировать расходы, планировать уровень прибыли и рентабельности производства.

Все перечисленные выше функции контроля и учета должны найти отражение в разрабатываемом Web-приложении.

## 1.4 Обусловленность выбора средств разработки и языков программирования

Реализовать автоматизированную систему учета и контроля производства мебели было решено посредством клиент-серверной технологии с использованием таких средств Web-разработки как: язык программирования PHP, предназначенный для разработки серверной части, система управления базами данных PostgreSQL, язык программирования клиентской части JavaScript, язык разметки гипертекста HTML, язык стилей CSS. Все перечисленные средства в совокупности дают весь необходимый арсенал для разработки профессиональных Web-приложений.

PHP – один из самых популярных скриптовых языков в области Web-программирования серверной части. На PHP написана львиная доля всех разрабатываемых в мире Web-продуктов.

В его составе имеется огромное количество встроенных средств для разработки Web-приложений, а также множество фреймворков, позволяющих упростить процесс разработки. Среди основных средств выделяются наличие POST и GET-методов передачи данных на сервер, записи переменных окружения Web-сервера в предопределённые массивы. Большим преимуществом PHP является также взаимодействие с большим количеством различных систем управления базами данных, таких как MySQL, PostgreSQL, Oracle, Microsoft SQL Server, Sybase, ODBC, mSQL и многих других. Посредством PHP осуществляется автоматизированная отправка HTTP-заголовков. PHP предоставляет возможность работы с HTTP-авторизацией, с cookies и сессиями, с локальными и удалёнными файлами, сокетами. С помощью PHP возможна обработка файлов, загружаемых на сервер и работа с XForms [5].

Сегодня PHP используется огромным количеством разработчиков.

Для реализации возможности хранения большого объема данных, их обработки, поиска, фильтрации, сортировки, обновления данных,

интерактивности системы необходимо разработать базу данных, предоставляющую все эти возможности пользователю. Для данного проекта была выбрана реляционная СУБД PostgreSQL. PostgreSQL - это свободно распространяемая объектно-реляционная система управления базами данных, наиболее развитая из открытых СУБД в мире и являющаяся реальной альтернативой коммерческим базам данных. Она широко распространена, быстра, удобна и проста в использовании, обладает такими качествами как надежность, производительность, расширяемость, простота использования. PostgreSQL поддерживает SQL, имеет богатый набор типов данных [6].

PostgreSQL оснащен также удобным графическим средством для формирования SQL запросов. А в его приложении PgAdmin предлагается удобный интерфейс для заполнения таблиц без использования программного кода.

Верстка страниц осуществлялась с помощью языка разметки гипертекста HTML, таблиц стилей CSS, языка JavaScript и его фреймворка JQuery, а также фреймворка PHP, используемого в качестве шаблона оформления и основного функционала, Yii. Все эти средства позволяют создать особенный современный, неповторимый стиль, яркий дизайн с визуальными и анимационными эффектами. Архитектура Web-приложения представлена на рисунке 1.2.



Рис. 1.2. Архитектурная схема Web-приложения

Таким образом, в ходе работы над теоретической частью проекта были изучены бизнес процессы учета и контроля производства продукции, структура мебельного предприятия ООО «Триллиант», определены функции и задачи, которые необходимо автоматизировать, выбраны программные средства их реализации.

## ГЛАВА 2 РАЗРАБОТКА WEB-ПРИЛОЖЕНИЯ УЧЕТА И КОНТРОЛЯ ПРОИЗВОДСТВА МЕБЕЛИ

### 2.1 Описание архитектуры клиент-сервер

Начнем с описания архитектуры клиент-сервер, которая была использована для построения приложения. Архитектура клиент-сервер представляет собой деление автоматизированной системы на три основных части:

- визуализация данных;
- прикладная часть;
- компонент управления базой данных [6].

В клиент-серверной архитектуре персональные компьютеры (клиенты) объединены в локальную сеть, в этой же сети находится и сервер баз данных, на котором содержатся общие для всех клиентов базы данные и СУБД. Клиенты получают возможность сами выполнять многочисленные приложения. Постраничная навигация, представление результатов, их печать и прочие рутинные возможности ложатся на клиентские машины. Однако, сама СУБД и базы данных полностью располагаются на сервере, который превосходит клиентские машины по вычислительным возможностям.

Так любая программа, выполняющая ту или иную задачу, должна обмениваться информацией с пользователем, осуществлять обработку этой информации в рамках автоматизации того или иного бизнес-процесса, и, наконец, хранить данные используемые в программе, на том или ином постоянном носителе.

В клиент-серверной архитектуре хотя бы один из трех ее компонентов полностью выполняется на другом компьютере, и взаимодействие между компонентами на разных компьютерах осуществляется через ту или иную

сетевую среду посредством передачи запросов на получение того или иного ресурса [7].

Взаимодействие клиентской и серверной частей приложения осуществляется через сеть – локальную или глобальную. При этом с точки зрения клиента и сервера взаимодействие осуществляется прозрачно, соответственно сетевой компонент здесь включает в себя совокупность необходимого сетевого оборудования, набор программных технологий, обеспечивающих передачу данных между узлами сети, а также собственно протокол или протоколы для обмена запросами и результатами их выполнения.

Компонент визуализации прикладной задачи осуществляет ввод информации пользователем с помощью тех или иных средств, а также вывод информации на экран и печать. Компонент визуализации для архитектуры клиент-сервер всегда исполняется на рабочем месте пользователя.

Компонент прикладной логики решает собственно ту или иную задачу, связанную с обработкой данных в той или иной предметной области. Этот компонент может быть распределен между клиентской и серверной частью различным образом в зависимости от применяемой модели.

Компонент хранения базы данных осуществляет физические операции, связанные с хранением данных, чтением информации из БД и записью. В архитектуре клиент-сервер это выполняется на сервере (см. рис. 2.1).



Рис 2.1. Место базы данных в архитектуре клиент-сервер

В архитектуре клиент-сервер выделяются несколько различных моделей приложения, в зависимости от распределения компонентов приложения между клиентской и серверной частями. Самым распространенным средством общения между клиентом и сервером в этом случае является SQL (структурированный язык запросов) - стандартный непроцедурный язык, ориентированный на обработку данных [7].

Архитектура клиент-сервер необходима также тогда, когда задача обеспечения целостности информации становится критической. Под критической мы здесь понимаем ситуацию, в которой цена ошибки в данных может быть сопоставима со стоимостью создания системы клиент-сервер. Прежде всего, это актуально для финансовых служб предприятий.

## **2.2 Описание использованных в проекте фреймворков**

Фреймворк это набор всевозможных библиотек или инструментов для быстрого решения повседневных рутинных задач. Чаще всего программисты используют одну из распространенных архитектур приложения (к примеру MVC) для разделения проекта на логические сегменты, модули [8].

Model-View-Controller (MVC, «Модель-Представление-Контроллер», «Модель-Вид-Контроллер») – схема разделения данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер – таким образом, что модификация каждого компонента может осуществляться независимо.

- Модель (Model) предоставляет данные и реагирует на команды контроллера, изменяя свое состояние.
- Представление (View) отвечает за отображение данных модели пользователю, реагируя на изменения модели.
- Контроллер (Controller) интерпретирует действия пользователя, оповещая модель о необходимости изменений

Структурная схема MVC Yii представлена на рис. 2.2.

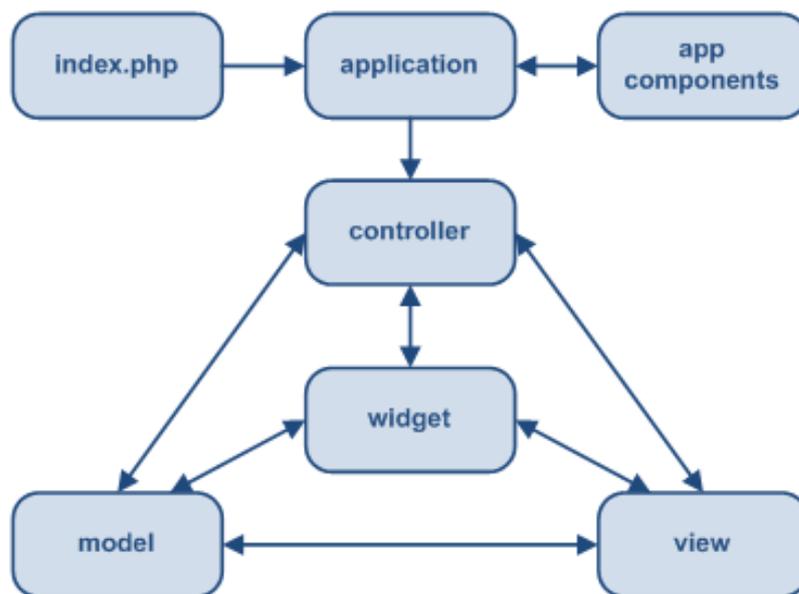


Рис. 2.2. Статическая структура приложения Yii

Главная цель фреймворка - предоставить разработчику удобную среду для проекта с большим и хорошо расширяемым функционалом.

Сам фреймворк предлагает нам уже встроенные классы для: работы с базой данных, создания функциональных форм, валидации, логирования и др. Все эти классы можно легко использовать в проектах, при этом их подключение и использование будет максимально простым [8].

Диаграмма, представленная на рисунке 2.3, описывает типичную последовательность процесса обработки пользовательского запроса приложением.

1. Пользователь осуществляет запрос посредством URL `http://www.example.com/index.php?r=post/show&id=1`, и Web-сервер обрабатывает его, запуская скрипт инициализации `index.php`.

2. Скрипт инициализации создает экземпляр приложения и запускает его на выполнение.

3. Приложение получает подробную информацию о запросе пользователя от компонента приложения `request` (см. рис. 2.3).

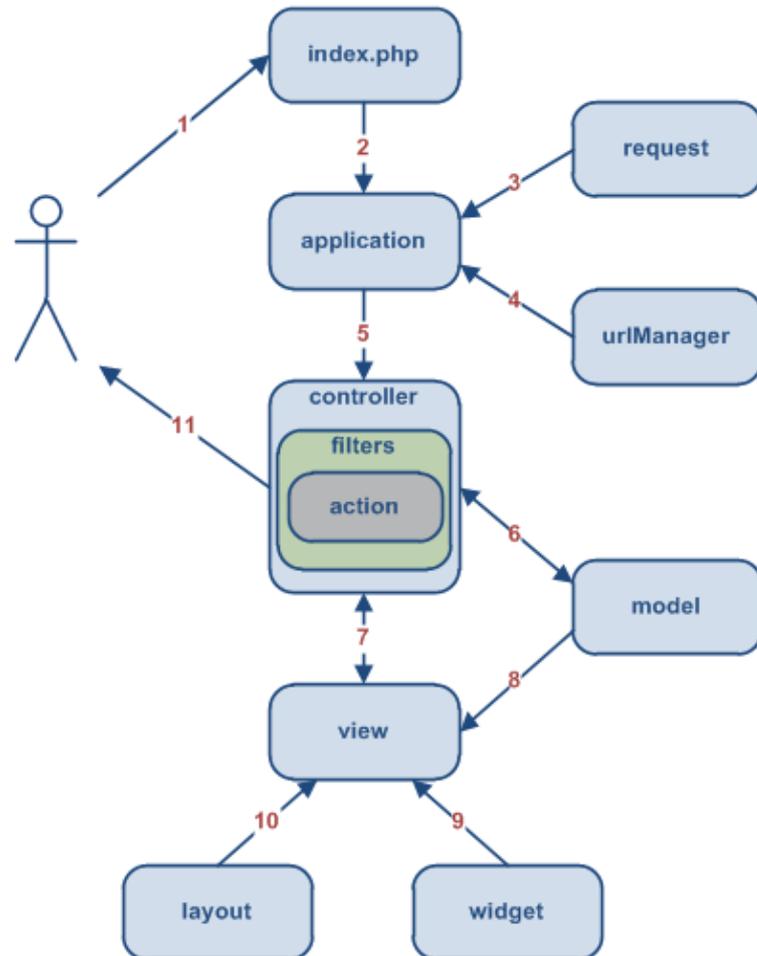


Рис. 2.3. Типичная последовательность работы приложения Yii

4. Приложение определяет запрошенный контроллер и действие при помощи компонента urlManager. В данном примере контроллером будет post, относящийся к классу PostController, а действием – show, суть которого определяется контроллером [9].

5. Приложение создаёт экземпляр запрашиваемого контроллера для дальнейшей обработки запроса пользователя. Контроллер определяет соответствие действия show методу actionShow в классе контроллера. Далее создаются и применяются фильтры (например, access control, benchmarking), связанные с данным действием, и, если фильтры позволяют, действие выполняется.

6. Действие считывает из базы данных модель Post с ID равным 1.

7. Действие подключает представление `show`, передавая в него модель `Post`.
8. Представление получает и отображает атрибуты модели `Post`.
9. Представление подключает некоторые виджеты.
10. Сформированное представление вставляется в макет страницы.
11. Действие завершает формирование представления и выводит результат пользователю.

Большим плюсом Yii является отличная поддержка расширений, которая позволяет нам подключать сторонние библиотеки [9].

Еще один из плюсов - структурирование архитектуры вашего приложения. Фреймворк содержит в себе перечень стандартных папок, в которых предполагается что вы будете размещать необходимые части вашего приложения (отдельная папка для тем, контроллеров, расширений, конфигурационных файлов). Это большой плюс, потому что позволяет вам поддерживать все файлы в порядке на нужных местах.

Yii – это фреймворк, написанный на языке PHP. Главным плюсом Yii является поддержка ООП, скорость работы и, конечно же, техническая поддержка от разработчиков. Фреймворк включает в себя большой набор библиотек, которые помогут в создании полноценного Web-приложения, отвечающего всем современным стандартам (интегрированное использование Ajax, встроенная поддержка интернационализации приложения, простой инструмент работы с базой данных). Тот код, который занимал бы у вас 100 строчек чистого php кода, может быть сокращен до десяти благодаря встроенным методам фреймворка. Yii является бесплатным ПО и распространяется под лицензией «new BSD» [9].

Еще один фреймворк, который был использован в проекте – это JQuery. JQuery является просто произведением инженерного искусства. Она умело покрывает достаточно широкий диапазон повседневных функций и предоставляет при этом удобный API для расширений, с помощью которых можно добавить любую другую функциональность. Абстрактность в ней

заложена на уровне ядра — речь идет о выборе DOM-элементов — и она извлекает из него максимум пользы. И что очень важно, использование этой библиотеки подразумевает следование хорошему стилю в программировании и хорошо сочетается с другими частями JavaScript-кода [10].

Главным моментом в создании хорошего JavaScript-кода для дальнейшего использования является тщательное управление пространством имен. В JavaScript существует единое глобальное пространство имен (объект `window`), и многие программисты (и даже некоторые библиотеки) засоряют его безо всякой надобности. Более благоразумные разработчики сводят к минимуму свое вторжение в это пространство, используя некоторые методы, например, модульную модель [10].

jQuery вводит единственный объект в глобальное пространство имен — функцию/объект `jQuery`. Все остальное — это либо непосредственное свойство `jQuery`, либо метод объекта, возвращаемого вызовом функции `jQuery` [11].

Что можно сказать об улучшениях языка? Большинство библиотек предоставляют некоторое подобие функций отображения, фильтрации и обрезания, которые, к несчастью, отсутствуют в тех движках JavaScript, которые включены в большинство браузеров. Некоторые библиотеки напрямую расширяют встроенные в JavaScript классы `String` и `Array`, но также не до конца безопасно. `String.prototype` и `Array.prototype` являются самостоятельными глобальными пространствами имен, и добавление в них каких-либо свойств влечет опасность коллизий, связанных с использованием одних и тех же имен переменных в разных контекстах [11].

В проекте используются анимационные эффекты для элементов с использованием jQuery. Это, например, двойной эффект с наложением для текста пунктов меню, выдвигание картинки при наведении указателя и другие.

## 2.3 Описание базы данных

В состав любой автоматизированной системы должна входить база данных. Включение базы данных позволяет пользователю редактирование информации, ее хранение, отображение в удобном виде, организация любых запросов пользователей.

В широком смысле слова база данных – это совокупность сведений о конкретных объектах реального мира в какой-либо предметной области. Под предметной областью принято понимать часть реального мира, подлежащего изучению для организации управления и, в конечном счете, автоматизации, например предприятие [12].

В современной технологии баз данных предполагается, что создание базы данных, ее поддержка и обеспечение доступа пользователей к ней осуществляются централизованно с помощью специального программного инструментария - системы управления базами данных [13].

Система управления базами данных (СУБД) - это комплекс программных и языковых средств, необходимых для создания баз данных, поддержания их в актуальном состоянии и организации поиска в них необходимой информации.

В проекте использовалась реляционная модель данных. Реляционная модель ориентирована на организацию данных в виде двумерных таблиц. Каждая реляционная таблица представляет собой двумерный массив и обладает следующими свойствами [14]:

- каждый элемент таблицы – один элемент данных;
- все столбцы в таблице однородные, т.е. все элементы в столбце имеют одинаковый тип (числовой, символьный и т.д.) и длину;
- каждый столбец имеет уникальное имя;
- одинаковые строки в таблице отсутствуют;
- порядок следования строк и столбцов может быть произвольным.

Отношения представлены в виде таблиц, строки которых соответствуют кортежам или записям, а столбцы – атрибутам отношений, доменам, полям [15].

Первичным ключом называется поле, однозначно определяющее запись. Ключ будет являться составным, если записи однозначно определяются значениями нескольких полей.

Чтобы связать две реляционные таблицы, необходимо ключ родительской таблицы ввести в состав ключа дочерней таблицы (возможно совпадение ключей); в противном случае нужно ввести в структуру первой таблицы внешний ключ – ключ второй таблицы [15].

Для разработки базы данных в проекте использовалась СУБД PostgreSQL. Составим физическую и на ее основе логическую модели базы данных – структуру таблиц и связей между ними. Логическая модель базы данных представлена на рис. 2.4.

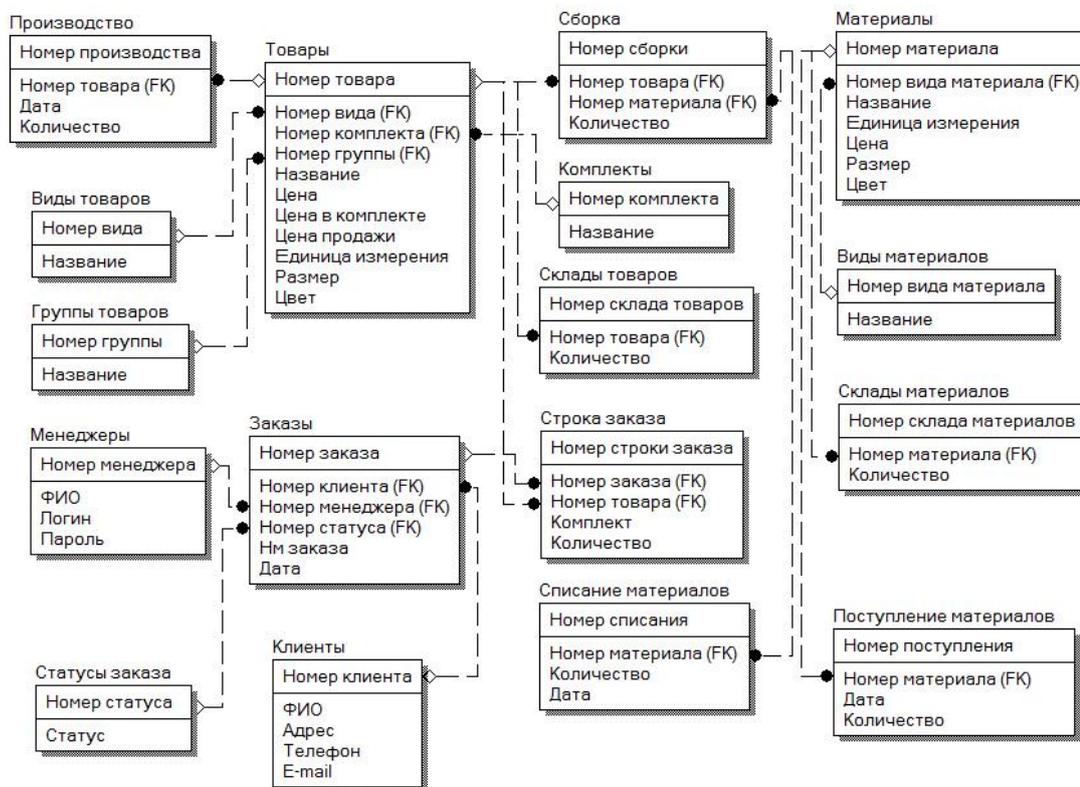


Рис. 2.4. Логическая модель базы данных

В данном проекте использовались связанные между собой таблицы как в отношении один-ко-многим, так и много-ко-многим. Соответствующая физическая модель приведена на рис. 2.5.

