**Министерство образования Иркутской области**

Государственное бюджетное профессиональное

образовательное учреждение Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

ПП.09.02.03.21.173.03

ОТЧЕТ ПО ПРЕДДИПЛОМНОЙ ПРАКТИКЕ

|  |  |  |
| --- | --- | --- |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Н.В. Пидгирнер) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Д.С. Зан) |

Иркутск 2021

**Содержание**

[ВВЕДЕНИЕ 4](#_Toc72399422)

[1. Содержание темы дипломного проекта 5](#_Toc72399423)

[2. Описание предметной области дипломного проекта 6](#_Toc72399424)

[3. Инструментальные средства разработки и проектирования 7](#_Toc72399425)

[4. Техническое задание дипломного проекта 9](#_Toc72399426)

[5. Проектирование программного продукта 10](#_Toc72399427)

[5.1. Функциональное проектирование 10](#_Toc72399428)

[5.2. Нормальная форма 13](#_Toc72399429)

[5.3. Проектирование базы данных 14](#_Toc72399430)

[6. Разработка программного продукта 15](#_Toc72399431)

[6.1. Проектирование интерфейса 15](#_Toc72399432)

[6.2. Разработка информационной системы 19](#_Toc72399433)

[6.3. Реализация базы данных в конкретной СУБД 22](#_Toc72399434)

[ЗАКЛЮЧЕНИЕ 27](#_Toc72399435)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 28](#_Toc72399436)

[Приложение А Техническое задание 30](#_Toc72399437)

## ВВЕДЕНИЕ

Преддипломная практика является закреплением знаний, овладение практическими навыками, умениями и технологиями практической деятельности, а также формирование профессиональной компетентности.

В процессе прохождения преддипломной практики необходимо выполнить следующие задание:

* Составить техническое задание на разработку ПО (в соответствии с ГОСТ);
* Провести предпроектный анализ (описать предметную область, определить и описать архитектуру программного обеспечения, составить список источников (не менее 15 источников));
* Выполнить анализ инструментов используемых в разработке программного обеспечения (не просто перечисление, а сравнение и выводы);
* Провести проектирование ПО (проектирование программного продукта, проектирование БД, пользовательского интерфейса (ER-модель, описание, нормальная форма, какая и почему, связи), функциональное проектирование (описать этапы создания работы или деятельность самой работы с помощью IDEF 0, IDEF 3, DFD, USES CASE и т.д.));
* Кратко продемонстрировать реализацию (разработку ДП).

# Содержание темы дипломного проекта

Темой дипломного проекта является разработка приложения «Интернет-магазин книг». Она обоснована развитием инфокоммуникационных технологий, с помощью которых можно организовать товарооборот через сеть Интернет. Такая возможность может значительно повысить продаваемость товара и организовать хранение необходимых данных товара в одном месте. Отсутствие такой возможности приводит к проблеме утери данных и большим временным затратам на выборку данных.

Углубимся в структуру работы интернет-магазина. Интернет-магазин – весьма распространенная и востребованная реализация товарооборота. Множество магазинов и не только книг используют данную реализацию для организации продаж.

Цель дипломной разработать приложение «Интернет-магазин книг». Цель разработки – упрощение процесса оказания услуг, накопление клиентской базы, осуществление хранения данных о товарах.

# Описание предметной области дипломного проекта

Областью применения программного продукта является интернет-магазин книг.

Книга — один из видов печатной продукции: непериодическое издание, состоящее из сброшюрованных или отдельных бумажных листов (страниц) или тетрадей, на которых нанесена типографским или рукописным способом текстовая и графическая информация, имеющее, как правило, твёрдый переплёт Основное назначение книг является передача различной информации в печатном виде.

Магазин — предприятие розничной торговли, размещённое в стационарном здании, оборудованном для продаж товаров и оказания услуг.

Интернет-магазин — сайт, торгующий товарами посредством сети Интернет.

Процесс оформления заказа книги прост и понятен. Для этого необходимо выбрать книгу, выбрать количество и оформить заказ.

Для создания такого приложения нужна общая база данных, включающая всю необходимую информацию. Приложение является очень актуальной на сегодняшний день, т.к. оног автоматизирует работу с базой данных и оформлением заказов.

# Инструментальные средства разработки и проектирования

Для разработки информационной системы было решено использовать PHPStorm, PHPMyAdmin, MySQL, Tailwindcss, Laravel, Xampp, Apache, Draw.io, Microsoft Visio.

Основные преимущества PHPStorm:

1. автодополнение кода;
2. предотвращение ошибок;
3. выполнение запросов в среде разработки;
4. глубокий анализ кода JavaScript, его популярных библиотек и фреймворков, CSS, HTML, PHP и его популярные библиотеки и фреймворки.

Основные преимущества PHPMyAdmin:

1. простой интерфейс;
2. выполнение всех функций манипуляций с таблицами;
3. настройка прав доступа, конфигураций серверов.

Основные преимущества MySQL:

1. простота;
2. эффективность;
3. быстрота;
4. безопасность;
5. гибкость.

К преимуществам Tailwindcss можно отнести:

1. описывает свойства css, а не компоненты;
2. настройка через js без написания своего css;
3. удаление неиспользуемых классов.

Основные преимущества Laravel:

1. высокая безопасность;
2. аутентификация;
3. шаблонизатор;
4. миграции баз данных;
5. MVC-архитектура.

Преимущества XAMPP:

1. кроссплатформенность;
2. отсутствие лишних программных компонентов;
3. легкая настройка.

Причины выбора apache:

1. быстрая настройка;
2. надёжность;
3. безопасность;
4. гибкость.

Таким образом, для реализации цели дипломного проекта было решено использовать следующие средства разработки:

* на этапе проектирования, для создания ER-модели использовалось CASE средства – draw io;
* для создания структурных схем использовалось CASE средства – Microsoft Visio;
* для создания контекстной и диаграмм декомпозиции использовался графический редактор – Microsoft Visio;
* на этапе разработки информационной системы, использовались WEB-технологии – HTML, PHP, JavaScript, CSS, Tailwindcss, Laravel;
* для работы с базой данных использовался метод языка программирования PHP - PDO.

Все программные продукты имеют огромный ряд достоинств, одно из которых открытая и доступная документация.

# [Техническое задание](#_Toc532432111) дипломного проекта

В начале разработки создавалось техническое задание, в котором указывались основные требования.

Для создания технического задания использовался стандарт ГОСТ 34.602-89.

Согласно ГОСТ 34.602-89 техническое задание должно включать следующие разделы:

1. общие сведения;
2. назначение и цели создания системы;
3. требования к системе в целом;
   1. требования к структуре и функционированию системы;
   2. требования к надежности;
   3. требования к безопасности;
   4. требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы;
4. требования к документированию;
5. состав и содержание работ по созданию системы.

Техническое задание на разработку приложения представлено в Приложение А.

# Проектирование программного продукта

# Функциональное проектирование

Функциональное проектирование – это подход к проектированию, который нацелен, прежде всего, на создание эффективно работающего объекта. Функциональное проектирование позволяет раскрыть все функции системы.

Выполнение требуемой функции – главная цель и основа разработки объекта. Во внимание принимаются, прежде всего, функциональные показатели качества и показатели надёжности.