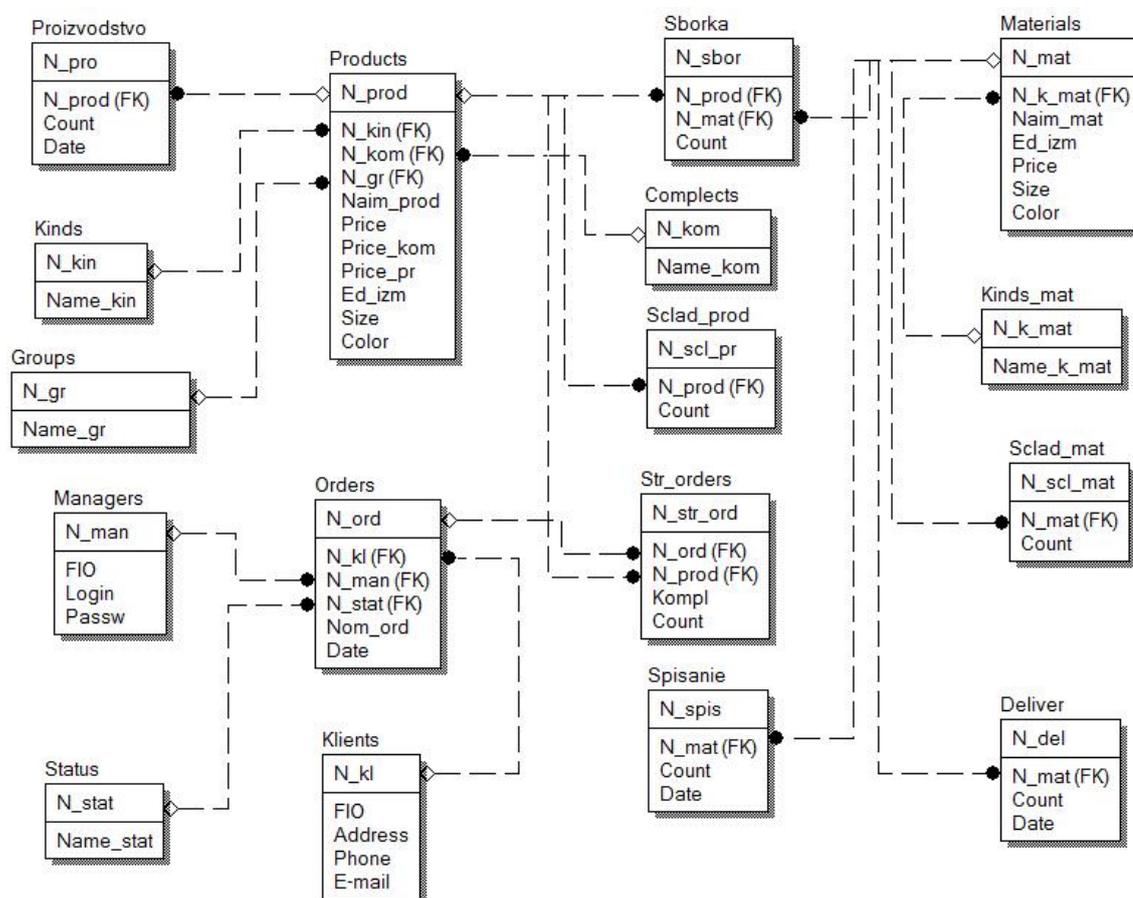


Рис. 2.5. Физическая модель базы данных

В таблице ГРУППЫ\_ТОВАРОВ хранится информация о том, какие группы товаров производятся на предприятии, например, кухни, детские, прихожие и тд., в таблице ВИДЫ\_ТОВАРОВ перечислены виды мебели – шкафы, кресла, кровати, диваны и тд. В таблице КОМПЛЕКТЫ будут храниться данные об изготавливаемых комплектах, например, стенка Джорджия, детская Радуга и др.

В базе данных также хранится информация об основных и вспомогательных материалах, из которых изготавливаются изделия, о заказчиках и заказах, о поступлении и списании материалов, о складах продукции.

Таблицы связаны отношениями один-ко-многим или многие-ко-многим. Использование отношения один-ко-многим можно рассмотреть на примере родительской таблицы ВИДЫ\_МАТЕРИАЛОВ и дочерней таблицы МАТЕРИАЛЫ, так как одному виду соответствует несколько материалов, но один материал принадлежит только одному виду. Первичный ключ родительской таблицы номер\_вида связан с вторичным (или внешним) ключом дочерней таблицы [16].

Отношение многие-ко-многим рассмотрим на примере таблиц ТОВАРЫ, ЗАКАЗЫ и СТРОКА\_ЗАКАЗА. Так как один товар может входить в несколько заказов и в каждый заказ входит много товаров, таблицы ТОВАРЫ и ЗАКАЗЫ необходимо связать отношением много-ко-многим. Для правильной организации связи необходимо создать дополнительную связующую таблицу СТРОКА\_ЗАКАЗА, в которой будет храниться информация о том, какой товар, в каком количестве и когда был заказан.

В логической модели отсутствует также таблица СБОРКА. Так как таблицы ТОВАРЫ и МАТЕРИАЛЫ связаны отношением много-ко-многим (один товар состоит из многих материалов и один материал используется для изготовления многих товаров), нужна таблица связи СБОРКА.

Создание таблиц, редактирование, запросы – все реализуется средствами структурированного языка запросов SQL.

Создание базы данных и таблиц осуществляется с помощью команды CREATE. На рис. 2.6 показано создание базы данных.

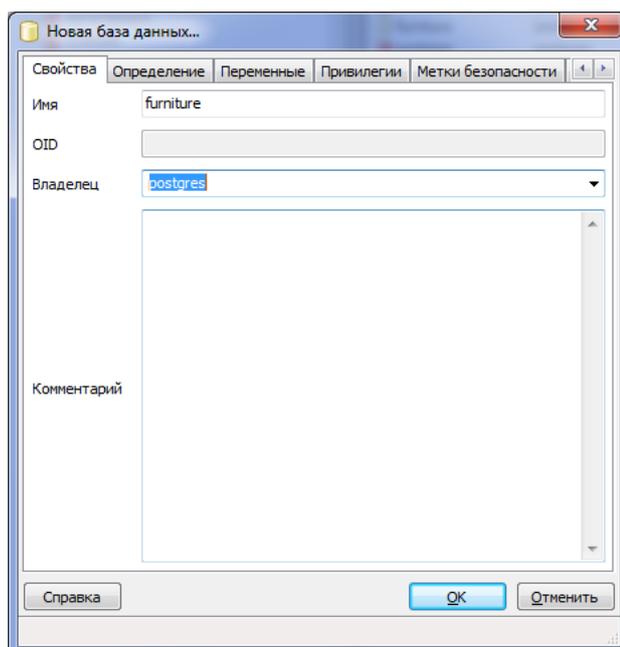


Рис. 2.6. Создание базы данных

На рис. 2.7 показано создание базы данных с помощью языка SQL.

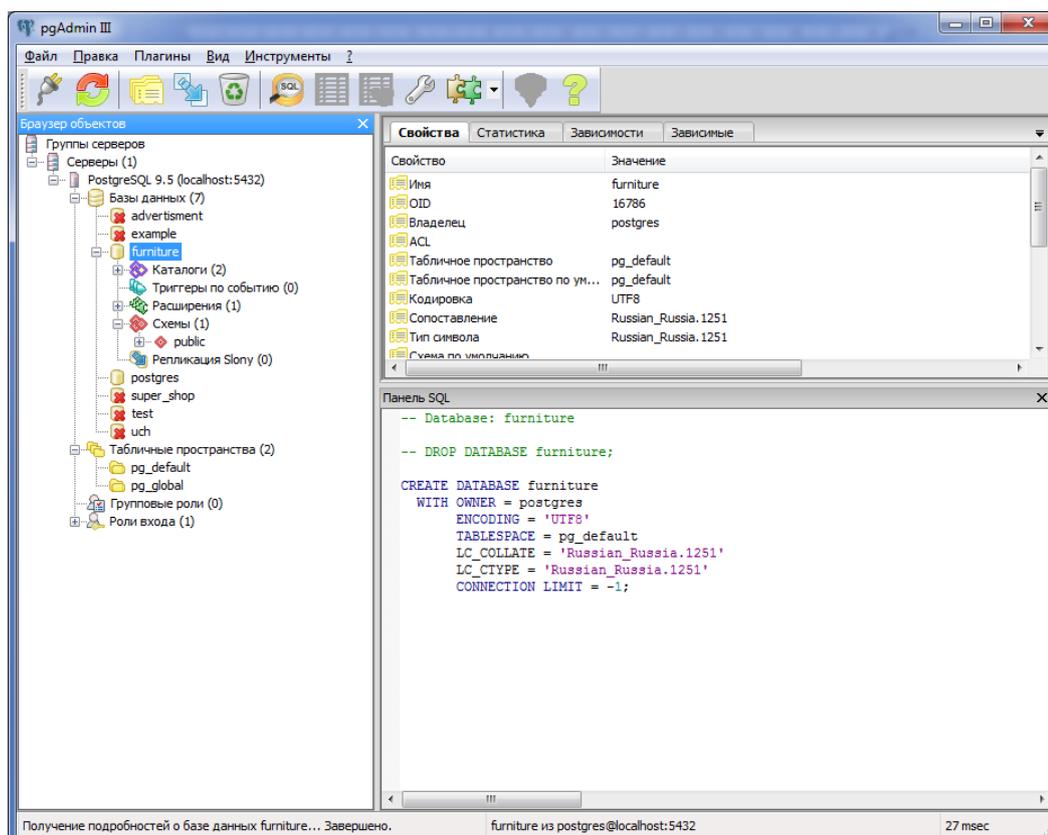


Рис. 2.7. Создание базы данных с помощью SQL

С помощью команды `create table` создаются таблицы базы данных. На рисунке 5 показано создание таблицы МАТЕРИАЛЫ, в которой содержатся как первичный, так и вторичный ключ. Для того, чтобы записи нумеровались автоматически используется специальный тип `serial primary key`. Вторичный ключ создается с помощью предложения `foreign key` с указанием таблицы и ключа связи. Здесь в качестве родительской таблицы указана таблица ВИДЫ\_МАТЕРИАЛОВ, а ключом связи является `n_r_mat`. Организация ссылочной целостности происходит с помощью предложения `ON UPDATE CASCADE, ON DELETE CASCADE`. Таким образом, при удалении или изменении записи в родительской таблице удаляются или изменяются соответствующие записи в дочерней таблице. На рис. 2.8 приведен SQL код, выполняющий запрос на создание таблицы [18].

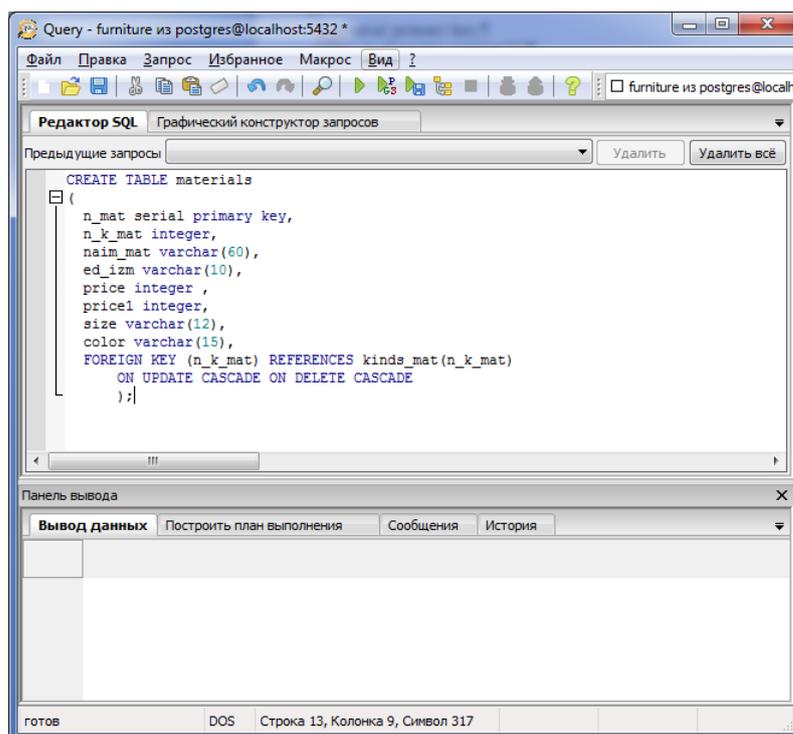


Рис. 2.8. Реализация связи один-ко-многим

Чтобы заполнить таблицы данными, используется команда `INSERT TABLE`, а в приложении PgAdmin предлагается удобный интерфейс для заполнения таблиц без использования программного кода (см. рис. 2.9).

	n_mat [PK] serial	n_k_mat integer	naim_mat character varying(60)	ed_izm character varying(10)	price integer	price1 integer	size character varying(12)	color character varying(15)
1	1	1	древестружечная плита	лист	50	50	3500x1750x10	венге
2	2	1	древестружечная плита 1	лист	60	60	3500x1750x10	дуб
3	3	1	древестружечная плита 2	лист	60	60	3500x1750x10	орех
4	4	1	доска дуб обрезная сухая	шт	50	50	1500x2450	орех
5	5	1	ПЕХ для мембранного прессования	кв.м	150	150	250x1500	металлик
6	6	1	зеркало фантазия	кв.м	300	300	1000x1000	
7	7	1	стекло с тонирующими пленками	кв.м	200	200	1200x550	серый
8	8	1	ПЕХ пленка для фасадов	кв.м	7	7	500x5000	белый
9	9	1	древестружечная плита 3	лист	50	50	3500x1750x10	металлик
10	10	1	МДФ ламинированная односторонняя	лист	70	70	1500x2500	вишня
11	11	1	доска бук обрезная сухая	шт	50	50	1500x2500	вишня
12	12	1	шит мебельный сращенный	лист	500	500	5000x4000	дуб
13	13	1	кромка клеевая	п.м.	2	2	35x450	ольха
14	14	1	ПЕХ пленка для фасадов однотонная	кв.м	100	100	4500x3500	ольха
15	15	1	стекло метакриловое полупрозрачное	кв.м	250	250	1000x1200	серый
16	16	1	ПЕХ пленка для фасадов матовая древесная	кв.м	5	5	500x5000	белый
17	17	1	доска сосна обрезная сухая	шт	20	20	1500x1200	кедр
18	18	1	ПЕХ пленка для фасадов страйп	кв.м	5	5	1400x520	прозрачный
19	19	1	ABS кромка фоновая шагрень	п.м.	90	90	5000x20	белый мрамор
20	20	1	стекло матовое с защитной пленкой	кв.м	9	9	1800x2000	серый
21	21	1	шит мебельный кедр сращенный	лист	5	5	500x700	кедр
22	22	1	МДФ ламинированная белая	лист	14	14	500x700x1200	белый мрамор
23	23	1	ДВП декорированная	лист	8	8	2000x3000	ольха
24	24	1	ДВП декорированная	лист	8	8	2000x3000	вишня
25	25	2	лак алкидный быстросохнущий	литр	5	5		вишня
26	26	2	лак алкидный быстросохнущий	литр	5	5		кедр
27	27	2	лак алкидный быстросохнущий	литр	5	5		белый мрамор
28	28	2	лак алкидный быстросохнущий	литр	5	5		ольха
29	29	2	шуруп 2,5x10	шт	2	2	2,5x10	
30	30	2	шуруп	шт	2	2	2,3x7	

Рис. 2.9. Заполнение таблиц данными

Для редактирования данных таблицы и их удаления используются команды UPDATE и DELETE. В программе все имеющиеся данные редактируются с помощью данных команд. В листинге 2.1 приведен пример запроса на редактирование записи о клиентах [18].

Листинг 2.1 Запрос на редактирование записи таблицы

```

$q3="update clients set
f_k="".$f."" ,
address="".$a."" ,
tel="".$tel.""
where n_kl="".$_GET['in'].";";

```

Выборка данных из таблиц осуществляется с помощью SQL команды SELECT. Данные запросы можно очень быстро реализовать с помощью графического конструктора запросов PgAdmin (рис. 2.10).

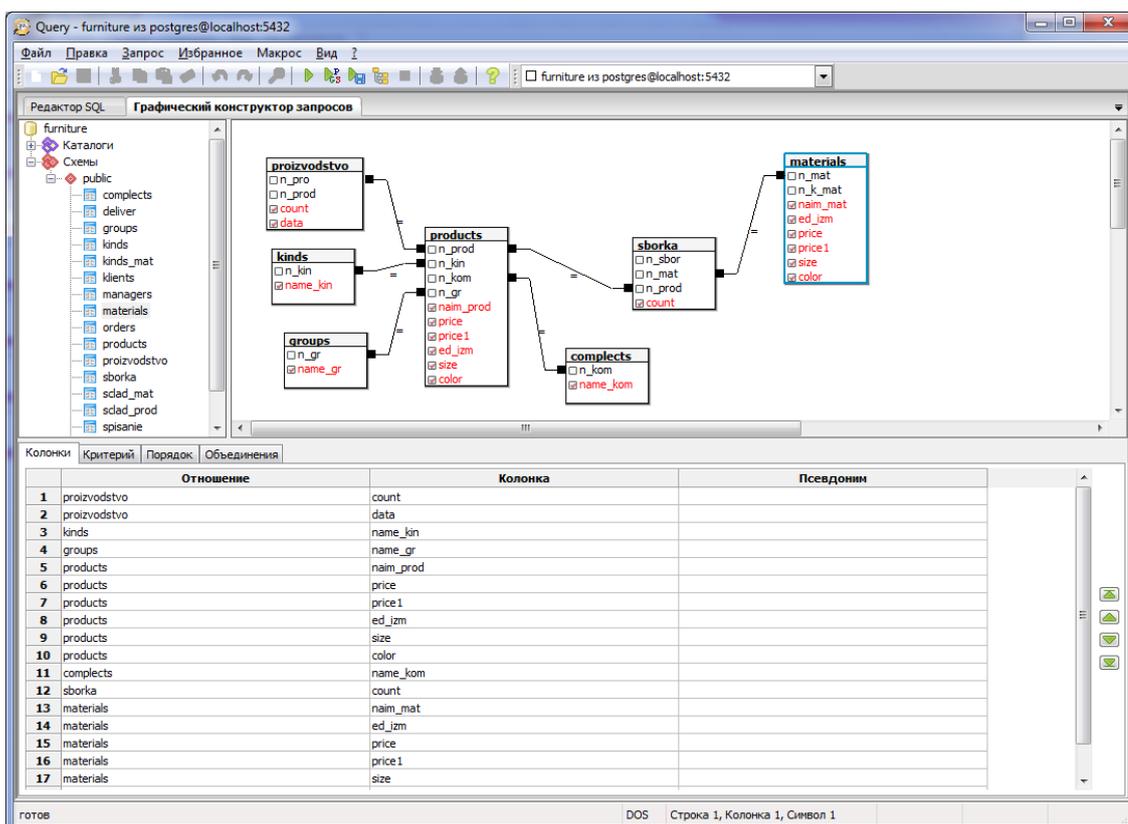


Рис. 2.10. Конструирование запросов на выборку

Приведем примеры различных запросов. Выборка данных из нескольких связанных таблиц с условием совпадения по похожим данным представлена ниже [19].

Листинг 2.2 Выборка данных из нескольких связанных таблиц

```

$result=pg_query($dbconn,"SELECT
    name_kin,naim_prod,      products.size,      products.color,naim_mat,
    materials.ed_izm,
    materials.price, sborka.count,products.n_prod
FROM kinds, products, materials, sborka
WHERE
    kinds.n_kin = products.n_kin AND
    products.n_prod = sborka.n_prod AND
    materials.n_mat = sborka.n_mat and
    name_kin like ".$di."%" and naim_prod like ".$de."%"
    and naim_mat like ".$du."%';");

```

Далее показаны выборки из нескольких таблиц с выборкой по точному совпадению.

Листинг 2.3 Выборки из нескольких таблиц с выборкой по точному совпадению

```

$result=pg_query($dbconn,
    "SELECT
    products.n_prod,
    products.naim_prod,
    products.size,
    products.color,
    products.price,
    products.price1
FROM products
WHERE
    products.n_prod=0 "._SESSION['str']."
    ");
$result=pg_query($dbconn,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
FROM orders, klients, managers, status
WHERE
    klients.n_kl = orders.n_kl AND
    managers.n_man = orders.n_man AND
    status.n_stat = orders.n_stat and
    klients.n_kl="._GET["it1"].".";"); [20]

```

## 2.4 Описание этапов разработки

Работа по созданию приложения проводилась в несколько этапов:

- 1) Поиск и сбор информации о функциях, структуре и задачах работы отдела контроля и учета производства мебели, систематизация данной информации;
- 2) Разработка логической модели базы данных, структуры таблиц, входящих в нее, связей между таблицами, ключей и индексов;
- 3) Физическое создание базы данных, таблиц и связей, заполнение таблиц данными;
- 4) Создание запросов;

- 5) Разработка логической структуры системы автоматизации работы менеджера и определение путей навигации по ее составным частям;
- 6) Определение и создание стиля оформления пользовательского интерфейса, планирование мест взаимного расположения элементов на странице, их размеров для удобства и простоты использования данного ресурса;
- 7) Верстка страниц с помощью HTML, создание стилей CSS, подготовка и внедрение фреймов, изображений, таблиц, списков, гиперссылок, форматирование текста;
- 8) Алгоритмизация и программирование основных функций программы на языках Java Script и PHP;
- 9) Тестирование и отладка программы.

Первые четыре пункта этапов разработки были описаны подробно в теоретическом разделе.

Структурная схема приложения представлена на рис. 2.11.



Рис. 2.11. Структурная схема приложения

Дизайн и стиль оформления были реализованы с помощью HTML, CSS, функций Java Script, библиотеки JQuery и фреймворка Yii [21, 22].

Титульная страница содержит сменяющие друг друга во времени слайды. Это реализуется с помощью следующей функции JQuery:

#### Листинг 2.4 Функция JQuery смены слайдов

```
$(function () {
    $("#slider").responsiveSlides({
        auto: true,
        speed: 500,
        namespace: "callbacks",
        pager: true,
    });
});
```

При наведении указателя на пункты меню создается эффект копирования текста, копия, плавно поворачиваясь, появляется и исчезает (см. приложение). Это реализуется с помощью CSS трансформации при наведении указателя [22].

#### Листинг 2.5 Эффект копирования текста

```
.top-nav ul li a::before {
    color: #b66ca9;
    content: attr(data-hover);
    position: absolute;
    opacity: 0;
    text-shadow: 0 0 1px rgba(255,255,255,0.3);
    -webkit-transform: scale(1.1) translateX(10px) translateY(-10px)
    rotate(4deg);
    -moz-transform: scale(1.1) translateX(10px) translateY(-10px)
    rotate(4deg);
    transform: scale(1.1) translateX(10px) translateY(-10px) rotate(4deg);
    -webkit-transition: -webkit-transform 0.3s, opacity 0.3s;
    -moz-transition: -moz-transform 0.3s, opacity 0.3s;
    transition: transform 0.3s, opacity 0.3s;
    pointer-events: none;
}
```

Прозрачность и смена цвета происходит с помощью функций Java Script, которые срабатывают при наведении указателя и при выходе указателя за границу элемента [23]:

Листинг 2.6 Прозрачность и смена цвета с функциями Java Script

```
function f1()
  {if (g!=1) {
    document.getElementById("de1").style.backgroundColor="white";
    document.getElementById("de1").style.color="#a00ba2";
    document.getElementById("de1").style.opacity=1;
  };
}
```

С помощью стилей CSS реализуется механизм всплывающих окон:

Листинг 2.7 Механизм всплывающих окон

```
#parent_popup {
  height: 100%;
  position: fixed;
  width: 100%;
  z-index: 100;
  top: 0;
  left: 0;
}
#popup {
  background: url(J0143754.GIF);
  height: 635px;
  position: fixed;
  top: 0%;
  left: 15%;
  color: #a00ba2;
  width: 70%;
  border: #a00ba2 1px solid;
} [24, 25]
```

Сессии – это механизм, позволяющий однозначно идентифицировать браузер и создающий для этого браузера файл на сервере, в котором хранятся переменные сеанса (см. приложение).

В данном проекте сессии используются для авторизации на сайте, а также для хранения переменных, значения которых необходимо помнить при переходе на другую страницу. Так как различным группам пользователей предоставляются свои права доступа в систему, необходимо помнить вошедшего все время, пока он находится on-line. Также для добавления товаров в заказ пользователю необходимо каждый раз открывать разные окна и возвращаться в заказ. При этом программа должна помнить все выбранные пользователем товары [26].

#### Листинг 2.8 Включение сессии на каждой странице

```
<?php
  $session = Yii::$app->session;
  $session['nc'] =0;
  $session['str']="";
  ?>
```

Далее в программе используются данные переменные в любом окне и стираются тогда, когда будет включена команда unset.

После ввода логина и пароля и нажатия на кнопку «Войти» система в цикле просматривает таблицу базы данных «Менеджеры» и проверяет, есть такие логин и пароль в базе. Если есть, то ему предоставляется доступ в систему. Если пользователь вошел под логином администратора, ему предоставляется доступ к дополнительным функциям редактирования данных [27].

#### Листинг 2.9 Проверка логина и пароля в базе

```
if (isset($_POST['edt1']) && isset($_POST['edt2']))
{
  $result=pg_query($connection,"select * from managers;");
  $r=pg_num_rows($result);
  $u=0;
  for ($i=0; $i < $r; $i++)
  { $rows=pg_fetch_row($result,$i);
    if ($_POST['edt1']==$rows[2] && $_POST['edt2']==$rows[3])
    {
      $u=1;
    }
  }
}
```

Продолжение. Листинг 2.9 Проверка логина и пароля в базе

```

        $_SESSION['n_man']=$rows[0];
        $_SESSION['log']=$rows[1];
    }
}
if ($u==0)
{
    header("Location: ".$_SERVER['HTTP_REFERER']);
}
}

```

В проекте часто используются ифреймы. С помощью IFRAME можно в произвольном месте страницы выделить прямоугольную область со своей полосой прокрутки и загружать в нее внешний \*.html документ (см. приложение).

В данном проекте используются IFRAME для того, чтобы перезагружалась не полностью вся страница при обращениях PHP к серверу, а только ее часть – IFRAME. Например, при поиске данных по критериям необходимо отправить запрос серверу при возникновении события нажатия клавиши клавиатуры. Если при этом будет перезагружаться вся страница, мы не увидим полученного результата, так как вернемся на исходную страницу. С использованием же ифрейма перезагрузится только часть страницы с таблицей, а основная страница вместе со всей информацией останется на экране [28].

Чтение данных из таблиц и вывод их на экран производится с помощью Java Script и PHP. Из таблиц базы данных с помощью оператора цикла с предусловием берутся записи, заносятся в массив, а затем выводятся в соответствующие компоненты формы (см. приложение). Для связи PHP переменных с переменными Java Script были созданы невидимые компоненты формы, из которых данные записывались в массив и обрабатывались [29, 30].

## Листинг 2.10 Чтение данных из таблиц и вывод их на экран

```

$result=pg_query($connection,
    "SELECT
    products.n_prod,
    products.naim_prod,
    products.size,
    products.color,
    products.price,
    products.price1
FROM
    products
WHERE
    products.n_prod=0 ".$_SESSION['str']."
    ");
$result=pg_query($dbconn,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
    FROM orders, klients, managers, status
    WHERE
        klients.n_kl = orders.n_kl AND
        managers.n_man = orders.n_man AND
        status.n_stat          =          orders.n_stat          and
    klients.n_kl=".$_GET["it1"].".";");
<table>
    <tr>
        <td class="td1">№</td>
        <td class="td1">Наименование</td>
        <td class="td1">Цена</td>
        <td class="td1">Цена в компл.</td>
        <td class="td1">Размер</td>
        <td class="td1">Цвет</td>
        <td class="td1">Выбрать</td>
    </tr>
    $i=0;
    while($i<$n)
    { echo '<form method=get target="ifrm11" action="is1.php"
    name="ki'.Yii::t($i+1).'" id="ki">';
    echo '<tr>';
    echo '<td>'.Yii::t($i+1).'\</td>';
    echo '<td><div id="name_kin'.Yii::t
    ($i+1).'">'.Sa[$i]['name_kin'].'\</div></td>';
    echo '<td><div
    id="naim_prod'.Yii::t($i+1).'">'.Sa[$i]['naim_prod'].'\</div></td>';
    echo '<td><div
    id="size'.Yii::t($i+1).'">'.Sa[$i]['size'].'\</div></td>';

```

Продолжение. Листинг 2.10 Чтение данных из таблиц и вывод их на экран

```

        echo '<td><div
id="color'.Yii::t($i+1)."'>'.Sa[$i]['color'].'</div></td>';
        echo '<td><div
id="naim_mat'.Yii::t($i+1)."'>'.Sa[$i]['naim_mat'].'</div></td>';
        echo '<td><div
id="ed_izm'.Yii::t($i+1)."'>'.Sa[$i]['ed_izm'].'</div></td>';
        echo '<td><div
id="price'.Yii::t($i+1)."'>'.Sa[$i]['price'].'</div></td>';
        echo '<td><div
id="count'.Yii::t($i+1)."'>'.Sa[$i]['count'].'</div></td>';
        echo '<td><input type="submit" id="tut'.Yii::t($i+1)."'
value='.Yii::t($i+1).'
        style="background-color:white;
        width:90px;height:19px;color:#d86ecd;cursor:pointer;"
        onclick="gr_s('.Yii::t($i+1).')"></td>';
        echo '</tr>';

```

В программе использовались математические формулы для расчетов себестоимости продукции, итоговой суммы по заказам и др. Формула расчета себестоимости продукции приведена ниже (2.1). Чтобы рассчитать себестоимость, нужно просуммировать количество, умноженное на цену каждого материала, который входит в состав продукта. Ко всей сумме нужно прибавить все дополнительные затраты, связанные с производством продукции. Это может быть электроэнергия, зарплата сотрудникам, амортизация, страховые отчисления и другие.

$$S = \sum_i c_i \cdot p_i + D_i, \quad (2.1)$$

где  $S$  – себестоимость  $i$ -го продукта,  $c$  – количество израсходованного на производство данного продукта материала,  $p$  – стоимость единицы данного материала,  $D$  – дополнительные затраты, связанные с производством.

Чтобы рассчитать итоговую сумму по данному заказу, нужно просуммировать произведения цен на количество всех товаров, входящих в данный заказ (2.2).

$$S = \sum_i c_i \cdot p_i, \quad (2.2)$$

где  $S$  – общая сумма по заказу,  $c$  – количество товара, входящего в заказ,  $p$  – цена товара. Программа работает таким образом, что если продукция заказана в комплекте, то на нее действует скидка (см. приложение). Скидка рассчитывается как 10 процентов от стоимости товара. Таким образом, цена товара со скидкой будет рассчитываться по формуле (2.3).

$$p_d = p - 0,1 \cdot p, \quad (2.3)$$

где  $p_d$  – цена со скидкой,  $p$  – цена товара.

Реализация расчетов по приведенным выше формулам происходит в программе либо с помощью SQL запросов, либо с помощью функций JavaScript. Либо непосредственным обращением к таблице базы данных с помощью алгоритмов цикла и условия (см. приложение).

## **2.5 Пользовательский интерфейс и руководство работы с программой**

После запуска приложения открывается страница с главным меню с возможностями входа в систему администратора и менеджера. Привилегии доступа разграничены в зависимости от возлагаемых на них функций. Так менеджер может просмотреть всю необходимую информацию, провести производственный учет, контроль, необходимые расчеты, оформить заказ,

списать материал или осуществить сборку. Администратор же помимо этого может редактировать данные. На рис. 2.12 показано главное окно программы [31].

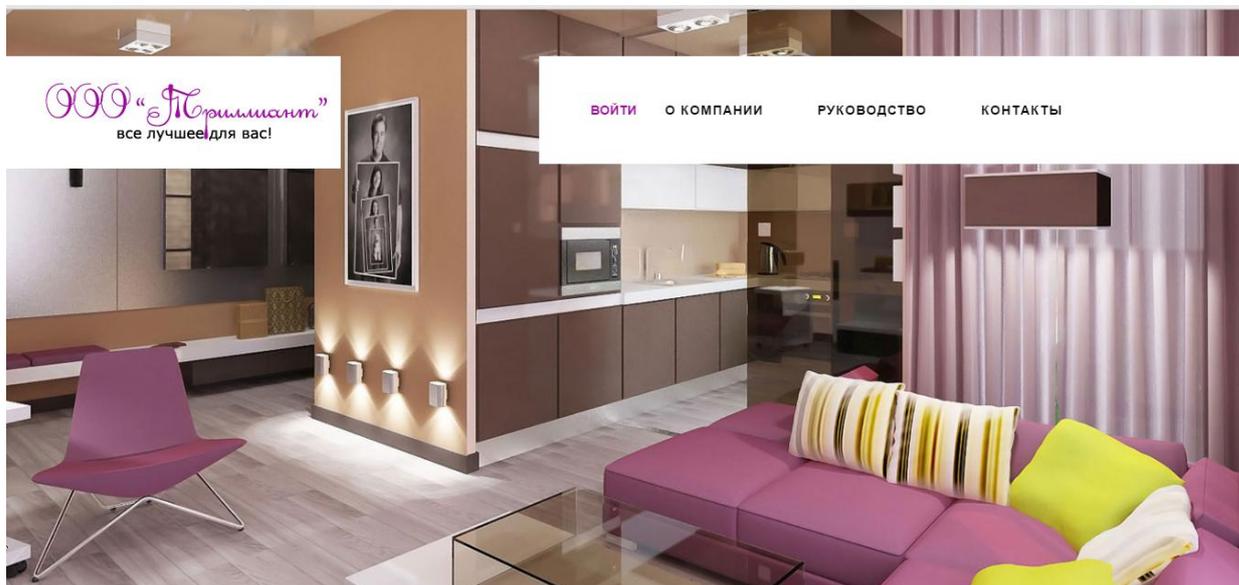


Рис. 2.12. Главное окно программы

На следующем рис. 2.13 представлено окно авторизации пользователя. Войти в систему имеет право только зарегистрированный менеджер. Регистрировать сотрудников может только администратор (см. приложение).

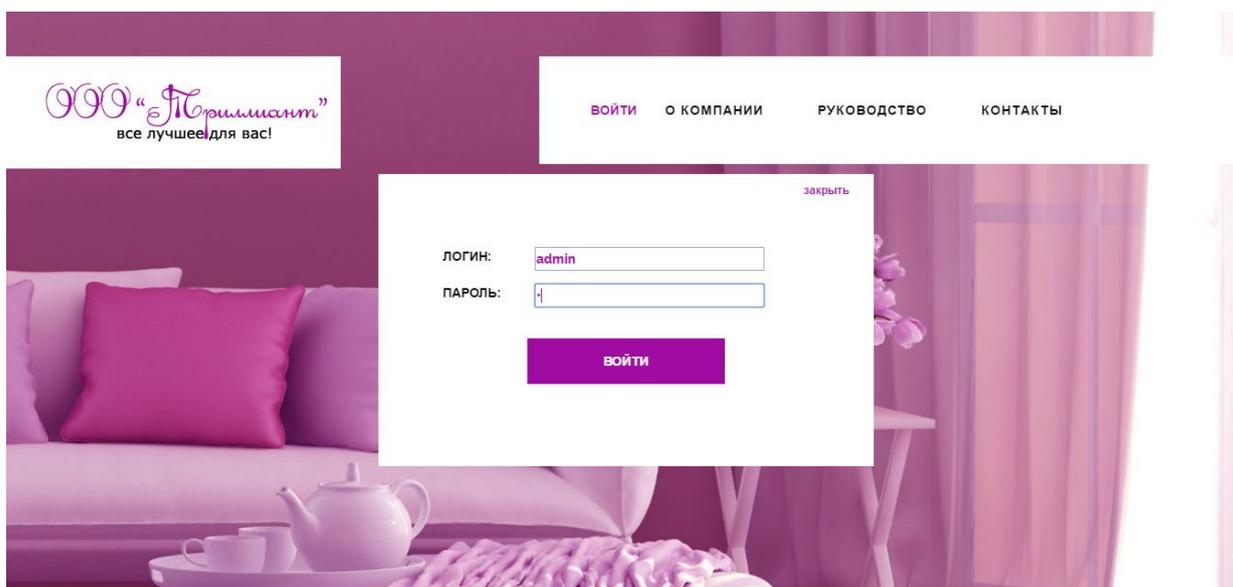


Рис. 2.13. Авторизация пользователя

После входа в систему открывается рабочее окно менеджера с вертикальным и горизонтальным меню. Горизонтальное меню служит, главным образом, для доступа к справочной информации (рис. 2.14).

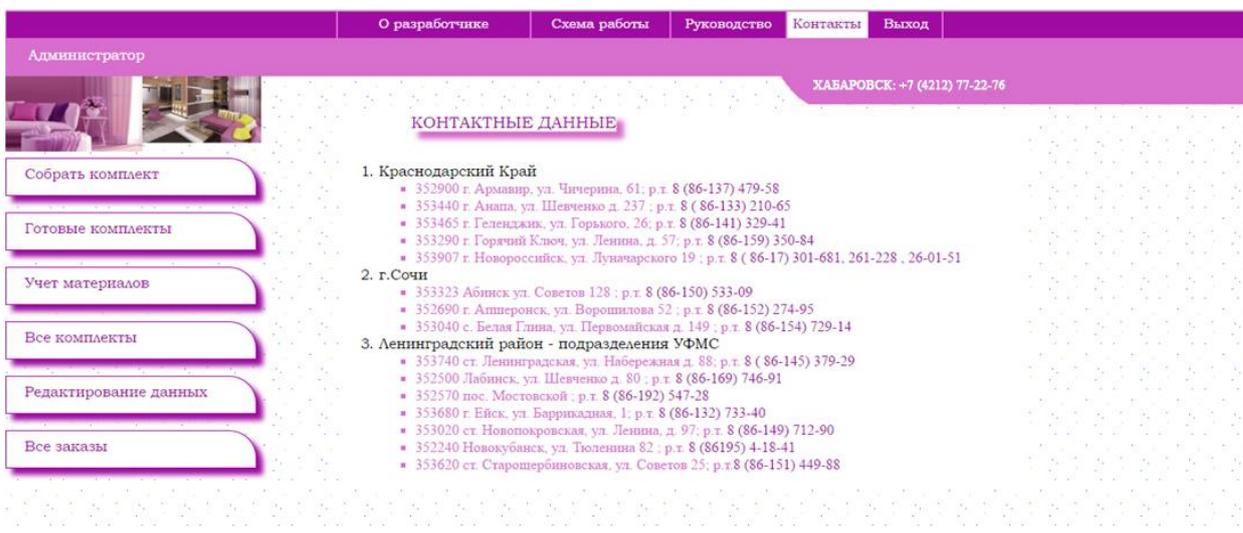


Рис. 2.14. Пример функции горизонтального меню

С помощью пунктов вертикального меню реализуются основные возможности программы [32]. При клике на пункт меню «Собрать комплект» отрываются списки для быстрого поиска необходимых товаров (рис. 2.15).