Контекстная диаграмма – это модель, представляющая систему как набор иерархических действий, в которой каждое действие преобразует некоторый объект или набор объектов.

На контекстной диаграмме на рисунке 1 показаны входные данные, управление механизм, выходные данные и функция.

В центре диаграммы показана приложение «Интернет-магазин книг». Левая стрелка-это входные данные, т.е. «Данные с веб-страниц» и «Данные из БД». Верхняя стрелка — это управление «Алгоритм обработчика событий». Нижняя стрелка – это механизм «Пользователь». А правая - выходные данные «Обработанные данные».

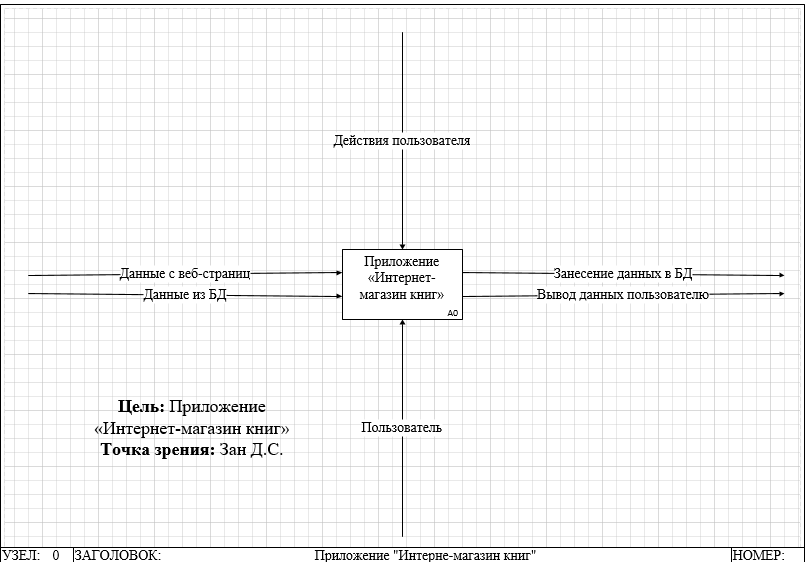


Рисунок 1 – Контекстная диаграмма модуля

Для демонстрации деталей в процессе необходимо спроектировать диаграмму декомпозиции.

Диаграмма декомпозиции – это разбиение функции, т.е. приложения «Интернет-магазин книг».

На рисунке 2 показана диаграмма декомпозиции, которая расписывает работу приложения из контекстной диаграммы. На этой диаграмме показаны 11 функций — это «Аутентификация», «Авторизация», «Обзор», «Магазин», «Корзина», «Заказы», «Личный кабинет», «Изменение БД», «Манипулирование данными ролей», «Вывод данных», «Манипулирование данными пользователей». А также на этой диаграмме присутствуют входные данные, выходные данные, управления и механизмы.

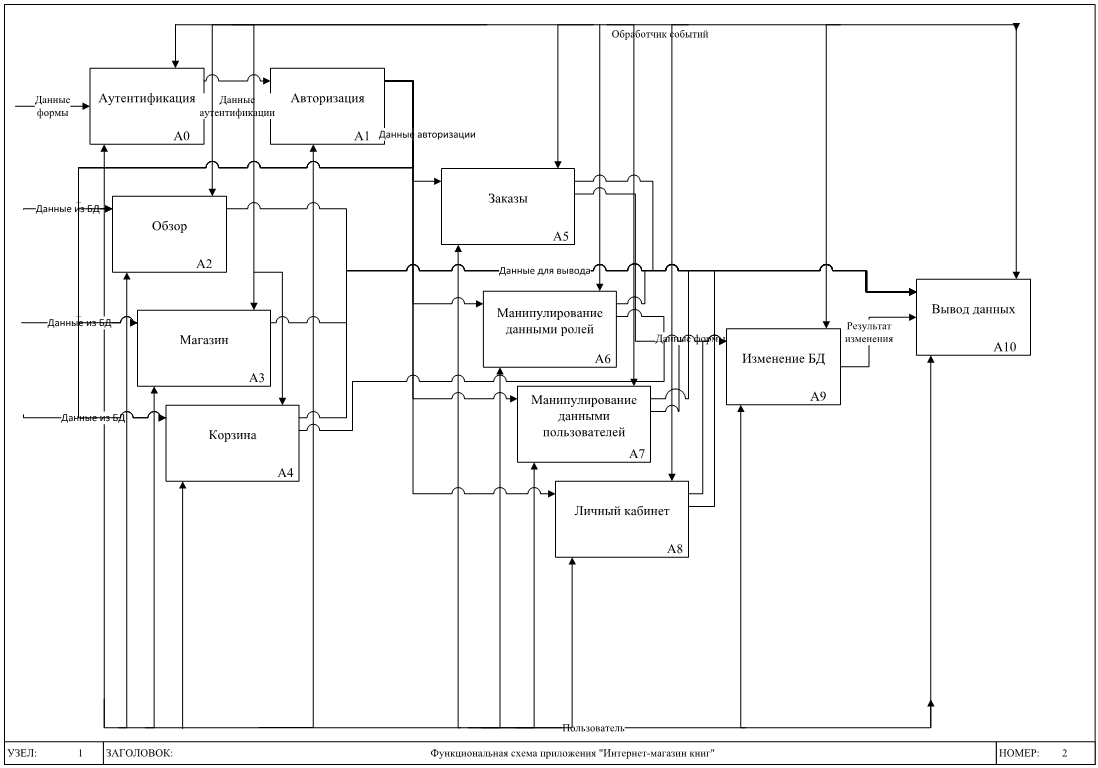


Рисунок 2 – Диаграмма декомпозиций

На рисунке 3 показана диаграмма IDEF3, которая расписывает работу приложения из рисунка 2.