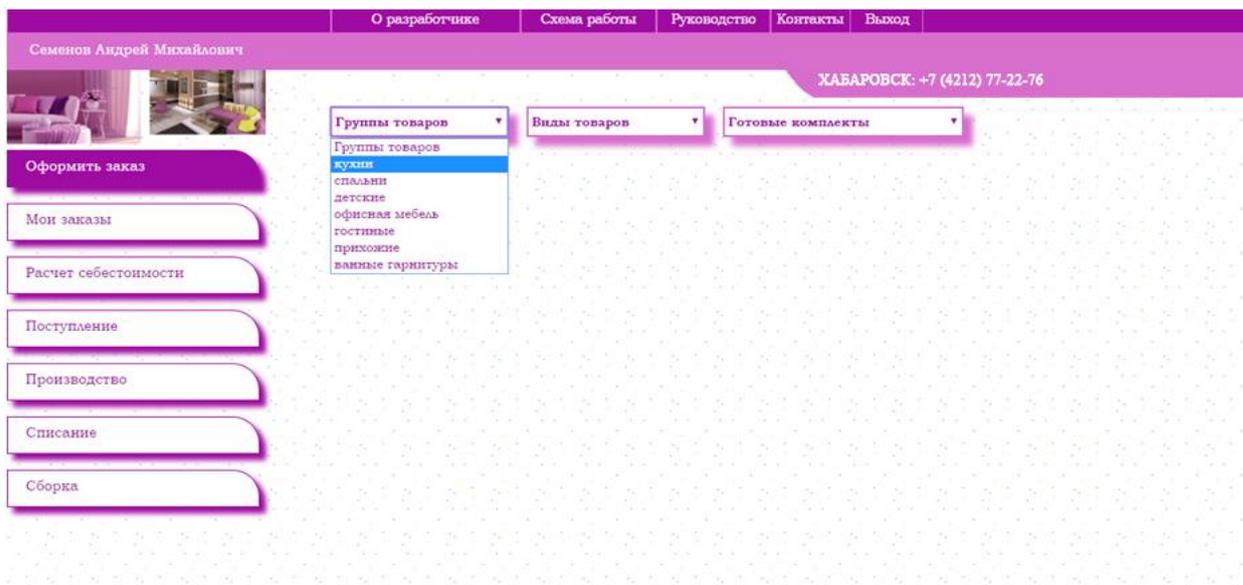


Рис. 2.15. Оформление заказа

При выборе группы товаров все товары фильтруются и выводятся на экран (см. рис. 2.16).

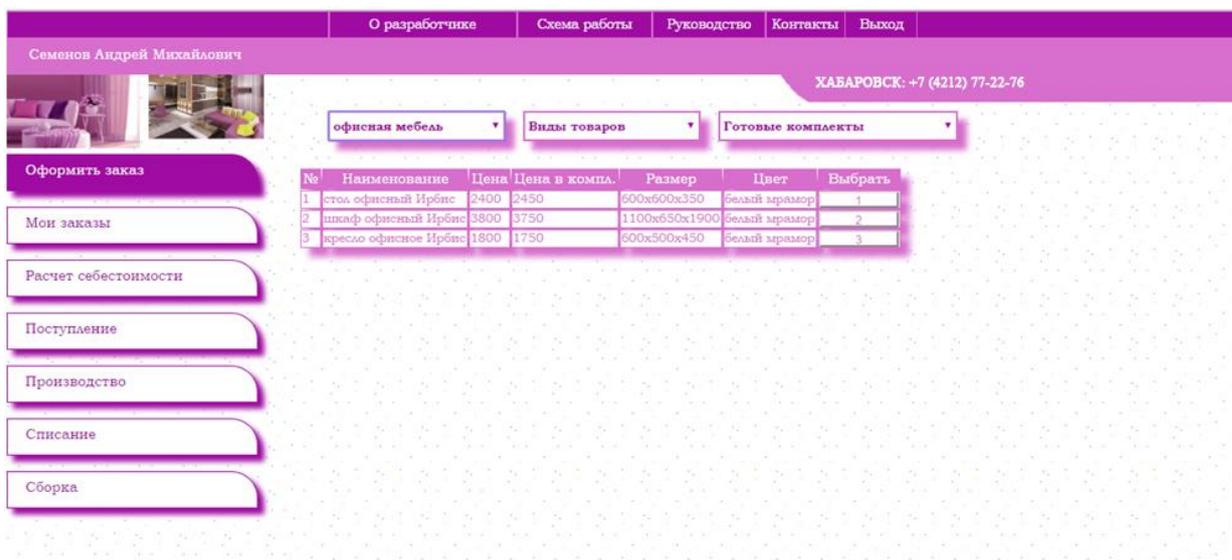


Рис. 2.16. Выбор группы товаров

Выбор вида товара представлен на рис. 2.17.

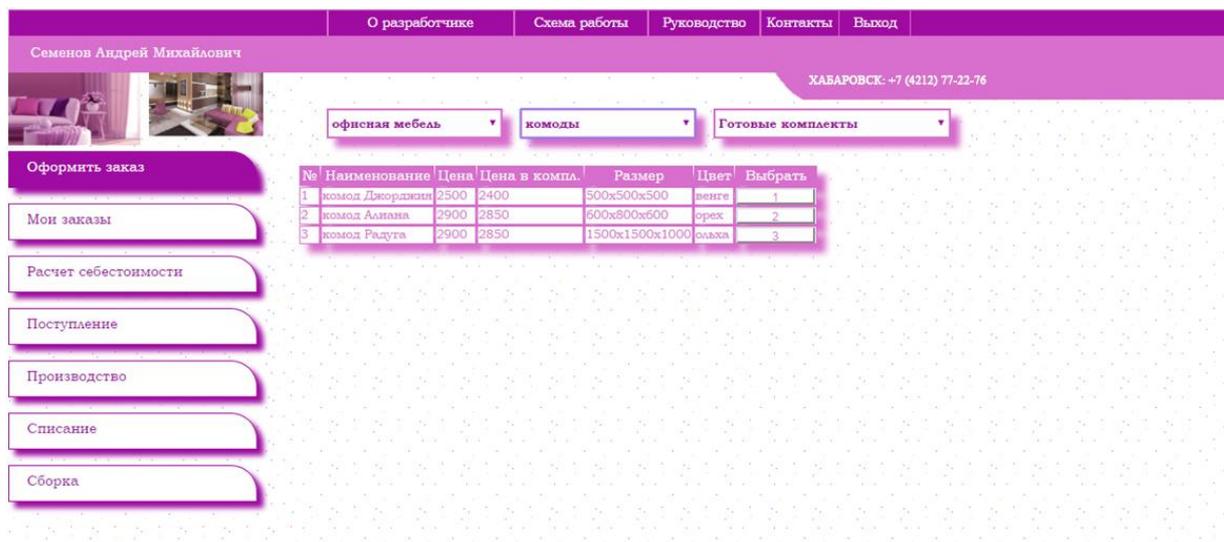


Рис. 2.17. Выбор вида товара

На рис. 2.18 показана работа функции фильтрации готовых комплектов (см. приложение).

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

офисная мебель | комоды | гарнитур кухонный Византия

№	Наименование	Цена	Цена в компа.	Размер	Цвет	Выбрать
1	стол кухонный Византия	3500	3400	500x500x500	вишня	1
2	шкаф кухонный Византия	3000	2900	400x500x500	вишня	2
3	полка кухонная Византия	2000	1900	350x500x400	вишня	3

Рис. 2.18. Фильтрация по комплектам

Если ничего не выбирать из представленных списков, отобразятся все имеющиеся товары (см. рис. 2.19).

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

Группы товаров | Виды товаров | Готовые комплекты

№	Наименование	Цена	Цена в компа.	Размер	Цвет	Выбрать
1	шкаф с Джорджи	3650	3500	800x600x1800	венге	1
2	шкаф с угрюссоло Джорджи	4290	4000	1000x600x2000	венге	2
3	пенал Джорджи	2540	2500	600x500x1600	венге	3
4	комод Джорджи	2500	2400	500x500x500	венге	4
5	стол кухонный Византия	3500	3400	500x500x500	вишня	5
6	шкаф кухонный Византия	3000	2900	400x500x500	вишня	6
7	полка кухонная Византия	2000	1900	350x500x400	вишня	7
8	кровать Аллана	4500	4400	500x1000x600	орех	8
9	шкаф платяной Аллана	4250	400	1000x500x2000	орех	9
10	шкаф обувной Аллана	2500	2450	800x800x200	орех	10
11	комод Аллана	2900	2850	600x800x600	орех	11
12	стол гостинный Вечер	3200	3250	600x700x800	кедр	12
13	диван Вечер	4000	3900	1000x2500x800	кедр	13
14	кресло Вечер	2500	2400	900x900x900	кедр	14
15	шкаф Вечер	3500	3450	1500x1500x900	кедр	15
16	шкаф Венеция	4500	4400	1100x700x1900	клен	16
17	шкаф угловой Венеция	3900	3800	100x600x1900	клен	17
18	полка Венеция	1900	1950	600x600x500	клен	18
19	ванная полка София	1400	1350	600x600x250	белый	19
20	важный шкаф София	2000	1950	750x600x460	белый	20
21	шкаф Радуга	3500	3450	800x100x1800	ольха	21
22	зеркало Радуга	2600	2500	500x500x10	ольха	22
23	кровать детская Радуга	3800	3750	2000x1000x500	ольха	23
24	комод Радуга	2900	2850	1500x1500x1000	ольха	24
25	стол офисный Ирбис	2400	2450	600x600x350	белый мрамор	25
26	шкаф офисный Ирбис	3800	3750	1100x650x1900	белый мрамор	26
27	кресло офисное Ирбис	1800	1750	600x500x450	белый мрамор	27

Рис. 2.19. Все имеющиеся товары

При клике на кнопку «Выбрать» выбранный товар добавляется в список заказа. На рис. 2.20 представлен список товаров, выбранных пользователем для заказа (см. приложение).

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

спальни | диваны | стенка Джорджия

Оформить заказ

Мои заказы

Расчет себестоимости

Поступление

Производство

Списание

Сборка

### ОФОРМЛЕНИЕ ЗАКАЗА

Выбранные товары

№	Наименование	Размер	Цвет	Цена	Цена в компл.	Кол-во	Комплект	Сумма
1	шкаф с Джорджия	800x600x1800	венге	3650	3500	1	<input checked="" type="checkbox"/>	
2	шкаф с антресолями Джорджия	1000x600x2000	венге	4290	4000	1	<input checked="" type="checkbox"/>	
3	стена Джорджия	600x600x1600	венге	2540	2500	2	<input checked="" type="checkbox"/>	
4	комод Джорджия	500x500x500	венге	2500	2400	2	<input checked="" type="checkbox"/>	
5	шкаф обувной Алана	800x800x200	орех	2500	2450	3	<input type="checkbox"/>	
6	диван Ветер	1000x2500x800	кедр	4000	3900	1	<input type="checkbox"/>	

Информация о заказчике

Фамилия:  Адрес:  Дата:

Имя:  Телефон:

Отчество:  Email:

ИТОГО: 0 руб.

Рис. 2.20 Оформление заказа

После выбора товаров для заказа пользователь отмечает, входят данные товары в готовый комплект или нет и выставляет количество товара. Автоматически рассчитывается сумма по каждому товару и итоговая сумма. Последним шагом оформления заказа будет заполнение информации о заказе. После клика по кнопке «Заказать» все данные сохраняются в базе данных. Заказ на данный момент имеет статус «Принят».

Для контроля и редактирования заказов, служит следующий пункт меню «Мои заказы». Для удобства поиска и фильтрации по клиентам, менеджерам и статусу предусмотрены списки выбора. На рис. 2.21 показан весь список заказов без проведения фильтрации.

№	Номер	Клиент	Менеджер	Статус	Дата	Показать
1	1000	Галкина Татьяна Васильевна	Чиркова Алina Васильевна	отгружен	2017-04-25	1
2	1001	Галкина Татьяна Васильевна	Белоусова Марина Сергеевна	принят	2017-05-17	2
3	1002	Смирнов Степан Афанасьевич	Огородова Дарья Игоревна	принят	2017-05-04	3
4	1003	Смирнов Степан Афанасьевич	Белоусова Марина Сергеевна	в работе	2017-04-15	4
5	1004	Гончаренко Марина Александровна	Петров Михаил Иванович	в работе	2017-04-20	5
6	1005	Гончаренко Марина Александровна	Огородова Дарья Игоревна	выполнен	2017-03-24	6
7	1006	Гончаренко Марина Александровна	Чиркова Алina Васильевна	завершен	2017-02-14	7
8	1007	Овчинников Дмитрий Степанович	Огородова Дарья Игоревна	принят	2017-05-16	8
9	1008	Овчинников Дмитрий Степанович	Чиркова Алina Васильевна	отгружен	2017-02-28	9
10	1009	Худяков Вадим Сергеевич	Белоусова Марина Сергеевна	принят	2017-05-06	10
11	1010	Иванов Александр Петрович	Петров Михаил Иванович	принят	2017-05-02	11
12	1011	Иванов Александр Петрович	Петров Михаил Иванович	в работе	2017-03-24	12
13	1012	Смирнова Светлана Васильевна	Огородова Дарья Игоревна	принят	2017-05-20	13
14	1013	Крюкова Ирина Викторовна	Чиркова Алina Васильевна	в работе	2017-04-02	14
15	1015	Гончарова Татьяна Владимировна	Белоусова Марина Сергеевна	завершен	2017-01-25	15
16	1016	Гончарова Татьяна Владимировна	Огородова Дарья Игоревна	выполнен	2017-01-14	16
17	1014	Крюкова Ирина Викторовна	Огородова Дарья Игоревна	отгружен	2017-03-05	17
18	1017	Гончарова Татьяна Владимировна	Петров Михаил Иванович	завершен	2017-02-09	18
19	1018	Завьялов Кирилл Иванович	Чиркова Алina Васильевна	принят	2017-05-19	19
20	1019	Семенов Андрей Михайлович	Огородова Дарья Игоревна	завершен	2017-01-05	20
21	1020	Семенов Андрей Михайлович	Белоусова Марина Сергеевна	завершен	2017-02-16	21
22	1021	Семенов Андрей Михайлович	Петров Михаил Иванович	в работе	2017-04-26	22

Рис. 2.21. Список всех заказов

Окно фильтрации заказов выбранного менеджера представлено на рис. 2.22.

№	Номер	Клиент	Менеджер	Статус	Дата	Показать
1	1012	Смирнова Светлана Васильевна	Огородова Дарья Игоревна	принят	2017-05-20	1
2	1002	Смирнов Степан Афанасьевич	Огородова Дарья Игоревна	принят	2017-05-04	2
3	1007	Овчинников Дмитрий Степанович	Огородова Дарья Игоревна	принят	2017-05-16	3
4	1014	Крюкова Ирина Викторовна	Огородова Дарья Игоревна	отгружен	2017-03-05	4
5	1016	Гончарова Татьяна Владимировна	Огородова Дарья Игоревна	выполнен	2017-01-14	5
6	1005	Гончаренко Марина Александровна	Огородова Дарья Игоревна	выполнен	2017-03-24	6
7	1019	Семенов Андрей Михайлович	Огородова Дарья Игоревна	завершен	2017-01-05	7

Рис. 2.22. Заказы выбранного менеджера

После клика на кнопку «Показать» открывается окно с подробной информацией о данном заказе с возможностью изменения статуса заказа (см. рис. 2.23).

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

О разработчике | Схема работы | Руководство | Контакты | Выход

Выберите клиента: Чиркова Алина Васильевна

Выберите статус

Заказ № 1013

Дата: 2017-04-02 | Статус: в работе

Клиент: Крюкова Ирина Викторовна  
Менеджер: Чиркова Алина Васильевна

№	Наименование	Размер	Цвет	Комплект	Цена	Кол-во	Сумма
1	шкаф Венеция	1100x700x1900	клен	к	4500	1	4500
2	шкаф угловой Венеция	100x600x1900	клен	к	3900	1	3900
3	полка Венеция	600x600x500	клен	к	1900	1	1900

ИТОГО: 10300 руб.

Распечатать

Рис. 2.23. Подробная информация о выбранном заказе

Пункт меню «Расчет себестоимости» предоставляет пользователю возможность просмотра материалов, входящих в состав выбранного товара и расчета его себестоимости (рис. 2.24).

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

О разработчике | Схема работы | Руководство | Контакты | Выход

По виду товара | По товару | По материалу

№	Вид	Товар	Размер	Цвет	Материал	Ед.изм.	Цена	Кол-во	Показать
1	кровать	кровать детская Радуга	2000x1000x500	оak	доска дуб обрешетка сухая	шт	50	1	1
2	кровать	кровать детская Радуга	2000x1000x500	оak	ПВХ пленка для фасадов однотонная	кв.м	100	3	2
3	кровать	кровать детская Радуга	2000x1000x500	оak	ДВП декорированная	лист	8	1	3
4	кресла	кресло Вечер	900x900x900	келдр	древесностружечная плита 1	анст	60	2	4
5	кресла	кресло Вечер	900x900x900	келдр	стекло метакриловое полупрозрачное	кв.м	250	3	5
6	кресла	кресло Вечер	900x900x900	келдр	ПВХ пленка для фасадов страйт	кв.м	5	3	6
7	кресла	кресло Вечер	900x900x900	келдр	штг мебельный сращенный	лист	5	1	7
8	кресла	кресло Вечер	900x900x900	келдр	клей универсальный	штг	25	2	8
9	кресла	кресло офисное Ирис	600x500x450	белый мрамор	зеркало Фантазия	кв.м	300	2	9
10	кресла	кресло офисное Ирис	600x500x450	белый мрамор	стекло метакриловое полупрозрачное	кв.м	250	2	10
11	кресла	кресло офисное Ирис	600x500x450	белый мрамор	МДФ ламинированная белая	лист	14	2	11
12	кровать	кровать детская Радуга	2000x1000x500	оak	перемычка дверная	штг	50	4	12
13	комоды	комод Радуга	1500x1500x1000	оak	ABS кромка фоновая шагреня	п.м.	90	1	13
14	комоды	комод Радуга	1500x1500x1000	оak	ДВП декорированная	лист	8	2	14
15	комоды	комод Радуга	1500x1500x1000	оak	шуруп	штг	2	2	15
16	комоды	комод Радуга	1500x1500x1000	оak	ручка дверная	штг	30	3	16
17	комоды	комод Джорджия	500x500x500	венге	штг мебельный сращенный	лист	500	2	17
18	комоды	комод Джорджия	500x500x500	венге	ПВХ пленка для фасадов страйт	кв.м	5	3	18
19	комоды	комод Джорджия	500x500x500	венге	шуруп 2,5x10	штг	2	1	19
20	кровать	кровать Алана	500x1000x600	орех	стекло с тонирующими пленками	кв.м	200	1	20
21	кровать	кровать Алана	500x1000x600	орех	древесностружечная плита 3	лист	50	2	21
22	кровать	кровать Алана	500x1000x600	орех	штг мебельный сращенный	лист	500	3	22
23	кровать	кровать Алана	500x1000x600	орех	ABS кромка фоновая шагреня	п.м.	90	4	23
24	кровать	кровать Алана	500x1000x600	орех	ДВП декорированная	лист	8	4	24
25	комоды	комод Алана	600x800x600	орех	ПВХ пленка для фасадов	кв.м	7	3	25
26	комоды	комод Алана	600x800x600	орех	стекло метакриловое полупрозрачное	кв.м	250	1	26

Рис 2.24. Фильтрация по виду товара

При вводе каждого символа в строку ввода все товары фильтруются по виду товара, по наименованию и по материалу. Это необходимо для удобства поиска. На рис. 2.25, 2.26 показана фильтрация товаров по наименованию и материалу [33].

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

По виду товара По товару По материалу

к крн

№	Вид	Товар	Размер	Цвет	Материал	Ед.изм.	Цена	Кол-во	Показать
1	кресла	кресло Ветер	900x900x900	кедр	древесностружечная плита 1	лист	60	2	1
2	кресла	кресло Ветер	900x900x900	кедр	стекло метакриловое полупрозрачное	кв.м	250	3	2
3	кресла	кресло Ветер	900x900x900	кедр	ПВХ пленка для фасадов страйп	кв.м	5	3	3
4	кресла	кресло Ветер	900x900x900	кедр	шит мебельный кедр сращенный	лист	5	1	4
5	кресла	кресло Ветер	900x900x900	кедр	клей универсальный	шт	25	2	5
6	кресла	кресло офисное Ирибе	600x500x450	белый мрамор	зеркало Фантазия	кв.м	300	2	6
7	кресла	кресло офисное Ирибе	600x500x450	белый мрамор	стекло метакриловое полупрозрачное	кв.м	250	2	7
8	кресла	кресло офисное Ирибе	600x500x450	белый мрамор	МДФ ламинированная белая	лист	14	2	8

Оформить заказ  
Мои заказы  
Расчет себестоимости  
Поступление  
Производство  
Списание  
Сборка

Рис. 2.25. Просмотр товаров

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

По виду товара По товару По материалу

к крн а

№	Вид	Товар	Размер	Цвет	Материал	Ед.изм.	Цена	Кол-во	Показать
1	кресла	кресло Ветер	900x900x900	кедр	древесностружечная плита 1	лист	60	2	1

Оформить заказ  
Мои заказы  
Расчет себестоимости  
Поступление

Рис. 2.26. Фильтрация товаров по наименованию и материалу

При выборе товара отображается информация о входящих в его состав материалах, представленная на рис. 2.27.

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

О разработчике | Схема работы | Руководство | Контакты | Выход

По виду товара | По товару | По материалу

Оформить заказ

Мои заказы

**Расчет себестоимости**

Поступление

Производство

Списание

Сборка

№ Вид

1 кресло

2 кресло

3 кресло

4 кресло

5 кресло

6 кресло

7 кресло

8 кресло

9 кресло

10 кресло

11 кресло

12 кресло

13 комплект

14 комплект

15 комплект

16 комплект

17 комплект

18 комплект

19 комплект

20 кресло

21 кресло

22 кресло

23 кресло

кресло Вечер

Израсходованные материалы

№	Вид	Сырье	Цвет	Размер	Ед.из	Цена	Кол-во	Сумма
1	основные	древесностружечная плита 1	дуб	3500x1750x10	лист	60	2	120
2	основные	стекло метакриловое полупрозрачное	серый	1000x1200	кв.м	250	3	750
3	основные	ПВХ пленка для фасадов страйп	прозрачный	1400x520	кв.м	5	3	15
4	основные	лист мебельный кедр сращенный	кедр	500x700	лист	5	1	5
5	вспомогательные	клей универсальный		шт	25	2	50	

Прочие затраты

возвратные отходы	454	топливо и энергия	45
амортизация ОС и НМА	12	зарплата работников	25
страховые отчисления	23	общепроизводственные	56
общезастыственные		потери от брака	
прочие расходы		расходы на реализацию	

ИТОГО: 940 руб.

Распечатать

Рис. 2.27. Расчет себестоимости товара

Кроме расчета итоговой суммы по каждому материалу и расчета общей суммы расхода материалов пользователь может ввести суммы по другим затратам и рассчитать себестоимость товара [34, 35, 36].

В программе имеется возможность печати ведомостей и отчетов по оформлению заказов, расчету себестоимости, сборке и других. Отчеты формируются в формате pdf [35, 36].

С помощью функции «Поступление материалов» реализуется возможность учета материалов, поступающих на предприятие (см. рис. 2.28).

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

### ПОСТУПЛЕНИЕ МАТЕРИАЛОВ

№	Вид	Наименование	Ед.изм	Цена	Размер	Цвет	Дата	Кол-во
1	основные	древесностружечная плита	лист	50	3500x1750x10	венге	2017-04-25	5
2	основные	древесностружечная плита 1	лист	60	3500x1750x10	дуб	2017-04-12	5
3	основные	ПВХ для мембранного прессования	кв.м	150	250x1500	металлик	2017-05-12	4
4	основные	древесностружечная плита 3	лист	50	3500x1750x10	металлик	2017-05-10	6
5	основные	доска бук обрешечная сухая	шт	50	1500x2500	вишня	2017-04-15	12
6	основные	ПВХ пленка для фасадов однотонная	кв.м	100	4500x3500	оливка	2017-03-25	13
7	основные	ПВХ пленка для фасадов матовая древесная	кв.м	5	500x5000	белый	2017-03-06	11
8	основные	доска сосна обрешечная сухая	шт	20	1500x1200	кедр	2017-03-25	9
9	основные	ABS крошка фоновая шагреня	т.м.	90	5000x20	белый зрморо	2017-03-19	10
10	основные	стекло матовое с защитной пленкой	кв.м	9	1800x2000	серый	2017-03-29	5
11	основные	шпнт мебельный кедр сращенный	лист	5	500x700	кедр	2017-04-08	5
12	основные	МДФ ламинированная белая	лист	14	500x700x1200	белый зрморо	2017-04-09	8
13	вспомогательные	лак акриловый быстросохнущий	литр	5		белый зрморо	2017-05-08	19
14	вспомогательные	шурупы 2,5x10	шт	2	2,5x10		2017-05-01	64
15	вспомогательные	шурупы	шт	2	2,7x11		2017-03-29	12
16	вспомогательные	ручка дверная	шт	30	15x10	белый зрморо	2017-04-28	34
17	вспомогательные	клей универсальный	шт	25			2017-05-06	21

Поступление нового материала

Вид:  Наименование:

Ед.измер.:  Цена:

Размер:  Цвет:

Рис. 2.28. Учет поступления материалов

Для сохранения нового поступившего материала необходимо ввести данные о нем и нажать кнопку «Сохранить».

Новый изготовленный товар можно сохранить при выборе пункта меню «Производство». Здесь также отображается информация обо всех произведенных товарах (см. рис. 2.29).

Семенов Андрей Михайлович

ХАБАРОВСК: +7 (4212) 77-22-76

### ПРОИЗВОДСТВО ПРОДУКЦИИ

№	Группа	Вид	Комплект	Наименование	Цена	Цена в комп.	Размер	Цвет	Дата	Кол-во
1	гостиные	шкафы	стенка Джорджия	шкаф с Джорджия	800x600x1800	венге	3650	3500	2017-05-09	5
2	гостиные	шкафы	стенка Джорджия	шкаф с антресолю Джорджия	1000x600x2000	венге	4290	4000	2017-05-12	5
3	гостиные	шкафы	стенка Джорджия	пенал Джорджия	600x600x1600	венге	2540	2500	2017-04-25	4
4	гостиные	комоды	стенка Джорджия	комод Джорджия	500x500x500	венге	2500	2400	2017-04-15	1
5	кухни	столы	гарнитур кухонный Византия	стол кухонный Византия	500x500x500	вишня	3500	3400	2017-04-16	5
6	кухни	шкафы	гарнитур кухонный Византия	шкаф кухонный Византия	400x500x500	вишня	3000	2900	2017-04-20	9
7	кухни	полки	гарнитур кухонный Византия	полка кухонная Византия	350x500x400	вишня	2000	1900	2017-05-08	12
8	спальни	шкафы	спальня Алана	шкаф обрешечная Алана	800x800x200	орех	2500	2450	2017-05-20	11
9	спальни	комоды	спальня Алана	комод Алана	600x800x600	орех	2900	2850	2017-05-19	10
10	гостиные	диваны	гостиная Ветер	диван Ветер	1000x2500x800	кедр	4000	3900	2017-05-02	9
11	гостиные	кресла	гостиная Ветер	кресло Ветер	900x900x900	кедр	2500	2400	2017-04-09	6
12	гостиные	шкафы	гостиная Ветер	шкаф Ветер	1500x1500x900	кедр	3500	3450	2017-04-05	8
13	прихожие	шкафы	прихожая Венеция	шкаф угловой Венеция	100x600x1900	клен	3900	3800	2017-05-16	7
14	прихожие	полки	прихожая Венеция	полка Венеция	600x600x500	клен	1900	1950	2017-03-15	9
15	ванные гарнитуры	шкафы	ванная гарнитур София	ванная гарнитур София	750x600x460	белый	2000	1950	2017-03-18	2
16	детские	шкафы	детская Радуга	шкаф Радуга	800x100x1800	оливка	3500	3450	2017-03-12	8
17	детские	зеркала	детская Радуга	зеркало Радуга	500x500x10	оливка	2600	2500	2017-04-22	18
18	детские	кровати	детская Радуга	кровать детская Радуга	2000x1000x500	оливка	3800	3750	2017-03-15	11
19	детские	комоды	детская Радуга	комод Радуга	1500x1500x1000	оливка	2900	2850	2017-04-09	16
20	офисная мебель	столы	офисный кабинет Ирбис	стол офисный Ирбис	600x600x350	белый зрморо	2400	2450	2017-03-25	8
21	офисная мебель	кресла	офисный кабинет Ирбис	кресло офисное Ирбис	600x500x450	белый зрморо	1800	1750	2017-03-16	3

Рис. 2.29. Производство продукции

Для сохранения нового товара в базе данных нужно заполнить информацию о нем и нажать на кнопку «Сохранить» (рис. 2.30).

№	Категория	Тип	Наименование	Описание	Размеры	Цена	Дата	Статус
5	кухни	столы	кухонный Бизантия	кухонный Бизантия	500x500x			
6	кухни	шкафы	гарнитур кухонный Бизантия	шкаф кухонный Бизантия	400x500x500	3000	2900	2017-04-20
7	кухни	полки	гарнитур кухонный Бизантия	полка кухонная Бизантия	350x500x400	2000	1900	2017-05-08
8	спальни	шкафы	спальня Алана	шкаф обувной Алана	800x800x200	2500	2450	2017-05-20
9	спальни	комоды	спальня Алана	комод Алана	600x800x500	2900	2850	2017-05-19
10	гостиные	диваны	гостиная Вечер	диван Вечер	1000x2500x800	4000	3900	2017-05-02
11	гостиные	кресла	гостиная Вечер	кресло Вечер	900x900x900	2500	2400	2017-04-09
12	гостиные	шкафы	гостиная Вечер	шкаф Вечер	1500x1500x900	3500	3450	2017-04-05
13	прихожие	шкафы	прихожая Венеция	шкаф угловой Венеция	1000x600x1900	3900	3800	2017-05-16
14	прихожие	полки	прихожая Венеция	полка Венеция	600x600x500	1900	1950	2017-03-15
15	ванные гарнитуры	шкафы	ванная гарнитура София	ванная шкаф София	750x600x460	2000	1950	2017-03-18
16	детские	шкафы	детская Радуга	шкаф Радуга	800x100x1800	3500	3450	2017-03-12
17	детские	зеркала	детская Радуга	зеркало Радуга	500x500x10	2600	2500	2017-04-22
18	детские	кровати	детская Радуга	кровать детская Радуга	2000x1000x500	3800	3750	2017-03-15
19	детские	комоды	детская Радуга	комод Радуга	1500x1500x1000	2900	2850	2017-04-09
20	офисная мебель	столы	офисный кабинет Ирбис	стол офисный Ирбис	600x600x350	2400	2450	2017-03-25
21	офисная мебель	кресла	офисный кабинет Ирбис	кресло офисное Ирбис	600x500x450	1800	1750	2017-03-19

Производство нового продукта

Группа:  Вид:

Комплект:  Наименование:

Цена:

Размер:

Дата:

Цена\_в\_ком:

Цвет:

Количество:

Рис. 2.30. Сохранение в базе данных нового товара

Аналогичным образом ведется учет списанных товаров и сборка товара.

В практической части дипломного проекта была разработана информационная база данных, составлена структурная схема проекта, спроектирована разметка страниц и пути навигационных маршрутов по пунктам меню и другим элементам, разработаны и реализованы алгоритмы работы программы с помощью языков Web-программирования и фреймворков, разработаны справочная система и руководство пользователя.

## ГЛАВА 3 АПРОБАЦИЯ И ВНЕДРЕНИЕ WEB-ПРИЛОЖЕНИЯ

### 3.1 Тестирование работоспособности приложения

Объектом тестирования является программный продукт «Web-приложение для учета и контроля производства мебели».

При подготовке к вводу программы в опытную эксплуатацию необходимо провести испытания. Тестовые испытания представляют собой процесс проверки выполнения заданных функций системы, выявления и устранения недостатков в программном обеспечении, оборудовании и руководстве пользователя.

Следующие требования подлежат проверке:

- надежность расчетов;
- стабильное функционирование программы;
- корректность отображения и вывода исходной информации.

В качестве средства для проверки ошибок ввода/вывода, динамических и ошибок времени выполнения использовался отладчик. Логические ошибки устранялись вручную.

Испытания проходили в следующем порядке:

- поиск и отладка синтаксических ошибок и ошибок времени выполнения программы;
- корректность расчетов, проводимых в программе;
- проверка входной информации;
- корректность отображения выходной информации;
- проверка программы на устойчивость к сбоям в работе [32].

На языке Java Script написаны функции обработки вводимых пользователем данных. Если пользователь заполнил не все поля формы с личными данными, выводится соответствующее сообщение и программа прекращает свою работу (см. листинг 3.1).

## Листинг 3.1 Предусмотрение ввода неполной информации

```

function save($event)
{
    if (
        (document.getElementById('ed15').value=="||
        (document.getElementById('ed16').value=="||
        (document.getElementById('ed12').value=="||
        (document.getElementById('ed22').value=="||
        (document.getElementById('ed30').value=="
    )
    {alert('Вы заполнили не все поля!');
      $event.preventDefault();
    }
    if (document.getElementById('ed22').value ==")
    { document.getElementById('ed22').style.backgroundColor="red";
      document.getElementById('ed22').style.opacity=0.7;
    }
    if (document.getElementById('ed12').value ==")
    { document.getElementById('ed12').style.backgroundColor="red";
      document.getElementById('ed12').style.opacity=0.7;
    }
    if (document.getElementById('ed30').value ==")
    {
      document.getElementById('ed30').style.backgroundColor="red";
      document.getElementById('ed30').style.opacity=0.7;
    }
    if (document.getElementById('ed15').value ==")
    {
      document.getElementById('ed15').style.backgroundColor="red";
      document.getElementById('ed15').style.opacity=0.7;
    }
    if (document.getElementById('ed16').value ==")
    {
      document.getElementById('ed16').style.backgroundColor="red";
      document.getElementById('ed16').style.opacity=0.7;
    }
  }
}

```

При вводе некорректных логина и пароля для входа в систему выдается соответствующее сообщение.

После выбора пользователем пункта меню Вход/Регистрация, а также при редактировании данных в личном кабинете пользователю необходимо

заполнить поля формы. В программе предусмотрен вывод сообщений о том, что пользователь не зарегистрирован, о том, что данные некорректны или введены не в полном объеме. В поле для ввода фамилии, имени программой предусмотрен запрет на ввод цифр, первая буква автоматически становится заглавной [35].

Программа подвергалась многократному тестированию, то есть выполнению при заданных условиях с целью фиксации реальных результатов ее работы.

Каждая операция анализировалась, проверялась на правильность результата при различных условиях и значениях. Корректность отображения выходной информации проводилась на тестирующих примерах, в которых выявлялись ошибки и неточности при отображении на экране [36].

Таким образом, программа протестирована и готова к следующему этапу своего жизненного цикла – внедрению в производство.

### **3.2 Внедрение в производство**

Разработанное в данном дипломном проекте Web-приложение было успешно апробировано и протестировано. В программе не было выявлено критических ошибок и сбоев. Система работает соответственно поставленным целям и задачам ее функционирования. Были исправлены неточности описания, а также мелкие недочеты и сбои в ее работе. Web-сайт успешно выполняет свои функции на последних версиях актуальных браузеров. Старые версии браузеров не рекомендованы к использованию, так как в программе используются новейшие версии средств Web-разработки и поэтому возможны ошибки в разметке сайта и в его работе в целом.

Изначально сайт разрабатывался с оптимизацией под мобильные устройства. В связи с этим, он имеет урезанный функционал при открытии на мобильных платформах. На данном этапе программа была успешно внедрена в пользование компанией ООО «Триллиант».

## ЗАКЛЮЧЕНИЕ

В наше время необходимо использовать все доступные и удобные средства для автоматизации производства. Они позволяют отладить, оптимизировать работу предприятия и бизнес процессы, протекающие в нем. Внедрение автоматизированных систем способствует развитию бизнеса, а тем самым повышает уровень жизни и экономики государства.

В работе над проектом были изучены теоретические аспекты ведения учета и контроля производства на мебельном предприятии. В тесном сотрудничестве с работниками компании ООО «Триллиант» проводилось планирование этапов работы над проектом, обсуждение алгоритмов работы программы, производимых расчетов, разработано программное средство, которое обладает такими техническими характеристиками и функциями как надежность и защита от сбоев и ошибок, понятный и логичный интерфейс, высокая скорость обработки данных. Оно поможет менеджеру по учету и контролю производства продукции быстро и качественно выполнять свою работу.

В ходе работы над дипломным проектом были отработаны практические навыки использования современных средств Web-разработки, позволяющих создавать яркие с красивыми визуальными эффектами Web-страницы. Были изучены алгоритмические структуры, система команд, функций и процедур языков программирования PHP, Java Script, а также фреймворки JQuery, Yii, приобретен большой практический опыт разработки Web-приложений.

Цель проекта реализована в полной мере, задачи выполнены.

## СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. <http://www.rusprofile.ru/id/392982>;
2. <http://coolreferat.com/>;
3. [http://studopedia.ru/7\\_147152\\_uchet-i-kontrol-izderzhek-proizvodstva-i-prodazh-produktsii-po-vidam-rashodov.html](http://studopedia.ru/7_147152_uchet-i-kontrol-izderzhek-proizvodstva-i-prodazh-produktsii-po-vidam-rashodov.html);
4. <http://allbest.ru/o-2c0a65635a3bd78a5d43a88421216d37.html>;
5. Котеров Д. В., Костарев А. Ф. РНР 5 в подлиннике. Издательство: БХВ-Петербург, 2006 г.;
6. М.В. Кузнецов, И.В. Симдянов. MySQL 5– СПб.: БХВ-Петербург, 2010.;
7. Дейт, К. Введение в системы баз данных // 6-издание. / К. Дейт - Киев: Диалектика, 1998.;
8. [Yiiframework.com](http://yiiframework.com);
9. <http://rutracker.org/forum/viewtopic.php?t=4172581>;
10. <http://uroki-jquery.ru/>;
11. <http://jquerybook.ru/>;
12. Мейер, М. Теория реляционных баз данных. / М. Мейер- М.: Мир, 1987.;
13. Чен, П. Модель "сущность-связь" - шаг к единому представлению о данных //СУБД. / П.Чен - М. 1995.;
14. Ульман Л. MySQL. Руководство по изучению языка. Издательство: ДМК Пресс; Питер;
15. Грофф Дж.Р., Вайнберг П.Н. SQL полное руководство, ISBN; Издательство: 2001;
16. Осипов Д. Л., Базы данных и DELPHI. Теория и практика, Издательство: ВHV 2011;
17. Дж. Уорсли, Дж. Дрейк, PostgreSQL. Для профессионалов, Питер;
18. <http://citforum.ru/database/postgresql/>;

19. [http://c4.ucoz.net/blog/uchebnik\\_po\\_postgresql\\_na\\_russkom\\_jazyke/2010-02-26-20](http://c4.ucoz.net/blog/uchebnik_po_postgresql_na_russkom_jazyke/2010-02-26-20);
20. [http://foto-narva.ru/rqlxdzjnkfwcnrdu.html?page=uchebnik\\_po\\_postragesql](http://foto-narva.ru/rqlxdzjnkfwcnrdu.html?page=uchebnik_po_postragesql);
21. С.А. Филиппов Основы современного веб-программирования учебное пособие ля студентов высших учебных заведений Москва 2011;
22. Ричард Вагнер, Аллен Вайк, Java Script. Энциклопедия пользователя. Издательство: ДиаСофт, 2001 г.;
23. <http://zametkinapolyah.ru/veb-programmirovanie/zametki-po-javascript>;
24. <http://www.puzzleweb.ru/>;
25. Rasmus Lerdorf, Kevin Tatroe Programming PHP; Издание Товарищества В. В. Думнов, наследн. бр. Салаевых – Москва, 1995.;
26. Vikram Vaswani XML and PHP; New Riders - Москва, 2002. ;
27. Джон Коггзолл, PHP 5. Полное руководство. Издательство: Диалектика, 2006 г.;
28. <http://reftrend.ru/987830.html>;
29. <http://www.php.su/learnphp/?syntax>;
30. <https://ru.wikipedia.org/wiki/PHP>;
31. <http://masters.donntu.org/2008/fvti/reznichenko/library/article03.htm>;
32. Дюбуа, Поль MySQL; М.: Вильямс; Издание 2-е - Москва, 2004.;
33. Кузнецов М., Симдянов И. Объектно-ориентированное программирование на PHP; БХВ-Петербург - Москва, 2008.;
34. Маркин А. В., Шкарин С. С. Основы Web-программирования на PHP; Диалог-МИФИ - , 2012.;
35. Пауэрс Дэвид Adobe Dreamweaver, CSS, Ajax и PHP; БХВ-Петербург - Москва, 2009.;
36. Ульман, Ларри MySQL; М.: ДМК Пресс - Москва, 2004.