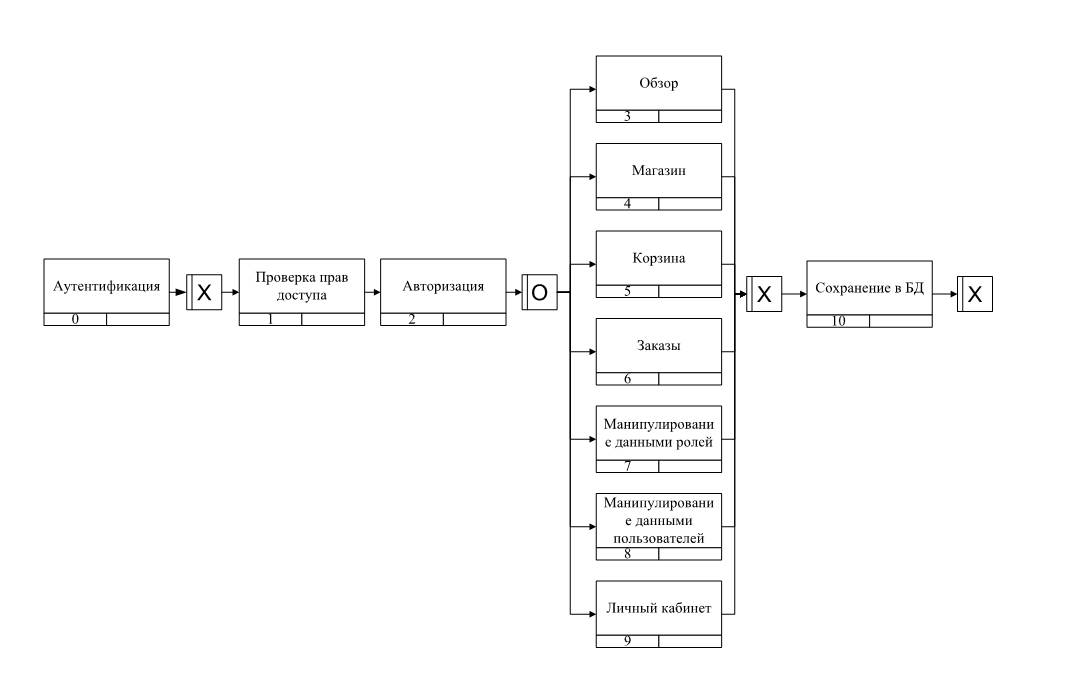


Рисунок 3 –IDEF3 диаграмма

На рисунке 4 показана диаграмма потоков данных, которая описывает передачу данных между разными функция и объектами приложения.

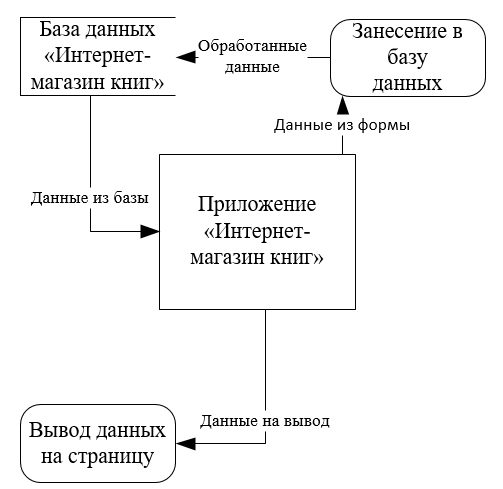


Рисунок 4 – Диаграмма потоков данных

Подробное функциональное проектирование позволяет глубже исследовать проектную область и цель проекта, что может позволить сэкономить время на разработке приложения, а также позволит избежать перенаписание кода.

# Нормальная форма

Для построения базы данных нужно было выявить нормальную форму.

Отношение находится в 3НФ так как, когда находится во 2НФ и каждый не ключевой атрибут нетранзитивно зависит от первичного ключа.

Проще говоря, второе правило требует выносить все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы.

# Проектирование базы данных

Проектирование приложения происходит при помощи CASE средств, которые позволяют быстро создавать схемы и реляционные модели программ.

Модель сущность-связь или схема баз данных – модель данных, позволяющая описывать концептуальные схемы предметной области.

На рисунке 5 изображена схема базы данных приложения «Интернет-магазин книг».

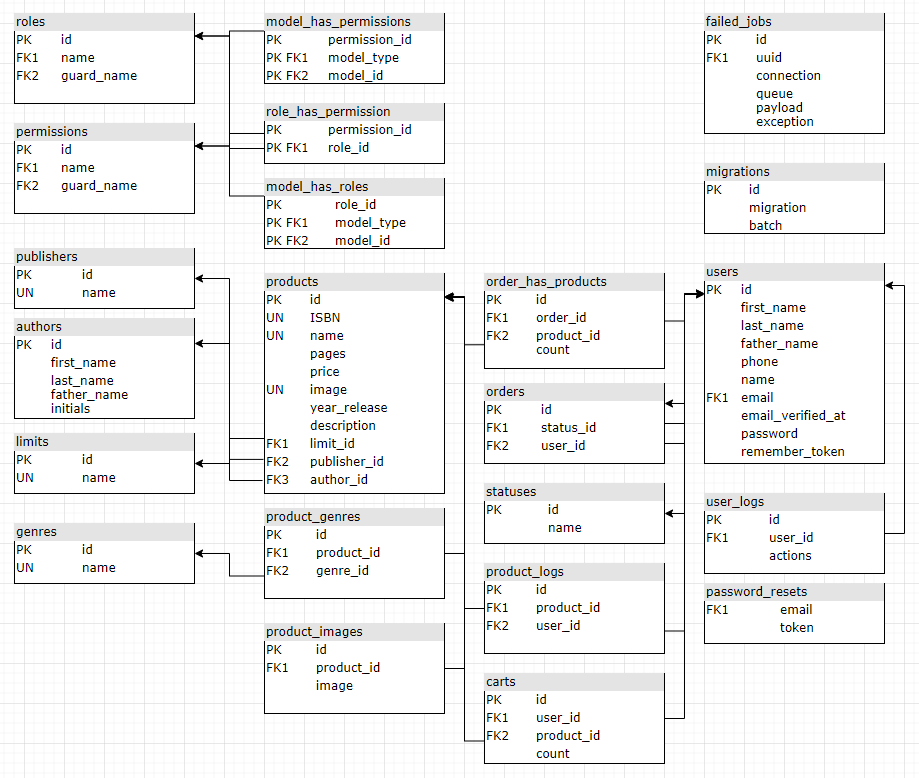


Рисунок 5 – Схема базы данных

База данных представляет собой 22 связанных таблиц. Для создания связей между таблицами используются внешние ключи.

Разработка ER-модели выполнена с помощью программы case-средств draw io.

# Разработка программного продукта

# Проектирование интерфейса

Интерфейс приложения – это одна из важнейших частей проектирования программного обеспечения.

Интерфейсом приложения «Интернет-магазин книг» является меню навигации, содержащее разделы приложения, рабочей области.

На рисунке 6 показан прототип раздела «Вход», который авторизоваться пользователю приложения.

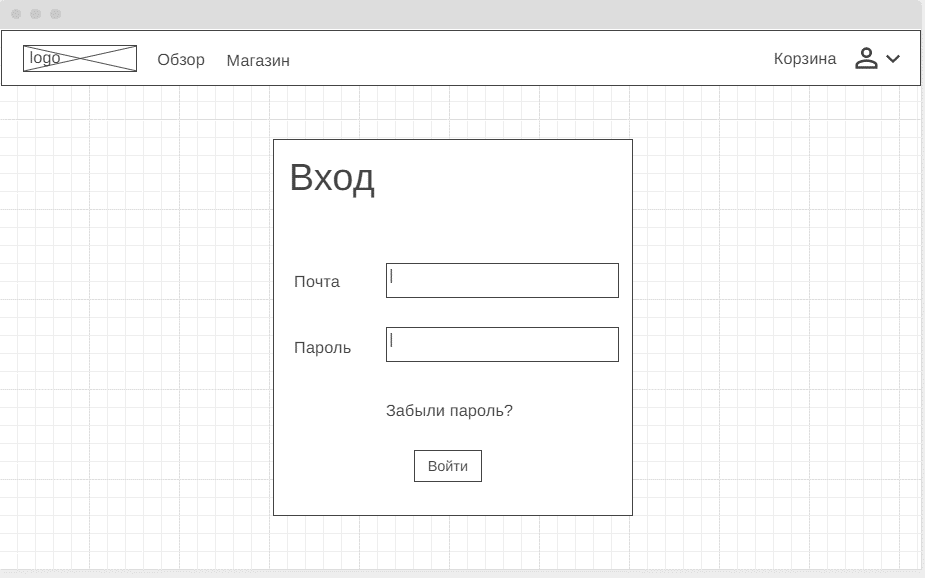


Рисунок 6 – Прототип раздела «Вход»

На рисунке 7 демонстрируется прототип основной страницы «Обзор», на которой пользователь может увидеть рекомендуемые приложением товары. На эту страницу может войти и не авторизованный пользователь приложения.