## ПРИЛОЖЕНИЯ

### Файл reg\_ok.php

```

<?php
use yii\web\Session;
$session = Yii::$app->session;
$session['nc'] =0;
$session['str']=";
?>
<?php
use yii\helpers\Html;
use yii\widgets\ActiveForm;
use app\modules\commerce\models\UtpOrder;
use kartik\select2>Select2;
$connection=Yii::$app()->db;
$connection=new CDbConnection("host=localhost dbname=furniture user=postgres
port=5432 password=1");
if ($connection===false)
{
    echo "Error opening DB!<br>";
} else echo "all ok";

public static function getArrayAttributesId($id) {
    $attributeModels = self::find()->where(['nomenclature_id' => $id])->all();
    $array = ArrayHelper::map($attributeModels, 'attribute_id', 'attribute_id');
    return $array;
}

$result=pg_query($connection,"select n_man , fio , login , passw
                    from managers;");
$r=pg_num_rows($result);

if (isset($_POST['edt1']) && isset($_POST['edt2']))
{

    $result=pg_query($connection,"select * from managers;");
    $r=pg_num_rows($result);
    $u=0;
    for ($i=0; $i < $r; $i++)
    { $rows=pg_fetch_row($result,$i);
      if ($_POST['edt1']==$rows[2] && $_POST['edt2']==$rows[3])
      {
          $u=1;

          $_SESSION['n_man']=$rows[0];
          $_SESSION['log']=$rows[1];
      }
    }
    if ($u==0)

```

```

    {
      header("Location: ".$_SERVER['HTTP_REFERER']);
    }
  }
?>
<link rel="shortcut icon" href="favicon.ico" type="image/ico">
<meta charset="utf-8"> <title>Система учета и контроля комплектации мебели</title>
<style>
  input.i1 {font-family: Bookman Old Style; font-size:12px; color:#d86ecd; font-weight:bold;
font-style: italic;}
  input.i3 {font-family: Bookman Old Style; font-size:14px; color:#d86ecd;}

  div.d2 {background-color:white; color: #a00ba2;
position:absolute;width:160px;height:35px; left:40px;
  top:30px;box-shadow: 10px 10px 10px #d86ecd;border:solid 2px #d86ecd;font-family:
Bookman Old Style;
  font-size: 14px;}
  .dd2 {background-color:white; color: #a00ba2; position:absolute;width:200px;height:35px;
left:40px;
  top:30px;box-shadow: 10px 10px 10px #d86ecd;border:solid 2px #d86ecd;font-family:
Bookman Old Style;
  font-size: 14px;}

  div.d3 {background-color:white; color: #a00ba2;
position:absolute;width:220px;height:30px; left:40px;
  top:10px;box-shadow: 10px 10px 10px #d86ecd;border:solid 2px #d86ecd;font-family:
Bookman Old Style;
  font-size: 14px;}

  .line1 {background-color: #d86ecd; width:100%; height:72px;
  position: fixed;display: inline;
  top: 0px;left: 0px;
  color:#ffffff; font-family: Bookman Old Style; font-size:15px;}
/*955 362 и линия верт*/

  .radio {box-shadow: 8px 8px 8px #d86ecd; display: inline;position: relative;top: 7px; left:
5px;}
  .checkB {background-color:#d86ecd; color: white; font-weight:bold;display: inline;position:
relative;top: 7px;left: 50px;}
  H3 {color:#d86ecd; font-family: Bookman old style; font-size:18px; font-
weight:normal;box-shadow: 8px 8px 8px #d86ecd;
  display: inline; position: relative;top: 0px;left: 0px;margin: 0; padding: 0;}
  select {font-weight:bold;color:white;background-color:#d86ecd;}

  .td1 {
  border: 2px solid #d86ecd;
  background-color: #d86ecd;
  box-shadow: 10px 10px 10px #d86ecd;
  font-family: Bookman Old Style;

```

```

font-size: 16px;
text-align: center;
color: white;
}
td{
border: 1px solid #d86ecd;
background-color:white;
box-shadow: 10px 10px 10px #d86ecd;
font-family: Bookman Old Style;
font-size:14px;
color:#d86ecd;
}
div.d1 {background-color:white;
/*display: inline-block; */
border:solid 1px #a00ba2;
color:#a00ba2;
position:fixed;width:20%;height:40px; left:0%;
top:50px;
box-shadow: 7px 7px 7px #a00ba2;
border-top-right-radius: 30px;
font-family: Bookman Old Style;
cursor: pointer;}
.dd:hover {background-color:#ffffff; color:#a00ba2 ; }

div.dd {
background-color:#a00ba2;

}
.s1 {background-color:#d86ecd; color:white; font-weight:normal;}
.s2 {background-color:white; color: #d86ecd; font-weight:normal;}
.ph {color:#d86ecd; font-family: Bookman old style; font-size:18px; font-weight:normal;
position: relative;top: 30px;}
body {background: url(WB02073_.GIF);}

#parent_popup {
height: 100%;
position: fixed;
width: 100%;
z-index: 100;
top: 0;
left: 0;
}
#popup {
background: url(WB02073_.GIF);
height: 635px;
position: fixed;
top: 0%;
left: 15%;
color: #a00ba2;
width: 70%;
border: #a00ba2 1px solid;

```

```
    }
</style>
<script>
var g=0;
function ft()
{
  document.getElementById("de8").style.display="none";
}
function mf1()
{if (g!=1) {
  document.getElementById("de1").style.backgroundColor="#a00ba2";
  document.getElementById("de1").style.color="white";
  document.getElementById("de1").style.opacity=0.7;
  }
}
function f1()
{if (g!=1) {
  document.getElementById("de1").style.backgroundColor="white";
  document.getElementById("de1").style.color="#a00ba2";
  document.getElementById("de1").style.opacity=1;
  }
}
function mf2()
{if (g!=2) {
  document.getElementById("de2").style.backgroundColor="#a00ba2";
  document.getElementById("de2").style.color="white";
  document.getElementById("de2").style.opacity=0.7;
  }
}
function f2()
{if (g!=2) {
  document.getElementById("de2").style.backgroundColor="white";
  document.getElementById("de2").style.color="#a00ba2";
  document.getElementById("de2").style.opacity=1;
  }
}
function mf3()
{if (g!=3) {
  document.getElementById("de3").style.backgroundColor="#a00ba2";
  document.getElementById("de3").style.color="white";
  document.getElementById("de3").style.opacity=0.7;
  }
}
function f3()
{if (g!=3) {
  document.getElementById("de3").style.backgroundColor="white";
  document.getElementById("de3").style.color="#a00ba2";
  document.getElementById("de3").style.opacity=1;
  }
}
function mf4()
```

```
{if (g!=4) {
  document.getElementById("de4").style.backgroundColor="#a00ba2";
  document.getElementById("de4").style.color="white";
  document.getElementById("de4").style.opacity=0.7;
};
}
function f4()
{if (g!=4) {
  document.getElementById("de4").style.backgroundColor="white";
  document.getElementById("de4").style.color="#a00ba2";
  document.getElementById("de4").style.opacity=1;
};
}
function mf5()
{if (g!=5) {
  document.getElementById("de5").style.backgroundColor="#a00ba2";
  document.getElementById("de5").style.color="white";
  document.getElementById("de5").style.opacity=0.7;
};
}
function f5()
{if (g!=5) {
  document.getElementById("de5").style.backgroundColor="white";
  document.getElementById("de5").style.color="#a00ba2";
  document.getElementById("de5").style.opacity=1;
};
}
function mf6()
{if (g!=6) {
  document.getElementById("de6").style.backgroundColor="#a00ba2";
  document.getElementById("de6").style.color="white";
  document.getElementById("de6").style.opacity=0.7;
};
}
function f6()
{if (g!=6) {
  document.getElementById("de6").style.backgroundColor="white";
  document.getElementById("de6").style.color="#a00ba2";
  document.getElementById("de6").style.opacity=1;
};
}
function mf7()
{if (g!=7) {
  document.getElementById("de7").style.backgroundColor="#a00ba2";
  document.getElementById("de7").style.color="white";
  document.getElementById("de7").style.opacity=0.7;
};
}
function f7()
{if (g!=7) {
  document.getElementById("de7").style.backgroundColor="white";
```

```

    document.getElementById("de7").style.color="#a00ba2";
    document.getElementById("de7").style.opacity=1;
};
}

function mfu1($ee)
{
    document.getElementById("fu"+$ee).style.backgroundColor="white";
    document.getElementById("fun"+$ee).style.color="#a00ba2";
}
function mfun1($ee)
{
    document.getElementById("fu"+$ee).style.backgroundColor="#a00ba2";
    document.getElementById("fun"+$ee).style.color="white";
}
    document.getElementById("de4").style.opacity=1;
    g=4;
}
function myf5()
{
    document.getElementById("de5").style.backgroundColor="#a00ba2";
    document.getElementById("de5").style.color="white";
    document.getElementById("de2").style.backgroundColor="white";
    document.getElementById("de2").style.color="#a00ba2";
    document.getElementById("de3").style.backgroundColor="white";
    document.getElementById("de3").style.color="#a00ba2";
    document.getElementById("de1").style.backgroundColor="white";
    document.getElementById("de1").style.color="#a00ba2";
    document.getElementById("de4").style.backgroundColor="white";
    document.getElementById("de4").style.color="#a00ba2";
    document.getElementById("de6").style.backgroundColor="white";
    document.getElementById("de6").style.color="#a00ba2";
    document.getElementById("de7").style.backgroundColor="white";
    document.getElementById("de7").style.color="#a00ba2";
    //document.getElementById("but").style.display="none";
    document.getElementById("div1").style.display="none";
    document.getElementById("div2").style.display="none";
    document.getElementById("div3").style.display="none";
    document.getElementById("div4").style.display="none";
    document.getElementById("div5").style.display="block";
    document.getElementById("div6").style.display="none";
    document.getElementById("div7").style.display="none";
    document.getElementById("div8").style.display="none";
    document.getElementById("div9").style.display="none";
    document.getElementById("div10").style.display="none";
    document.getElementById("div11").style.display="none";
    document.getElementById("div12").style.display="none";
    document.getElementById("div13").style.display="none";
    document.getElementById("de5").style.opacity=1;
    g=5;
}

```

```

function myf6()
{
  document.getElementById("de6").style.backgroundColor="#a00ba2";
  document.getElementById("de6").style.color="white";
  document.getElementById("de2").style.backgroundColor="white";
  document.getElementById("de2").style.color="#a00ba2";
  document.getElementById("de3").style.backgroundColor="white";
  document.getElementById("de3").style.color="#a00ba2";
  document.getElementById("de1").style.backgroundColor="white";
  document.getElementById("de1").style.color="#a00ba2";
  document.getElementById("de4").style.backgroundColor="white";
  document.getElementById("de4").style.color="#a00ba2";
  document.getElementById("de5").style.backgroundColor="white";
  document.getElementById("de5").style.color="#a00ba2";
  document.getElementById("de7").style.backgroundColor="white";
  document.getElementById("de7").style.color="#a00ba2";
  document.getElementById("div1").style.display="none";
  document.getElementById("div2").style.display="none";
  document.getElementById("div3").style.display="none";
  document.getElementById("div4").style.display="none";
  document.getElementById("div5").style.display="none";
  document.getElementById("div6").style.display="none";
  document.getElementById("div7").style.display="block";
  document.getElementById("div8").style.display="none";
  document.getElementById("div9").style.display="none";
  document.getElementById("div10").style.display="none";
  document.getElementById("div11").style.display="none";
  document.getElementById("div12").style.display="none";
  document.getElementById("div13").style.display="none";
  document.getElementById("de6").style.opacity=1;
  g=6;
}
function myf7()
{
  document.getElementById("div1").style.display="none";
  document.getElementById("div2").style.display="none";
  document.getElementById("div3").style.display="none";
  document.getElementById("div4").style.display="none";
  document.getElementById("div5").style.display="none";
  document.getElementById("div6").style.display="none";
  document.getElementById("div7").style.display="none";
  document.getElementById("div8").style.display="none";
  document.getElementById("div9").style.display="none";
  document.getElementById("div10").style.display="block";
  document.getElementById("div11").style.display="none";
  document.getElementById("div12").style.display="none";
  document.getElementById("div13").style.display="none";
  document.getElementById("but").style.display="none";
  var r2=document.getElementById("fd").value;
  var er=[];
  for (var i = 1; i <=r2 ; i++)

```

```

    {
      er[i]=document.getElementById('uj'+i).value;
      document.getElementById('jju'+er[i]+i).selected=true;
    }
    g=9;
  }
function myf8()
  {
    document.getElementById("div1").style.display="none";
    document.getElementById("div2").style.display="none";
    document.getElementById("div3").style.display="none";
    document.getElementById("div4").style.display="none";
    document.getElementById("div5").style.display="none";
    document.getElementById("div6").style.display="none";
    document.getElementById("div7").style.display="none";
    document.getElementById("div8").style.display="none";
    document.getElementById("div9").style.display="none";
    document.getElementById("div10").style.display="none";
    document.getElementById("div11").style.display="block";
    document.getElementById("div12").style.display="none";
    document.getElementById("div13").style.display="none";
    g=10;
  }
function myf9()
  {
    document.getElementById("div1").style.display="none";
    document.getElementById("div2").style.display="none";
    document.getElementById("div3").style.display="none";
    document.getElementById("div4").style.display="none";
    document.getElementById("div5").style.display="none";
    document.getElementById("div6").style.display="none";
    document.getElementById("div7").style.display="none";
    document.getElementById("div8").style.display="none";
    document.getElementById("div9").style.display="none";
    document.getElementById("div10").style.display="none";
    document.getElementById("div11").style.display="none";
    document.getElementById("div12").style.display="block";
    document.getElementById("div13").style.display="none";
    g=11;
  }
function myf10()
  {
    document.getElementById("div31").style.display="block";
    document.getElementById("div32").style.display="none";
    document.getElementById("div33").style.display="none";
  }
function myf11()
  {
    document.getElementById("div31").style.display="none";
    document.getElementById("div32").style.display="block";
    document.getElementById("div33").style.display="none";
  }

```

```

}
function myf12()
{
    document.getElementById("diva1").style.display="none";
    document.getElementById("diva2").style.display="none";
    document.getElementById("diva3").style.display="block";
}
function myff13()
{
    document.getElementById("divb1").style.display="none";
    document.getElementById("divb2").style.display="block";
    document.getElementById("divb3").style.display="none";
    document.getElementById("divb4").style.display="none";
}
function myff14()
{
    document.getElementById("divb1").style.display="none";
    document.getElementById("divb2").style.display="none";
    document.getElementById("divb3").style.display="block";
    document.getElementById("divb4").style.display="none";
}
function myff15()
{
    document.getElementById("divb1").style.display="none";
    document.getElementById("divb2").style.display="none";
    document.getElementById("divb3").style.display="none";
    document.getElementById("divb4").style.display="block";
}
function ta1()
{
    document.getElementById('ku').submit();
}
function ta2()
{
    document.getElementById('ky').submit();
}
function ta3()
{
    document.getElementById('kg').submit();
}
function ta4()
{
    document.getElementById('ki').submit();
}
function ta5()
{
    document.getElementById("div1").style.display="none";
    document.getElementById("div2").style.display="none";
    document.getElementById("div3").style.display="none";
    document.getElementById("div4").style.display="none";
    document.getElementById("div5").style.display="none";
}

```

```

document.getElementById("div6").style.display="block";
document.getElementById("div7").style.display="none";
document.getElementById("div8").style.display="none";
document.getElementById("div9").style.display="none";
document.getElementById("div10").style.display="none";
document.getElementById("div11").style.display="none";
document.getElementById("div12").style.display="none";
document.getElementById("div13").style.display="none";
}
function er($event)
{
}
function myfun1()
{
    document.getElementById("flb").submit();
}
</script>
</head>
<body onload="" >
    <div class="d1" style="top:162px;" onclick="myf1()" id='de1' onMouseMove="mf1()"
onMouseOut="f1()">
        <div style="position:relative;left:20px;top:8px;">Оформить заказ</div>
    </div>
    <div class="d1" style="top:222px;" onclick="myf2()" id='de2' onMouseMove="mf2()"
onMouseOut="f2()">
        <div style="position:relative;left:20px;top:8px;">Мои заказы</div>
    </div>
    <div class="d1" style="top:282px;" onclick="myf3()" id='de3' onMouseMove="mf3()"
onMouseOut="f3()">
        <div style="position:relative;left:20px;top:8px;">Расчет себестоимости</div>
    </div>
    <?php
    if ($_SESSION['n_man']==1)
    echo '<div class="d1" style="top:342px;" onclick="myf4()" id="de4"
onMouseMove="mf4()" onMouseOut="f4()">
        <div style="position:relative;left:20px;top:8px;">Поступление</div>
    </div>
    <div class="d1" style="top:402px;" onclick="myf5()" id="de5" onMouseMove="mf5()"
onMouseOut="f5()">
        <div style="position:relative;left:20px;top:8px;">Производство</div>
    </div>
    <div class="d1" style="top:462px;" onclick="myf6()" id="de6" onMouseMove="mf6()"
onMouseOut="f6()">
        <div style="position:relative;left:20px;top:8px;">Списание</div>
    </div>
    <div class="d1" style="top:522px;" onclick="myf7()" id="de7" onMouseMove="mf7()"
onMouseOut="f7()">
        <div style="position:relative;left:20px;top:8px;">Сборка</div>
    </div>;
else

```

```

    echo '<div class="d1" style="top:342px;display:none;" onclick="myf4()" id="de4"
onMouseMove="mf4()" onMouseOut="f4()">
    <div style="position:relative;left:20px;top:8px;display:none;">Поступление</div>
</div>
    <div class="d1" style="top:402px;display:none;" onclick="myf5()" id="de5"
onMouseMove="mf5()" onMouseOut="f5()">
    <div style="position:relative;left:20px;top:8px;display:none;">Производство</div>
</div>
    <div class="d1" style="top:462px;display:none;" onclick="myf6()" id="de6"
onMouseMove="mf6()" onMouseOut="f6()">
    <div style="position:relative;left:20px;top:8px;display:none;">Списание</div>
</div>
    <div class="d1" style="top:522px;" onclick="myf7()" id="de7" onMouseMove="mf7()"
onMouseOut="f7()">
    <div style="position:relative;left:20px;top:8px;">Сборка</div>
</div>;
?>
<div style="position:fixed;left:0px;top:68px;">
    
</div>
<div style="position:fixed;left:160px;top:70px;">
    
</div>
<div id='div1' style="position:absolute; top:83px; left:23%;
width:74%;background-color:white;display:none;">
<form action="iframe1.php" target="ifrm1" method=get id='fla'>
<select id="im1" name="im1" class="dd2" onChange="myf111()" style="left:40px;">
<?php
    $form = ActiveForm::begin();
    echo $form->field($model, 'status')->dropDownList([
        '0'=>'Группы товаров',
    ]);
    ActiveForm::end();
?>
<?php
    $result=pg_query($connection,"select * from groups;");
    $r=pg_num_rows($result);
    for ($i=0; $i < $r; $i++)
    {
        $rows=pg_fetch_row($result,$i);
        echo $form->field(".$rows[0].">".$rows[1].");
    }
?>
</select>
</form>
<form action="iframe1.php" target="ifrm1" method=get id='fla1'>
<select id="im2" name="im2" class="dd2" onChange="myf112()" style="left:260px;">
<?php
    $form = ActiveForm::begin();
    echo $form->field($model, 'status')->dropDownList([
        '0'=>'Виды товаров',

```

```

    );
    ActiveForm::end();
    ?>
<?php
    $result=pg_query($connection,"select * from kinds;");
    $r=pg_num_rows($result);
    for ($i=0; $i < $r; $i++)
    {
        $rows=pg_fetch_row($result,$i);
        echo $form->field(".$rows[0].">".$rows[1].");
    }
?>
</select>
</form>
<form action="iframe1.php" target="ifr1" method=get id='fla2'>
    <select id="im3" name="im3" class="dd2" onChange="myf113()"
style="left:480px;width:270px;">
    <?php
        $form = ActiveForm::begin();
        echo $form->field($model, 'status')->dropDownList([
            '0'=>'Готовые комплекты',
        ]);
        ActiveForm::end();
    ?>
<?php
    $result=pg_query($connection,"select * from complects;");
    $r=pg_num_rows($result);
    for ($i=0; $i < $r; $i++)
    {
        $rows=pg_fetch_row($result,$i);

        echo $form->field(".$rows[0].">".$rows[1].");
    }
?>
</select>
</form>
<iframe name='ifr1' id='ifr1' style="border:none;position:absolute;
width:100%;top:95px;height:900px" onLoad="">
</iframe>
</div> <!--div1-->
<div id='div2' style="position:absolute; top:83px; left:22%; width:70%;background-
color:white;display:none;">
    <form action="iframe2.php" target="ifr2" method=get id='fl'>
    <select id="it1" name="it1" class="dd2" onChange="mf111()" style="left:40px;">
    <?php
        $form = ActiveForm::begin();
        echo $form->field($model, 'status')->dropDownList([
            '0'=>'Выберите клиента',
        ]);
        ActiveForm::end();
    ?>

```

```

<?php
    $result=pg_query($connection,"select * from klients;");
    $r=pg_num_rows($result);
    for ($i=0; $i < $r; $i++)
    {
        $rows=pg_fetch_row($result,$i);
        echo $form->field(".$rows[0].">".$rows[1].");
    }
?>
</select>
</form>
<form action="iframe2.php" target="ifrm2" method=get id='fl1'>
    <select id="it2" name="it2" class="dd2" onChange="mf112()" style="left:260px;">
        <?php
            $form = ActiveForm::begin();
            echo $form->field($model, 'status')->dropDownList([
                '0'=>'Выберите менеджера',
            ]);
            ActiveForm::end();
        ?>

    <?php
        $result=pg_query($connection,"select * from managers;");
        $r=pg_num_rows($result);
        for ($i=0; $i < $r; $i++)
        {
            $rows=pg_fetch_row($result,$i);
            echo $form->field(".$rows[0].">".$rows[1].");
        }
    ?>
</select>
</form>
<form action="iframe2.php" target="ifrm2" method=get id='fl2'>
    <select id="it3" name="it3" class="dd2" onChange="mf113()"
style="left:480px;width:270px;">
        <?php
            $form = ActiveForm::begin();
            echo $form->field($model, 'status')->dropDownList([
                '0'=>'Выберите статус',
            ]);
            ActiveForm::end();
        ?>

    <?php
        $result=pg_query($connection,"select * from status;");
        $r=pg_num_rows($result);
        for ($i=0; $i < $r; $i++)
        {
            $rows=pg_fetch_row($result,$i);
            echo $form->field(".$rows[0].">".$rows[1].");
        }
    ?>
</select>

```

```

</form>
<iframe name='ifrm2' id='ifrm2' style="border:none;position:absolute;
width:100%;top:95px;height:900px" onLoad="">
</iframe>
</div> <!--div2-->
<div id='div3' style="position:absolute; top:103px; left:23%; width:70%;background-
color:white;display:none;">
  <div class="d3" onclick="myf41()">
    <div style="position:relative;top:5px; left:50px;">
      По виду товара</div>
    </div>
    <form action="iframe3.php" target="ifrm3" method=get id='flb'>
    <input type=text name='rtr1' id="rtr1" class=i1 style="width:220px;left:40px;
position:absolute;top:60px;" onKeyUp='myfun1()'>
    <div class="d3" onclick="myf42()" style="left:270px;">
      <div style="position:relative;top:5px; left:50px;">По товару</div>
    </div>
    <input type=text name=rtr2 id="rtr2" class=i1 style="width:220px;left:270px;
position:absolute;top:60px;" onKeyUp='myfun1()'>
    <div class="d3" onclick="myf43()" style="left:500px;">
      <div style="position:relative;top:5px; left:50px;">
        По материалу</div>
      </div>
    <input type=text name=rtr3 id="rtr3" class=i1 style="width:220px;left:500px;
position:absolute;top:60px;" onKeyPress="fkey1(event)" onKeyUp='myfun1()'>
    </form>
    <iframe name='ifrm3' id='ifrm3' style="border:none;position:absolute;
width:100%;top:95px;height:5000px" onLoad="">
  </iframe>
</div> <!--div3-->
<div id='div4' style="position:absolute; top:103px; left:23%; width:76%;background-
color:white;display:none;">
  <div style="position:relative;left:20%;">
    <H3 style="color:#a00ba2;">ПОСТУПЛЕНИЕ МАТЕРИАЛОВ</H3><br>
  </div>
<div id='divc1' style="position:absolute;top:45px;" >
  <table>
  <tr>
    <td class="td1">№</td>
    <td class="td1">Вид</td>
    <td class="td1">Наименование</td>
    <td class="td1">Ед.изм</td>
    <td class="td1">Цена</td>
    <td class="td1">Размер</td>
    <td class="td1">Цвет</td>
    <td class="td1">Дата</td>
    <td class="td1">Кол-во</td>
  </tr>
</div>
<?php
$result=pg_query($dbconn,"SELECT
name_k_mat,naim_mat,ed_izm,price,size,color,data,count

```

```

FROM
kinds_mat, materials, deliver
WHERE
kinds_mat.n_k_mat = materials.n_k_mat AND materials.n_mat = deliver.n_mat;
");
//$result=pg_query($dbconn,"select * from products;");
$r=pg_num_rows($result);
for ($i=0; $i < $r; $i++)
{ $rows=pg_fetch_row($result,$i);
  $a[$i]['name_k_mat']=$rows[0];
  $a[$i]['naim_mat']=$rows[1];
  $a[$i]['ed_izm']=$rows[2];
  $a[$i]['price']=$rows[3];
  $a[$i]['size']=$rows[4];
  $a[$i]['color']=$rows[5];
  $a[$i]['data']=$rows[6];
  $a[$i]['count']=$rows[7];
} $n=$i;
$i=0;
while($i<$n)
{
  echo '<tr>';
  echo '<td>'.Yii::t($i+1).'\</td>';
  echo '<td>'.Yii::t$a[$i]['name_k_mat'].'\</td>';
  echo '<td>'.Yii::t$a[$i]['naim_mat'].'\</td>';
  echo '<td>'.Yii::t$a[$i]['ed_izm'].'\</td>';
  echo '<td>'.Yii::t$a[$i]['price'].'\</td>';
  echo '<td>'.Yii::t$a[$i]['size'].'\</td>';
  echo '<td>'.Yii::t$a[$i]['color'].'\</td>';
  echo '<td>'.Yii::t$a[$i]['data'].'\</td>';
  echo '<td>'.Yii::t$a[$i]['count'].'\</td>';
  echo '</tr>';
  $i=$i+1;
} ?>
</table>
</div>
<div style="position:absolute;top:440px;left:40px;font-family:'Bookman Old Style';font-size:18px;color:#d86ecd;text-align:center;">
  Поступление нового материала
</div>
<div style="color:#a00ba2;position:absolute;top:430px;left:20px;">
  <div style="font-family:Bookman Old Style;
  color:#a00ba2;position:absolute;top:40px;left:10px;">
    Вид:
  </div>
  <div style="color:#a00ba2;position:absolute;top:40px;left:150px;">
  <select style="position:absolute;left:0px;top:0px;
  width:173px;color:#d86ecd;background-color:white;font-family:Bookman Old
Style;">
  <option value=1>Основной</option>
  <option value=2>Вспомогательный</option>

```

```

</select>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:70px;left:10px;">
    Ед.измер.:
</div>
<div style="color:#a00ba2;position:absolute;top:70px;left:150px;">
    <input type=text name='edt6' class='i1' id='p'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:100px;left:10px;">
    Размер:
</div>
<div style="color:#a00ba2;position:absolute;top:100px;left:150px;">
    <input type=text name='edt8' class='i1' id='logg1'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:40px;left:360px;">
    Наименование:
</div>
<div style="color:#a00ba2;position:absolute;top:40px;left:500px;">
    <input type=text name='edt5' class='i1' id='o' >
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:70px;left:360px;">
    Цена:
</div>
<div style="color:#a00ba2;position:absolute;top:70px;left:500px;">
    <input type=text name='edt7' class='i1' id='m'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:100px;left:360px;">
    Цвет:
</div>
<div style="color:#a00ba2;position:absolute;top:100px;left:500px;">
    <input type=text name='edt9' class='i1' id='pass1'>
</div>
    <input style="width:170px;position:absolute;left:150px;top:130px;height:20px;"
type=submit class=i1 value="Сохранить" onClick="">
</div>
<!--div4-->
<div id='div5' style="position:absolute; top:83px; left:23%; width:63%;background-
color:white;display:none;">
<div style="position:relative;left:20%;">
    <H3 style="color:#a00ba2;">ПРОИЗВОДСТВО ПРОДУКЦИИ</H3><br>
</div>
<div id='divc1' style="position:absolute;top:45px;" >
<table>
<tr>
<td class="td1">№</td>
<td class="td1">Группа</td>

```

```

<td class="td1">Вид</td>
<td class="td1">КОМПЛЕКТ</td>
<td class="td1">Наименование</td>
<td class="td1">Цена</td>
<td class="td1">Цена в компл.</td>
<td class="td1">Размер</td>
<td class="td1">Цвет</td>
<td class="td1">Дата</td>
<td class="td1">Кол-во</td>
</tr>
<?php
$result=pg_query($dbconn,"SELECT name_gr,name_kin,name_kom,naim_prod,
size,color,price,price1, count,data
FROM proizvodstvo,products,kinds,complects,groups
WHERE products.n_prod = proizvodstvo.n_prod AND kinds.n_kin = products.n_kin AND
complects.n_kom = products.n_kom AND groups.n_gr = products.n_gr;");
$r=pg_num_rows($result);
for ($i=0; $i < $r; $i++)
{ $rows=pg_fetch_row($result,$i);
  $a[$i]['name_gr']=$rows[0];
  $a[$i]['name_kin']=$rows[1];
  $a[$i]['name_kom']=$rows[2];
  $a[$i]['naim_prod']=$rows[3];
  $a[$i]['size']=$rows[4];
  $a[$i]['color']=$rows[5];
  $a[$i]['price']=$rows[6];
  $a[$i]['price1']=$rows[7];
  $a[$i]['count']=$rows[8];
  $a[$i]['data']=$rows[9];
}$n=$i;
$i=0;
while($i<$n)
{
  echo '<tr>';
  echo '<td>'.Yii::t ($i+1).</td>';
  echo '<td>'.Yii::t $a[$i]['name_gr'].</td>';
  echo '<td>'.Yii::t $a[$i]['name_kin'].</td>';
  echo '<td>'.Yii::t $a[$i]['name_kom'].</td>';
  echo '<td>'.Yii::t $a[$i]['naim_prod'].</td>';
  echo '<td>'.Yii::t $a[$i]['size'].</td>';
  echo '<td>'.Yii::t $a[$i]['color'].</td>';
  echo '<td>'.Yii::t $a[$i]['price'].</td>';
  echo '<td>'.Yii::t $a[$i]['price1'].</td>';
  echo '<td>'.Yii::t $a[$i]['data'].</td>';
  echo '<td>'.Yii::t $a[$i]['count'].</td>';
  echo '</tr>';
  $i=$i+1;
}??>
</table>
</div>

```

```
div style="position:absolute;top:940px;left:40px;font-family:'Bookman Old Style';font-size:18px;color:#d86ecd;text-align:center;">
```

```
    Производство нового продукта
```

```
</div>
```

```
<div style="color:#a00ba2;position:absolute;top:930px;left:20px;">
```

```
    <div style="font-family:Bookman Old Style;
        color:#a00ba2;position:absolute;top:40px;left:10px;">
```

```
        Группа:
```

```
    </div>
```

```
    <div style="color:#a00ba2;position:absolute;top:40px;left:150px;">
```

```
<select style="position:absolute;left:0px;top:0px;
width:173px;color:#d86ecd;background-color:white;font-family:Bookman Old
Style;">
```

```
    <?php
```

```
    $form = ActiveForm::begin();
    echo $form->field($model, 'status')->dropDownList([
        '1'=>'Стенка Джорджия',
        '2'=>'Спальни',
        '3'=>'Детские'
        '4'=>'Офисная мебель'
        '5'=>'Гостиные'
        '6'=>'Прихожие'
        '7'=>'Ванные гарнитуры'
    ]);
    ActiveForm::end();
    ?>
```

```
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:40px;left:360px;">
```

```
    Вид:
```

```
</div>
```

```
<div style="color:#a00ba2;position:absolute;position:absolute;top:40px;left:500px;">
```

```
<select style="position:absolute;left:0px;top:0px;
width:173px;color:#d86ecd;background-color:white;font-family:Bookman Old
Style;">
```

```
<?php
```

```
$form = ActiveForm::begin();
echo $form->field($model, 'status')->dropDownList([
    '1'=>'Шкафы',
    '2'=>'Кресла',
    '3'=>'Столлы'
    '4'=>'Диваны'
    '5'=>'Кровати'
    '6'=>'Комоды'
    '7'=>'Полки'
    '8'=>'Зеркала'
]);
ActiveForm::end();
?>
```

```
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:70px;left:10px;">
```

```
    Комплект:
```

```

</div>
<div style="color:#a00ba2;position:absolute;position:absolute;
top:70px;left:150px;">
<select style="position:absolute;left:0px;top:0px;
width:173px;color:#d86ecd;background-color:white;font-family:Bookman Old
Style;">
<?php
$form = ActiveForm::begin();
echo $form->field($model, 'status')->dropDownList([
'1'=>'Стенка Джорджия',
'2'=>'Спальни',
'3'=>'Детские'
'4'=>'Офисная мебель'
'5'=>'Гостиные'
'6'=>'Прихожие'
'7'=>'Ванные гарнитуры'
]);
ActiveForm::end();
?>
</select>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:70px;left:360px;">
Наименование:
</div>
<div style="color:#a00ba2;position:absolute;top:70px;left:500px;">
<input type=text name='edt7' class='i1' id='m'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:100px;left:10px;">
Цена:
</div>
<div style="color:#a00ba2;position:absolute;top:100px;left:150px;">
<input type=text name='edt8' class='i1' id='logg1'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:100px;left:360px;">
Цена_в_ком:
</div>
<div style="color:#a00ba2;position:absolute;top:100px;left:500px;">
<input type=text name='edt9' class='i1' id='pass1'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:130px;left:10px;">
Размер:
</div>
<div style="color:#a00ba2;position:absolute;top:130px;left:150px;">
<input type=text name='edt8' class='i1' id='logg1'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:130px;left:360px;">

```

```

    Цвет:
</div>
<div style="color:#a00ba2;position:absolute;top:130px;left:500px;">
  <input type=text name='edt9' class='i1' id='pass1'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:160px;left:10px;">
  Дата:
</div>
<div style="color:#a00ba2;position:absolute;top:160px;left:150px;">
  <input type=text name='edt8' class='i1' id='logg1'>
</div>
<div style="font-family:Bookman Old Style;
color:#a00ba2;position:absolute;top:160px;left:360px;">
  Количество:
</div>
<div style="color:#a00ba2;position:absolute;top:160px;left:500px;">
  <input type=text name='edt9' class='i1' id='pass1'>
</div>
  <input style="width:170px;position:absolute;left:150px;top:190px;height:20px;"
type=submit class=i1 value="Сохранить" onClick="">
</div>
</div> <!--div5-->
<div id='div6' style="position:absolute; top:83px; left:23%; width:62%;background-
color:white;display:none;height:100%;">
<div style="position:relative;left:30%;">
  <H3 style="color:#a00ba2;">УЧЕТ МАТЕРИАЛОВ</H3><br>
</div>
<div id='div7' style="position:absolute; top:83px; left:23%; width:55%;background-
color:white;display:none;">
<div id='div8' style="position:absolute; top:116px; left:23%; width:75%;background-
color:white;display:none;">
<div id='div9' style="position:absolute; top:103px; left:23%; width:55%;background-
color:white;display:none;">
<div id='div10' style="position:absolute; top:133px; left:25%; width:65%;background-
color:white;display:none;">
<div id='div11' style="position:absolute; top:113px; left:27%; width:56%;background-
color:white;display:none;">
  <div style="position:relative;left:30%;">
    <H3 style="color:#a00ba2;"> СХЕМА РАБОТЫ</H3><br>
  </div>
<div id='div12' style="position:absolute; top:113px; left:27%; width:55%;background-
color:white;display:none;">
<div style="position:relative;left:10%;">
  <H3 style="color:#a00ba2;"> КОНТАКТНЫЕ ДАННЫЕ</H3><br>
</div> <br>
<OL style="font-size:16px;font-family:'Bookman Old Style';">
  <LI>Краснодарский Край
  <UL type="square" style="color:#d86ecd">
    <LI><span style="font-family:'Times New Roman';">352900 г. Армавир, ул. Чичерина,
61; p.т. <font color='#a00ba2'>8 (86-137) 479-58</font></span></LI>

```

<LI><span style="font-family:'Times New Roman';">353440 г. Анапа, ул. Шевченко д. 237 ; п.т. <font color='#a00ba2'>8 ( 86-133) 210-65</font></span></LI>

<LI><span style="font-family:'Times New Roman';">353465 г. Геленджик, ул. Горького, 26; п.т. <font color='#a00ba2'>8 (86-141) 329-41</font></span></LI>

<LI><span style="font-family:'Times New Roman';">353290 г. Горячий Ключ, ул. Ленина, д. 57; п.т. <font color='#a00ba2'>8 (86-159) 350-84</font></span></LI>

<LI><span style="font-family:'Times New Roman';">353907 г. Новороссийск, ул. Луначарского 19 ; п.т. <font color='#a00ba2'>8 ( 86-17) 301-681, 261-228 , 26-01-51</font></span></LI>

</UL>

</li>

<LI>г.Сочи

<UL type="square" style="color:#d86ecd">

<LI><span style="font-family:'Times New Roman';">353323 Абинск ул. Советов 128 ; п.т. <font color='#a00ba2'>8 (86-150) 533-09 </font></span></LI>

<LI><span style="font-family:'Times New Roman';">352690 г. Апшеронск, ул. Ворошилова 52 ; п.т. <font color='#a00ba2'>8 (86-152) 274-95</font></span></LI>

<LI><span style="font-family:'Times New Roman';">353040 с. Белая Глина, ул. Первомайская д. 149 ; п.т. <font color='#a00ba2'>8 (86-154) 729-14</font></span></LI>

</UL>

</li>

<LI>Ленинградский район - подразделения УФМС

<UL type="square" style="color:#d86ecd">

<LI><span style="font-family:'Times New Roman';">353740 ст. Ленинградская, ул. Набережная д. 88; п.т. <font color='#a00ba2'>8 ( 86-145) 379-29</font></span></LI>

<LI><span style="font-family:'Times New Roman';">352500 Лабинск, ул. Шевченко д. 80 ; п.т. <font color='#a00ba2'>8 (86-169) 746-91</font></span></LI>