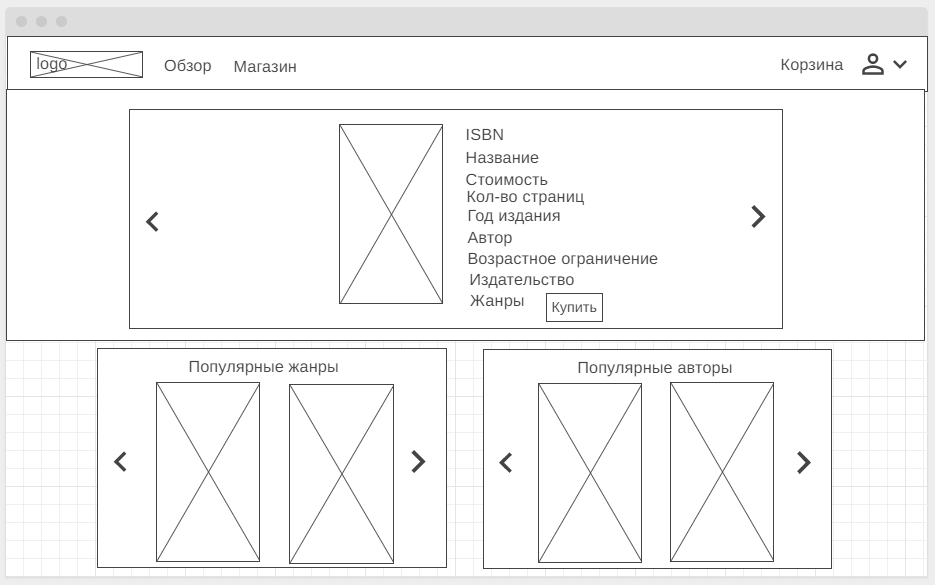


Рисунок 7 – Прототип страницы «Обзор»

На рисунке 8 демонстрируется прототип страницы «Магазин». На рабочем пространстве размещен список книг и список фильтров, которые можно применить к списку книг.

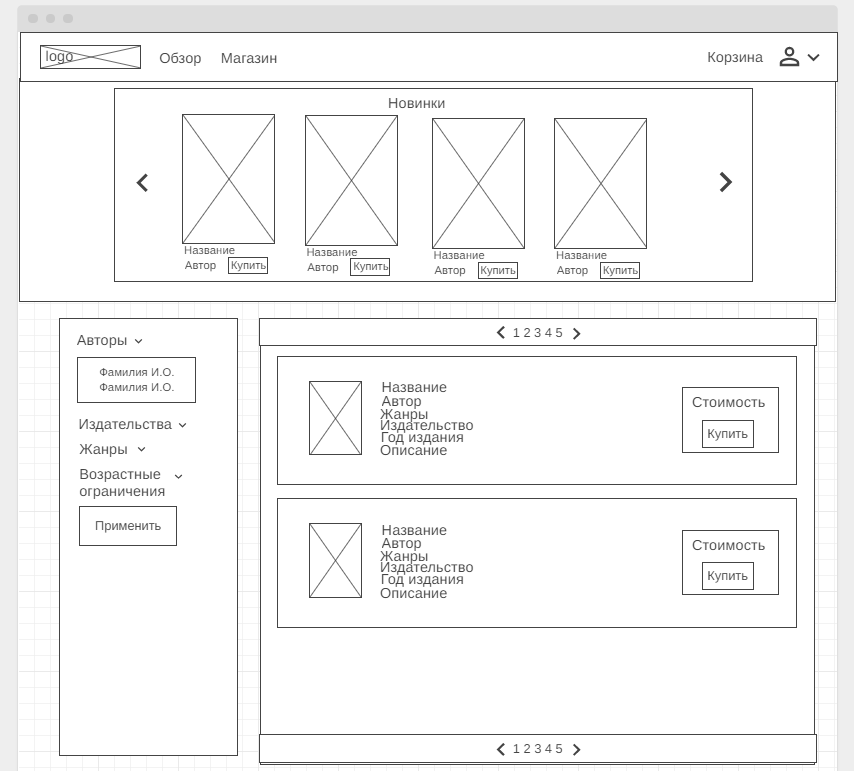


Рисунок 8 – Прототип страницы «Магазин»

На рисунке 9 демонстрируется прототип страницы «Корзина». На рабочем пространстве размещен список книг, выбранных пользователем для оформления заказа.

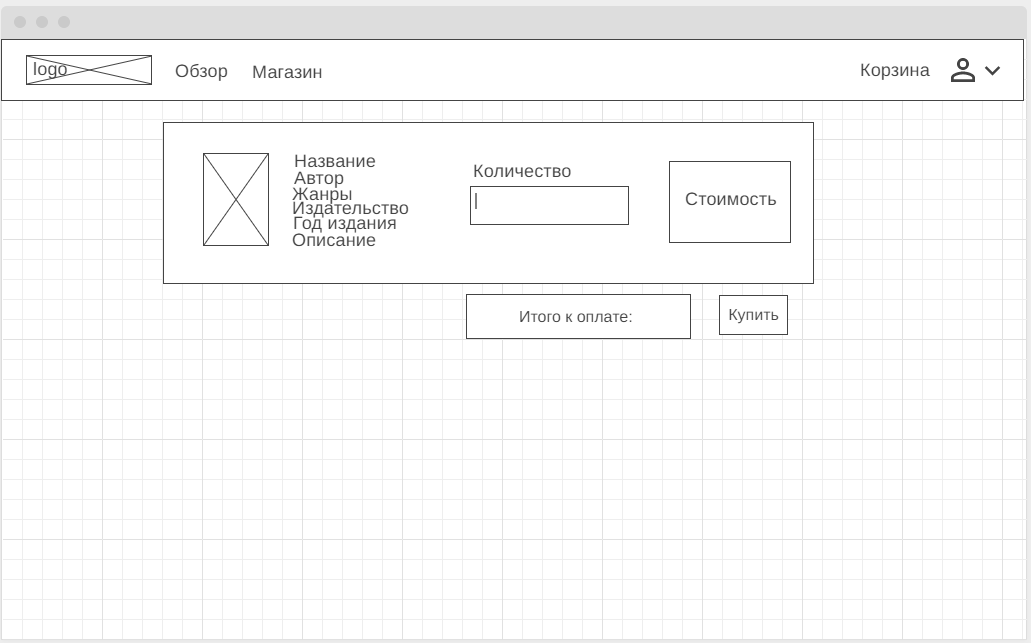


Рисунок 9 – Прототип страницы «Корзина»

На рисунке 10 демонстрируется прототип страницы «Регистрация», которая позволяет зарегистрировать нового пользователя приложения.

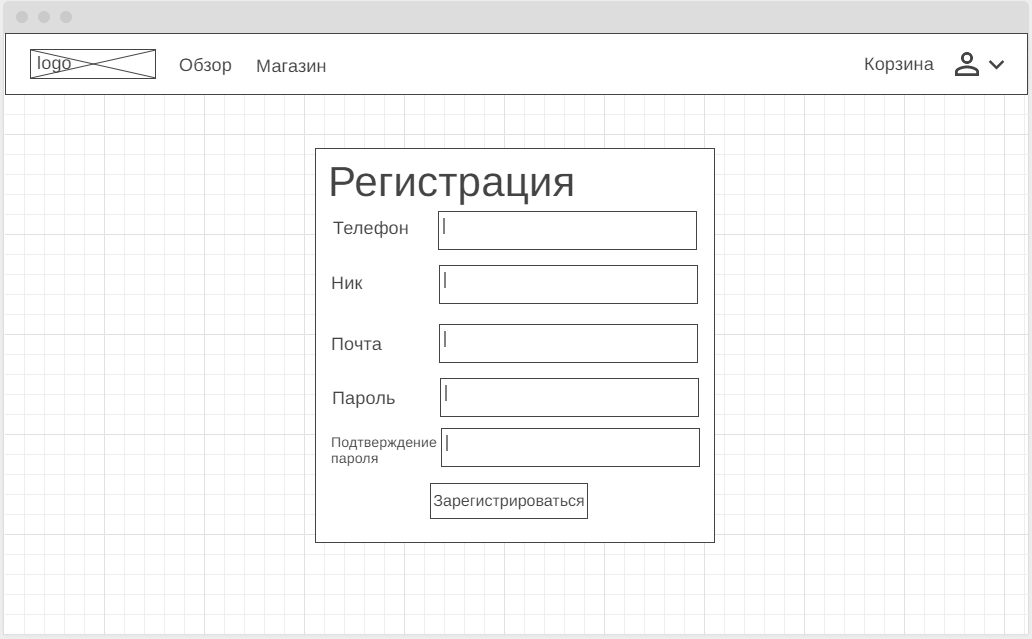


Рисунок 10 – Прототип страницы «Регистрация»

На рисунке 11 демонстрируется прототип страницы «Продукты». Данная страница доступна только пользователям с соответствующими правами доступа. На рабочем пространстве размещены форма для добавления нового товара и список форм существующих книг для редактирования или удаления данных.

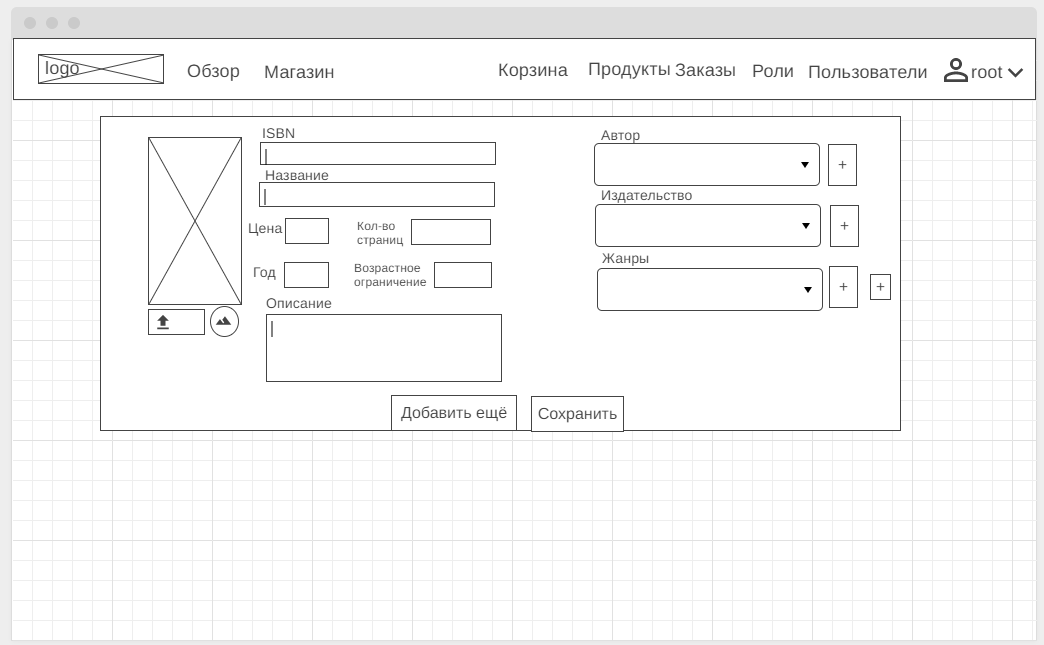


Рисунок 11 – Прототип страницы «Продукты»

На рисунке 12 демонстрируется прототип страницы «Заказы». Данная страница доступна только пользователям с соответствующими правами доступа. На рабочем пространстве размещены список заказов, которые по статусу не выданы клиенту. Страница позволяет изменить статус заказа.

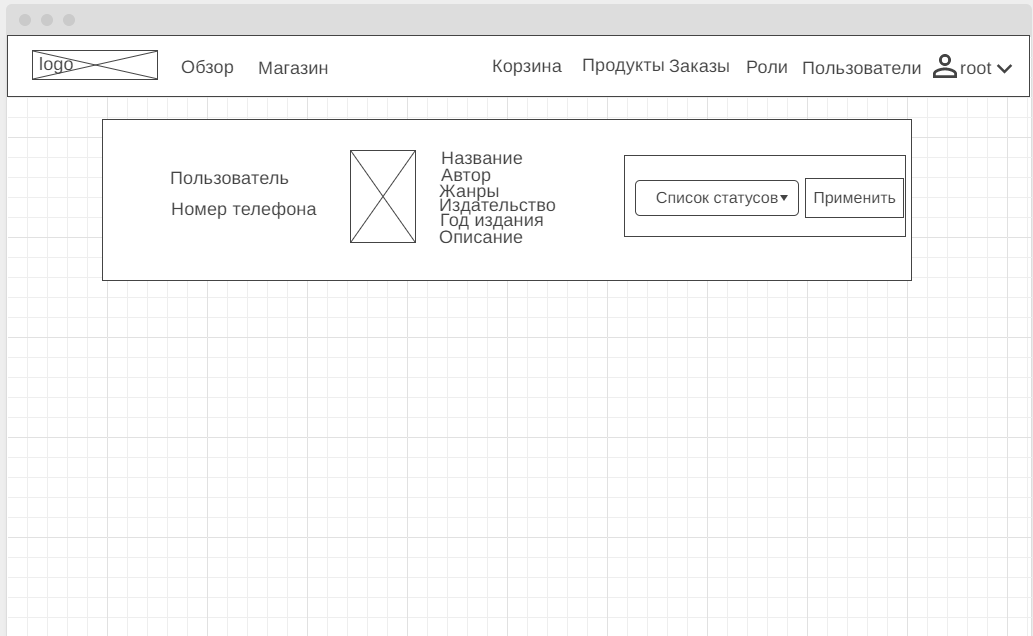


Рисунок 12 – Прототип страницы «Заказы»

Проектирование интерфейса пользователя позволяет определиться с расположение объектов и данных на рабочем пространстве. Также позволяет определиться с количеством страниц и избежать конфликтов с заказчиков.