<LI><span style="font-family:'Times New Roman';">352570 пос. Мостовской ; п.т. <font color='#a00ba2'>8 (86-192) 547-28 </font></span></LI>

<LI><span style="font-family:'Times New Roman';">353680 г. Ейск, ул. Баррикадная, 1; п.т. <font color='#a00ba2'>8 (86-132) 733-40</font></span></LI>

<LI><span style="font-family:'Times New Roman';">353020 ст. Новопокровская, ул. Ленина, д. 97; п.т. <font color='#a00ba2'>8 (86-149) 712-90</font></span></LI>

<LI><span style="font-family:'Times New Roman';">352240 Новокубанск, ул. Тюленина 82 ; п.т. <font color='#a00ba2'>8 (86195) 4-18-41 </font></span></LI>

<LI><span style="font-family:'Times New Roman';">353620 ст. Староцербиновская, ул. Советов 25; п.т.<font color='#a00ba2'>8 (86-151) 449-88</font></span></LI>

</UL>

</li>

</ol>

</div> <!--div12-->

<div id='div13' style="position:absolute; top:113px; left:27%; width:55%;background-color:white;display:none;">

</div> <!--div13-->

<div class="line1" >

<div style="position:absolute;left:1025px;width:372px;height:30px; background:#a00ba2;border:1px solid #d86ecd;">

</div>

<div style="position:absolute;left:1024px;width:372px;height:29px; background:#a00ba2;border:1px solid #d86ecd;">

<div style="position:absolute;left:10px;top:5px;width:98px;top::">

```

    </div>
</div>
<div style="position:absolute;left:927px;width:98px;height:30px;
background:#a00ba2;border:1px solid #d86ecd;">
</div>
<a href="registr.php">
<div id="fu1" style="position:absolute;left:927px;width:97px;height:29px;
background:#a00ba2;border:1px solid #d86ecd;cursor:pointer;"
onMouseMove="mfu1(1)" onMouseOut="mfun1(1)">
<div id="fun1" style="position:absolute;left:9px;top:5px;width:98px;text-align:center;
text-decoration:none;color:#ffffff;">
    Выход
</div>
</div>
</a>
<div style="position:absolute;left:829px;width:25px;height:30px;
background:#a00ba2;border:1px solid #d86ecd;">
</div>
<div id="fu2" style="position:absolute;left:829px;width:114px;height:29px;
background:#a00ba2;border:1px solid #d86ecd;cursor:pointer;"
onMouseMove="mfu1(2)" onMouseOut="mfun1(2)" onClick="myf11()">
<div id="fun2" style="position:absolute;left:12px;top:5px;width:115px;text-
align:center;
text-decoration:none;color:#ffffff;">
    Контакты
</div>
</div>
<div style="position:absolute;left:714px;width:140px;height:30px;
background:#a00ba2;border:1px solid #d86ecd;">
</div>
<div id="fu3" style="position:absolute;left:714px;width:139px;height:29px;
background:#a00ba2;border:1px solid #d86ecd;cursor:pointer;"
onMouseMove="mfu1(3)" onMouseOut="mfun1(3)" onClick="myf9()">
<div id="fun3" style="position:absolute;left:7px;top:5px;width:139px;text-
align:center;">
    Руководство
</div>
</div>
<div style="position:absolute;left:574px;width:153px;height:30px;
background:#a00ba2;border:1px solid #d86ecd;">
</div>
<div id="fu4" style="position:absolute;left:574px;width:152px;height:29px;
background:#a00ba2;border:1px solid #d86ecd;cursor:pointer;"
onMouseMove="mfu1(4)" onMouseOut="mfun1(4)" onClick="myf10()">
<div id="fun4" style="position:absolute;top:5px;width:152px;text-align:center;">
    Схема работы
</div>
</div>
<div style="position:absolute;left:363px;width:211px;height:30px;
background:#a00ba2;border:1px solid #d86ecd;">
</div>

```



## Файл iframe1.php

```

<?php
use yii\web\Session;
$session = Yii::$app->session;
?>
<?php
use yii\helpers\Html;
use yii\widgets\ActiveForm;
use app\modules\commerce\models\UtpOrder;
use kartik\select2>Select2;
?>
<script>
function gr_ss(kl)
{
    document.getElementById('parent_popup').style.display="block";
}
</script>
<style>
input.i1 { font-family: Bookman Old Style; font-size:12px; color:#d86ecd; font-weight:bold;
font-style: italic;}
.s1 {background-color:#a00ba2; color:white; font-weight:normal;}
.s2 {background-color:white; color: #a00ba2; font-weight:normal;}
.ph { color:#a00ba2; font-family: Bookman old style; font-size:18px; font-weight:normal;
    position: relative;top: 30px;}
div.d1 {background-color:#f4072d; color: white; position:fixed;width:20%;height:40px;
left:0%;
    top:50px;box-shadow: 10px 10px 10px #a00ba2;border-top-right-radius: 30px;font-family:
Bookman Old Style; }
    div.d2 {background-color:white; color: #f4072d;
position:absolute;width:220px;height:55px; left:40px;
    top:10px;box-shadow: 10px 10px 10px #a00ba2;border:solid 2px #a00ba2;font-family:
Bookman Old Style;
    font-size: 14px;}
    div.d3 {background-color:white; color: #f4072d;
position:absolute;width:220px;height:30px; left:40px;
    top:10px;box-shadow: 10px 10px 10px #a00ba2;border:solid 2px #d86ecd;font-family:
Bookman Old Style;
    font-size: 14px;}
    body { background: url(WB02073_.GIF)}
    .radio {box-shadow: 8px 8px 8px #a00ba2; display: inline;position: relative;top: 7px; left:
5px;}
    .checkB {background-color:#a00ba2; color: white; font-weight:bold;display: inline;position:
relative;top: 7px;left: 50px;}
    H3 {color:#d86ecd; font-family: Bookman old style; font-size:18px; font-
weight:normal;box-shadow: 8px 8px 8px #a00ba2;
    display: inline; position: relative;top: 0px;left: 0px;margin: 0; padding: 0;}
    select { font-weight:bold;color:white;background-color:#d86ecd;}
    .td1 {
        border: 2px solid #d86ecd;
        background-color: #d86ecd;
        box-shadow: 10px 10px 10px #d86ecd;

```

```

font-family: Bookman Old Style;
font-size: 16px;
text-align: center;
color: white;
}
td{
border: 1px solid #d86ecd;
background-color:white;
box-shadow: 10px 10px 10px #d86ecd;
font-family: Bookman Old Style;
font-size:14px;
color:#d86ecd;
}
#parent_popup {
height: 100%;
position: fixed;
width: 100%;
z-index: 100;
top: 0;
left: 0;
}
#popup {
background: url(WB02073_.GIF);
height: 635px;
position: fixed;
top: 0%;
left: 0%;
color: #d86ecd;
width: 90%;
border: #d86ecd 1px solid;
}
</style>
<meta charset="utf-8">
<link rel="stylesheet" type="text/css" href="styles.css">
<script type="text/javascript" src="js/jquery-3.1.1 (1).js"></script>
<script type="text/javascript" src="js/js.js"></script>
<script type="text/javascript" src="js/jquery-3.1.1.min (1).js"></script>
<script type="text/javascript" src="js/jquery-3.1.1.min (1).map"></script>
<script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.1/jquery.min.js"></script>
</head>
<body>
<div id='div41' style="position:absolute;top:0px;" >
<table>
<tr>
<td class="td1">№</td>
<td class="td1">Наименование</td>
<td class="td1">Цена</td>
<td class="td1">Цена в компл.</td>
<td class="td1">Размер</td>
<td class="td1">Цвет</td>

```

```

        <td class="td1">Выбрать</td>
    </tr>
<?php
    $dbconn=pg_connect("host=localhost dbname=furniture user=postgres port=5432
password=1");
    if ($dbconn===false)
    {
        echo "Error opening DB!<br>";
    } ;
    if (isset($_GET["im1"]))
    if ($_GET["im1"]!=0)
        $result=pg_query($dbconn,"select * from products where n_gr=".$_GET["im1"].");
    else $result=pg_query($dbconn,"select * from products;");
else if (isset($_GET["im2"]))
    if ($_GET["im2"]!=0)
        $result=pg_query($dbconn,"select * from products where n_kin=".$_GET["im2"].");
    else $result=pg_query($dbconn,"select * from products;");
    else if (isset($_GET["im3"]))
    if ($_GET["im3"]!=0)
        $result=pg_query($dbconn,"select * from products where n_kom=".$_GET["im3"].");
    else $result=pg_query($dbconn,"select * from products;");
    $r=pg_num_rows($result);
    for ($i=0; $i < $r; $i++)
    { $rows=pg_fetch_row($result,$i);
        $a[$i]['name_prod']=$rows[4];
        $a[$i]['n_prod']=$rows[0];
        $a[$i]['price']=$rows[5];
        $a[$i]['price1']=$rows[6];
        $a[$i]['size']=$rows[8];
        $a[$i]['color']=$rows[9];
    }$n=$i;
    $i=0;
    while($i<$n)
    { echo '<form method=get target="ifrm11" action="is3.php"
name="ki'.($i+1).'" id="ki">';
        echo '<tr>';
        echo '<td>'.Yii::t($i+1).</td>';
        echo '<td>'.Yii::t($a[$i]['name_prod']).</td>';
        echo '<td>'.Yii::t($a[$i]['price']).</td>';
        echo '<td>'.Yii::t($a[$i]['price1']).</td>';
        echo '<td>'.Yii::t($a[$i]['size']).</td>';
        echo '<td>'.Yii::t($a[$i]['color']).</td>';
        echo '<td><input type=submit id="tum'.Yii::t($i+1).'" value=' Yii::t($i+1).'
style="background-color:white;
        width:90px;height:19px;color:#d86ecd;cursor:pointer;"
        onclick="gr_ss('.'.($i+1).')"></td>';
        echo '</tr>';
        echo '<input name="inp" id="inp'.Yii::t($i+1).'" type="hidden"
value='Yii::t($a[$i]['n_prod']).>';
        echo '</form>';
        $i=$i+1;
    }
}

```



## Файл iframe2.php

```

<?php
use yii\web\Session;
$session = Yii::$app->session;
?>
<head>
<script>
    function gr_ss(kl)
    {
        document.getElementById('parent_popup').style.display="block";
        document.getElementById('rrr').innerHTML=document.getElementById('inp'+kl).value;
    }
</script>
<style>
    input.i1 {font-family: Bookman Old Style; font-size:12px; color:#d86ecd; font-weight:bold;
font-style: italic;}
    .s1 {background-color:#a00ba2; color:white; font-weight:normal;}
    .s2 {background-color:white; color: #a00ba2; font-weight:normal;}
    .ph {color:#a00ba2; font-family: Bookman old style; font-size:18px; font-weight:normal;
        position: relative;top: 30px;}
    div.d1 {background-color:#f4072d; color: white; position:fixed;width:20%;height:40px;
left:0%;
        top:50px;box-shadow: 10px 10px 10px #a00ba2;border-top-right-radius: 30px;font-family:
Bookman Old Style; }
    div.d2 {background-color:white; color: #f4072d;
position:absolute;width:220px;height:55px; left:40px;
        top:10px;box-shadow: 10px 10px 10px #a00ba2;border:solid 2px #a00ba2;font-family:
Bookman Old Style;
        font-size: 14px;}
    div.d3 {background-color:white; color: #f4072d;
position:absolute;width:220px;height:30px; left:40px;
        top:10px;box-shadow: 10px 10px 10px #a00ba2;border:solid 2px #d86ecd;font-family:
Bookman Old Style;
        font-size: 14px;}
    body {background: url(WB02073_.GIF)}
    .radio {box-shadow: 8px 8px 8px #a00ba2; display: inline;position: relative;top: 7px; left:
5px;}
    .checkB {background-color:#a00ba2; color: white; font-weight:bold;display: inline;position:
relative;top: 7px;left: 50px;}
    H3 {color:#d86ecd; font-family: Bookman old style; font-size:18px; font-
weight:normal;box-shadow: 8px 8px 8px #a00ba2;
        display: inline; position: relative;top: 0px;left: 0px;margin: 0; padding: 0;}
    select {font-weight:bold;color:white;background-color:#d86ecd;}
    .td1 {
        border: 2px solid #d86ecd;
        background-color: #d86ecd;
        box-shadow: 10px 10px 10px #d86ecd;
        font-family: Bookman Old Style;
        font-size: 16px;
        text-align: center;
        color: white;

```

```

    }
    td{
        border: 1px solid #d86ecd;
        background-color:white;
        box-shadow: 10px 10px 10px #d86ecd;
        font-family: Bookman Old Style;
        font-size:14px;
        color:#d86ecd;
    }
    #parent_popup {
    height: 100%;
    position: fixed;
    width: 100%;
    z-index: 100;
    top: 0;
    left: 0;
    }
    #popup {
    background: url(WB02073_.GIF);
    height: 535px;
    position: fixed;
    top: 0%;
    left: 7%;
    color: #d86ecd;
    width: 86%;
    border: #d86ecd 1px solid;
    }
</style>
<meta charset="utf-8">
<link rel="stylesheet" type="text/css" href="styles.css">
<script type="text/javascript" src="js/jquery-3.1.1 (1).js"></script>
<script type="text/javascript" src="js/js.js"></script>
<script type="text/javascript" src="js/jquery-3.1.1.min (1).js"></script>
<script type="text/javascript" src="js/jquery-3.1.1.min (1).map"></script>
<script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.1/jquery.min.js"></script>
</head>
<body>
    <div id='div41' style="position:absolute;top:0px;" >
<table>
<tr>
    <td class="td1">№</td>
    <td class="td1">Номер</td>
    <td class="td1">Клиент</td>
    <td class="td1">Менеджер</td>
    <td class="td1">Статус</td>
    <td class="td1">Дата</td>
    <td class="td1">Показать</td>
</tr>
<?php
$connection=Yii::app()->db;

```

```

$connection=new CDbConnection("host=localhost dbname=furniture user=postgres
port=5432 password=1");
if ($connection===false)
{
    echo "Error opening DB!<br>";
} else echo "all ok";
if (isset($_GET["it1"]))
if ($_GET["it1"]!=0)
    $result=new CDbConnection($connection,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
    FROM orders, klients, managers, status
    WHERE
        klients.n_kl = orders.n_kl AND
        managers.n_man = orders.n_man AND
        status.n_stat = orders.n_stat and klients.n_kl=".$_GET["it1"].";");
else
    $result=new CDbConnection($connection,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
    FROM orders, klients, managers, status
    WHERE klients.n_kl = orders.n_kl AND
        managers.n_man = orders.n_man AND
        status.n_stat = orders.n_stat;");
else if (isset($_GET["it2"]))
if ($_GET["it2"]!=0)
    $result=new CDbConnection($connection,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
    FROM orders, klients, managers, status
    WHERE
        klients.n_kl = orders.n_kl AND
        managers.n_man = orders.n_man AND
        status.n_stat = orders.n_stat and managers.n_man=".$_GET["it2"].";");
else
    $result=new CDbConnection($connection,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
    FROM orders, klients, managers, status
    WHERE klients.n_kl = orders.n_kl AND
        managers.n_man = orders.n_man AND
        status.n_stat = orders.n_stat;");
else if (isset($_GET["it3"]))
if ($_GET["it3"]!=0)
    $result=new CDbConnection($connection,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
    FROM orders, klients, managers, status
    WHERE
        klients.n_kl = orders.n_kl AND
        managers.n_man = orders.n_man AND
        status.n_stat = orders.n_stat and status.n_stat=".$_GET["it3"].";");
else
    $result=new CDbConnection($connection,
    "SELECT data, nom_ord, klients.fio, managers.fio, name_stat
    FROM orders, klients, managers, status

```



```
        onLoad="">
</iframe>
    <form method=get target="shopping" action="shopping.php" name="by" id="by">
    <input type="submit" name='but1' id='but1' value="Распечатать"
style="position:absolute;left:480px;
    width:150px;top:448px;height:45px;background-
color:#d86ecd;color:white;cursor:pointer;
    font-family:'Bookman Old Style';border:none;font-size:16px;" onClick="er(event)">
    </form>
    <iframe name='shopping' id='shopping'
style="border:none;position:absolute;width:270px;top:570px;height:30px;
    left:400px" scrolling=no onLoad="">
    </iframe>
</div>
</div>
</div>
```

## Файл iframe3.php

```

<?php
use yii\web\Session;
$session = Yii::$app->session;
?>
<head>
  <script>
    function gr_s(kl)
    {
      document.getElementById('parent_popup').style.display="block";
document.getElementById('rrr').innerHTML=document.getElementById('naim_prod'+kl).innerH
TML;
    }
  </script>
  <style>
    input.i1 {font-family: Bookman Old Style; font-size:12px; color:#d86ecd; font-weight:bold;
font-style: italic;}
    .s1 {background-color:#a00ba2; color:white; font-weight:normal;}
    .s2 {background-color:white; color: #a00ba2; font-weight:normal;}
    .ph {color:#a00ba2; font-family: Bookman old style; font-size:18px; font-weight:normal;
    position: relative;top: 30px;}
    div.d1 {background-color:#f4072d; color: white; position:fixed;width:20%;height:40px;
left:0%;
    top:50px;box-shadow: 10px 10px 10px #a00ba2;border-top-right-radius: 30px;font-family:
Bookman Old Style; }
    div.d2 {background-color:white; color: #f4072d;
position:absolute;width:220px;height:55px; left:40px;
    top:10px;box-shadow: 10px 10px 10px #a00ba2;border:solid 2px #a00ba2;font-family:
Bookman Old Style;
    font-size: 14px;}
    div.d3 {background-color:white; color: #f4072d;
position:absolute;width:220px;height:30px; left:40px;
    top:10px;box-shadow: 10px 10px 10px #a00ba2;border:solid 2px #d86ecd;font-family:
Bookman Old Style;
    font-size: 14px;}
    body {background: url(WB02073_.GIF)}
    .radio {box-shadow: 8px 8px 8px #a00ba2; display: inline;position: relative;top: 7px; left:
5px;}
    .checkB {background-color:#a00ba2; color: white; font-weight:bold;display: inline;position:
relative;top: 7px;left: 50px;}
    H3 {color:#d86ecd; font-family: Bookman old style; font-size:18px; font-
weight:normal;box-shadow: 8px 8px 8px #006400;
    display: inline; position: relative;top: 0px;left: 0px;margin: 0; padding: 0;}
    select {font-weight:bold;color:white;background-color:#d86ecd;}
    .td1 {
      border: 2px solid #d86ecd;
      background-color: #d86ecd;
      box-shadow: 10px 10px 10px #d86ecd;
      font-family: Bookman Old Style;
      font-size: 16px;
      text-align: center;

```

```

    color: white;
  }
  td{
    border: 1px solid #d86ecd;
    background-color:white;
    box-shadow: 10px 10px 10px #d86ecd;
    font-family: Bookman Old Style;
    font-size:14px;
    color:#d86ecd;
  }
  #parent_popup {
  height: 100%;
  position: fixed;
  width: 100%;
  z-index: 100;
  top: 0;
  left: 0;
  }
  #popup {
  background: url(WB02073_.GIF);
  height: 635px;
  position: fixed;
  top: 0%;
  left: 7%;
  color: #d86ecd;
  width: 86%;
  border: #d86ecd 1px solid;
  }
</style>
<meta charset="utf-8">
<link rel="stylesheet" type="text/css" href="styles.css">
<script type="text/javascript" src="js/jquery-3.1.1 (1).js"></script>
<script type="text/javascript" src="js/js.js"></script>
<script type="text/javascript" src="js/jquery-3.1.1.min (1).js"></script>
<script type="text/javascript" src="js/jquery-3.1.1.min (1).map"></script>
<script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.1/jquery.min.js"></script>
</head>
<body>
  <div id='div41' style="position:absolute;top:0px;" >
<table>
<tr>
  <td class="td1">№</td>
  <td class="td1">Вид</td>
  <td class="td1">Товар</td>
  <td class="td1">Размер</td>
  <td class="td1">Цвет</td>
  <td class="td1">Материал</td>
  <td class="td1">Ед.изм.</td>
  <td class="td1">Цена</td>
  <td class="td1">Кол-во</td>

```