# Разработка информационной системы

Суть реализации программного обеспечения заключается в кодировании – процесс написания программного кода. В связи с тем, что разрабатываемое приложение основано на веб-технологиях, в ходе разработки было принято решение использовать:

* фреймворк tailwind css - для клиентской части;
* фреймворк Vanilla JS - для скриптового языка JavaScript;
* фреймворк Laravel - для серверной составляющей модуля.

Благодаря фреймворку tailwind css приложение «Интернет-магазин книг» имеет простой, но стильный внешний вид, который продемонстрирован на рисунке 13.

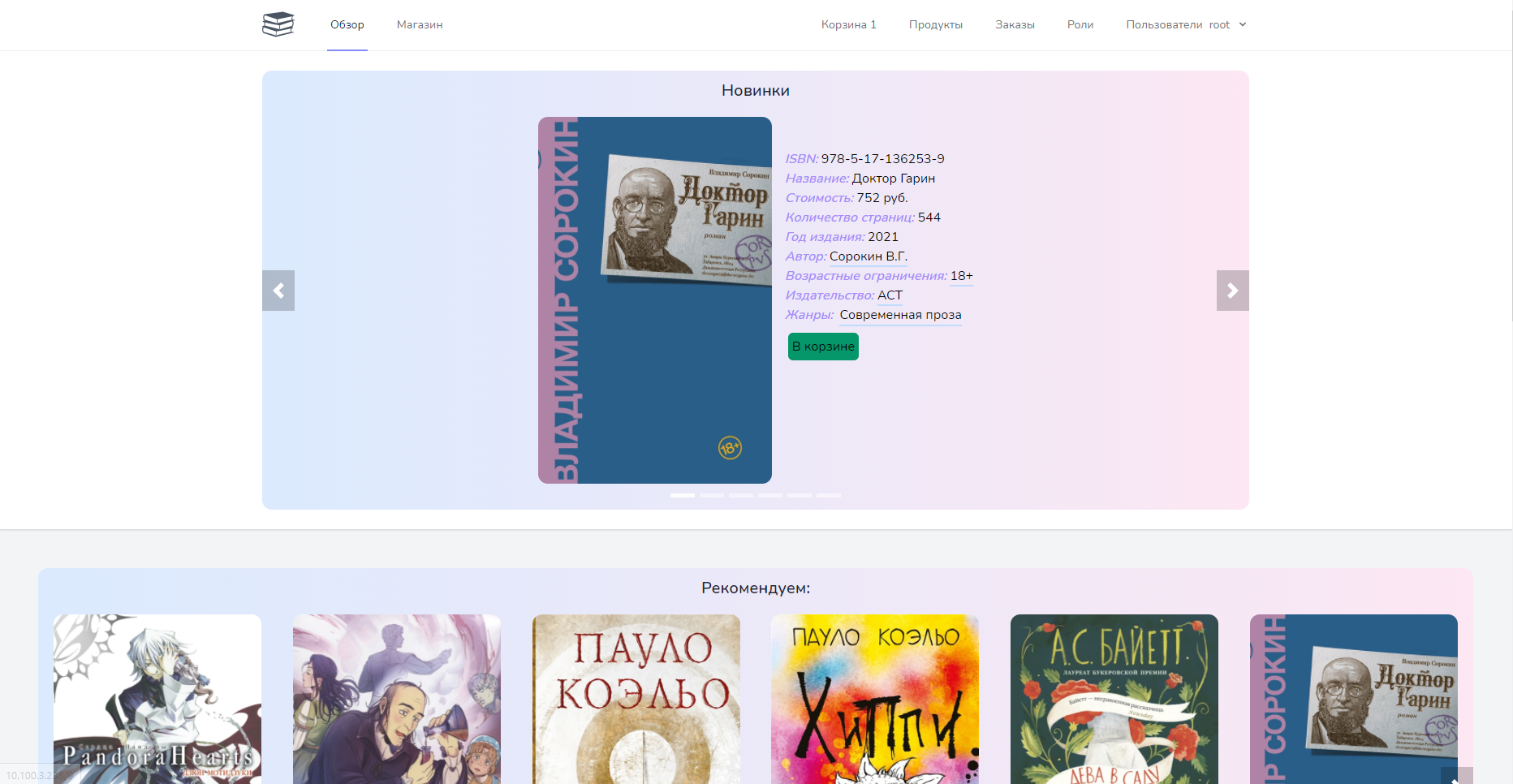


Рисунок 13 – Вкладка «Обзор»

Фреймворк Vanilla JS позволил организовать динамическое получение данных, что позволило реализовать поиск в разделах приложения.

Фреймворк Laravel, в данной разработке, позволяет осуществить защиту передачи данных и облегчить работу с БД. Фреймворк использует архитектурную модель MVC (Model View Controller) представленную на рисунке 14. Принцип работы, которой необходимо создавать файл контроллера, модель и представления.

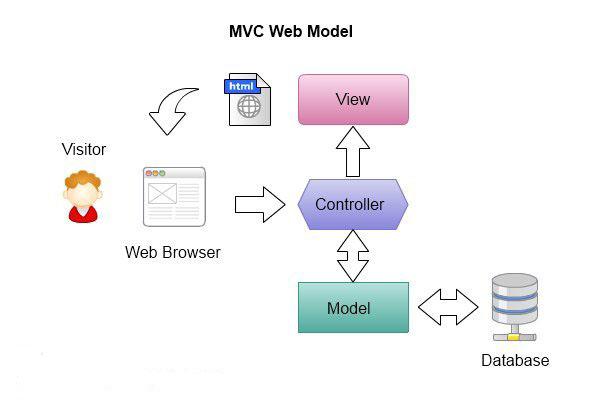


Рисунок 14 – Архитектурная модель MVC

Контроллеры отвечают за обработку запросов пользователя и вызов соответствующих представлений. Контроллер представляет собой класс, который содержит методы обработки функций. Все контроллеры проекта находятся в папке /app/http/Controllers. Самый крупный контроллер это ProductController. Он отвечает за обработку данных раздела «Продукты». На рисунке 15 представлен код функции create из контроллера ProductController.

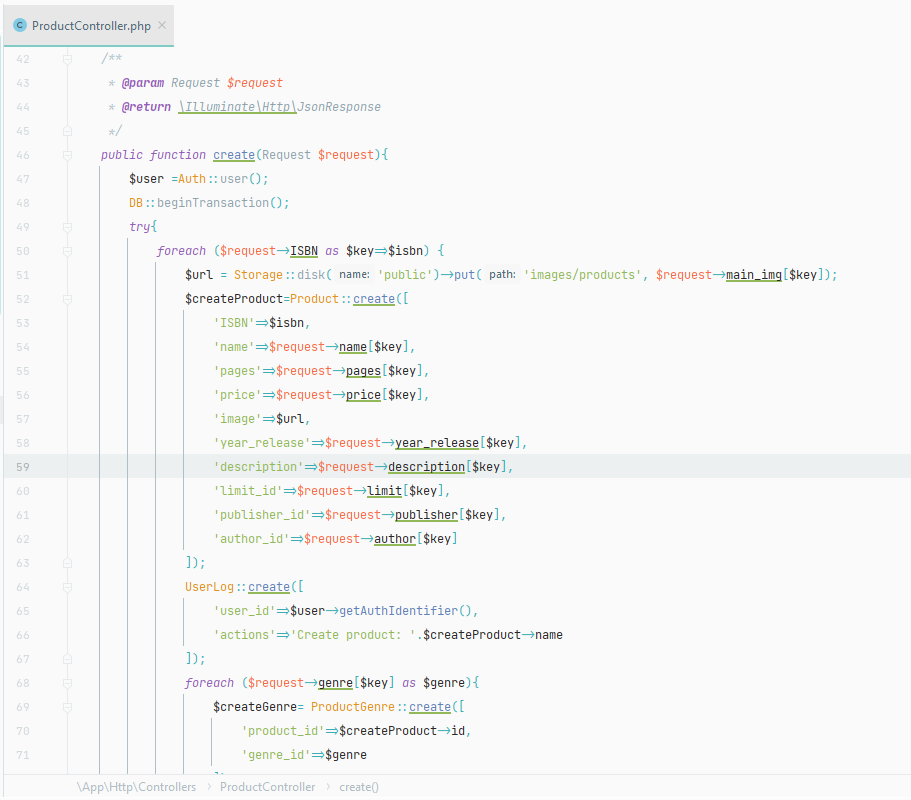


Рисунок 15 – Фрагмент контроллера ProductController

Представления отвечают за показ результатов запроса пользователю. Каждое представление – это отдельная страница, содержащая html и javascript код, а также php вставки. Все представления находятся в папке /resource/views. Представление страницы «Магазин» является одной из крупнейших представлений приложения. В ней данные из контроллера представляются в виде списка. На рисунке 16 изображен фрагмент кода этого представления.

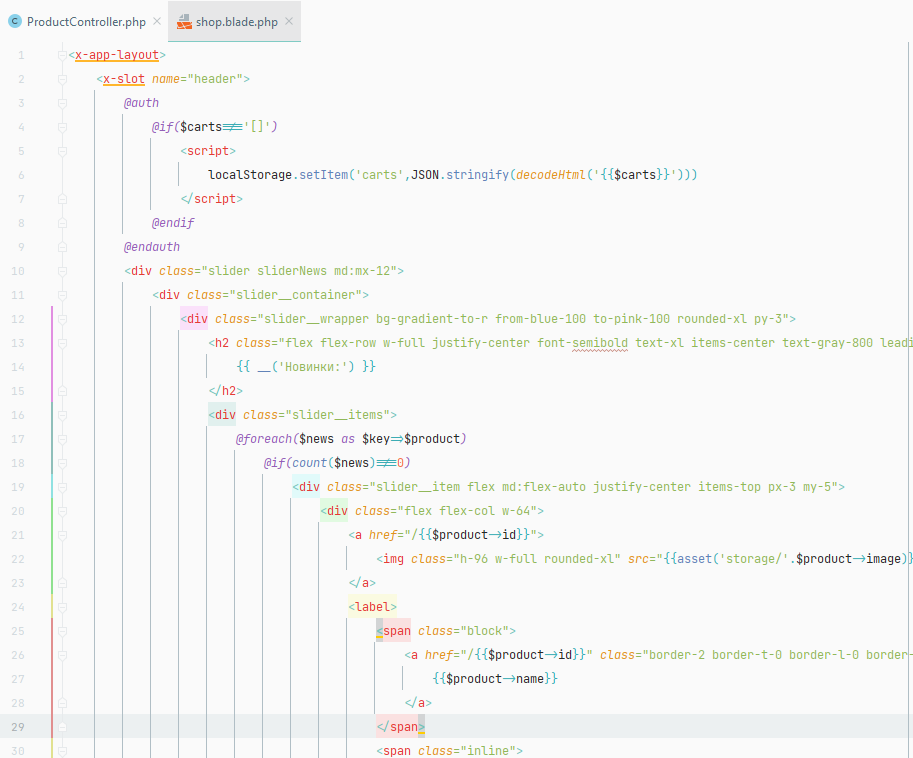


Рисунок 16 – Фрагмент представления страницы «Магазин»

Приложение содержит несколько крупных функций. Одна из самых крупных функций — это функция применения фильтров на странице «Магазин» (фрагмент кода на рисунке 17).

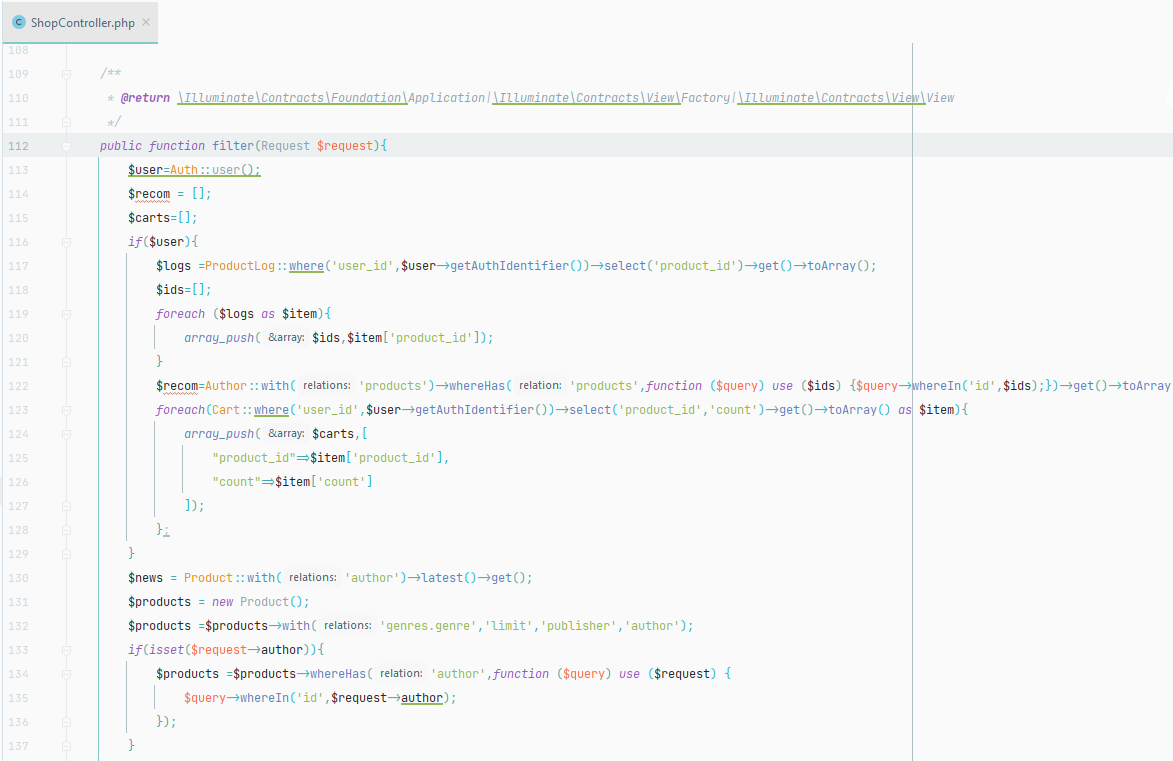


Рисунок 17 – Фрагмент функции фильтров

Эта функция применяет выбранные фильтры пользователям к странице. Функция написана на php.

# Реализация базы данных в конкретной СУБД

Модель сущность-связь или схема баз данных – модель данных, позволяющая описывать концептуальные схемы предметной области.

Таблицы базы данных приложения «Интернет-магазин книг» показана на рисунке 18.

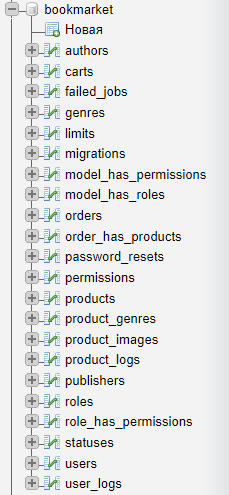


Рисунок 18 – База данных

База данных представляет собой 22 связанных таблиц. Для создания связей между таблицами используются внешние ключи.

Были выявлены следующие сущности и их атрибуты:

1. Авторы:

* имя автора;
* фамилия автора;
* отчество автора
* инициалы автора.

1. Возрастные ограничения:

* название возрастного ограничения.

1. Жанры:

* название жанра.

1. Издательства:

* название издательства.

1. Жанры книг:

* книга;
* жанр.

1. Изображения книги:

* книга;
* путь до изображения.

1. Книги:

* ISBN;
* название книги;
* количество страниц;
* стоимость книги;
* главное изображение книги;
* год издания книги;
* возрастное ограничение;
* издательство;
* автор.

1. Заказы:

* пользователь;
* статус.

1. Книги заказа:

* заказ;
* книга;
* количество.

1. Статусы:

* название статуса.

1. Действия с книгами:

* пользователь;
* книга.

1. Действия пользователей:

* пользователь;
* действие.

1. Пользователи:

* никнейм;
* имя пользователя;
* фамилия пользователя;
* отчество пользователя;
* телефон пользователя;
* почта;
* пароль;
* верификация почты.

1. Восстановление пароля:

* почта;
* дата изменения.

1. Миграции:

* миграция;
* количество.

1. Ошибки:

* соединение;
* очередь;
* дата ошибки;
* нагрузка;
* исключения.

1. Роли пользователя:

* название роли;
* дата создания роли.

1. Права доступа:

* название права;
* дата создания права.

1. Модели ролей:

* роль;
* тип модели.

1. Роли прав доступа:

* право доступа;
* роль.

1. Модели прав доступа:

* право доступа;
* тип модели.

1. Корзина:

* пользователь;
* книга.

Нормальная форма — свойство отношения в реляционной модели данных, характеризующее его с точки зрения избыточности, потенциально приводящей к логически ошибочным результатам выборки или изменения данных. Нормальная форма определяется как совокупность требований, которым должно удовлетворять отношение.

Третья нормальная форма, которую придерживается база данных, содержит в себе следующее требование:

– Отношение находится в 3НФ, когда находится во 2НФ и каждый не ключевой атрибут нетранзитивно зависит от первичного ключа. Проще говоря, второе правило требует выносить все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы.

Все таблицы содержат минимальный набор независимых данных. Большинство таблиц не содержат больше 5 столбцов.

Исходя из всего выше сказанного, уровень нормализации базы данных равен 3.

# ЗАКЛЮЧЕНИЕ

В результате преддипломной практики были выполнены следующие задания:

1. Составлено техническое задание на разработку;
2. Проведён предпроектный анализ;
3. Выполнен анализ инструментов, используемых в разработке приложения;
4. Проведено проектирование ПО (проектирование программного продукта, проектирование БД, пользовательского интерфейса (ER-модель, описание, нормальная форма, какая и почему, связи)), функциональное проектирование;
5. Кратко продемонстрирована реализация (разработки ДП).

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Bootstrap [Электронный ресурс] – Режим доступа: http://getbootstrap.ru – Bootstrap Самый популярный в мире front-end фреймворк. (Дата обращения: 01.05.21).
2. CyberForum [Электронный ресурс] – Режим доступа: http://www.cyberforum.ru – (Дата обращения: 03.05.21).
3. Draw.io это [Электронный ресурс] – Режим доступа: https://bizzapps.ru/p/draw-io/ – Документация (Дата обращения: 21.04.2021).
4. Hostinger Руководства [Электронный ресурс] – Режим доступа: https://www.hostinger.ru/rukovodstva/web-server-apache/ – Что такое Apache? (Дата обращения: 16.05.21).
5. Laravel по-русски [Электронный ресурс] – Режим доступа: <https://laravel.ru/docs/v5/queries> – Конструктор запросов (Дата обращения: 02.05.21).
6. PHP [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/PHP – PHP (Дата обращения: 10.05.21).
7. PHP.NET [Электронный ресурс] – Режим доступа: https://www.php.net/ – Документация по PHP (Дата обращения: 18.04.2021).
8. PHP-MyAdmin.ru [Электронный ресурс] – Режим доступа: https://php-myadmin.ru/doc/ability.html – Руководство. Возможности PhpMyAdmin (Дата обращения: 02.05.21).
9. PHPStorm [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/PHPStorm – PHPStorm (Дата обращения: 10.05.21).
10. SQL. Полное руководство. Грофф Дж. Р., Вайнберг П.Н., Оппелъ Э. Дж. 2015 – 959 с.
11. tailwindcss [Электронный ресурс] – Режим доступа: https://tailwindcss.com/docs – Getting started with Tailwind CSS (Дата обращения: 11.04.2021).
12. Базы данных [Электронный ресурс] – Режим доступа: https://life-prog.ru – Документация (Дата обращения: 21.04.2021).
13. Всё о схематизации с Microsoft Visio [Электронный ресурс] – Режим доступа: https://www.lucidchart.com/pages/ru/visio-что-это – Что такое Visio? (Дата обращения: 18.04.2021).
14. Компьютерра [Электронный ресурс] – Режим доступа: https://www.computerra.ru/230618/7-besplatnyh-instrumentov-wireframe-dlya-proektirovaniya-interfejsov-i-protsessov/– 7 бесплатных инструментов wireframe (Дата обращения: 11.05.2021).
15. Передача значений переменных из JavaScript в PHP и наоборот [Электронный ресурс] – Режим доступа: https://webformyself.com/peredacha-znachenij-peremennyx-iz-javascript-v-php-i-naoborot/ – Документация (Дата обращения: 27.04.2021).
16. Хабр [Электронный ресурс] – Режим доступа: https://habr.com/ru/company/ruvds/blog/539098/ – 9 самых популярных php-фреймворков (Дата обращения: 28.04.2021).
17. Электронный учебник: JavaScript [Электронный ресурс] – Режим доступа: https://learn.javascript.ru/manuals-specifications – Справочники и спецификации (Дата обращения: 13.04.2021).

# Приложение А Техническое задание

**Министерство образования Иркутской области**

Государственное бюджетное профессиональное образовательное учреждение

Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

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

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

ПРИЛОЖЕНИЕ «ИНТЕРНЕТ-МАГАЗИН КНИГ»

|  |  |  |
| --- | --- | --- |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Пидгирнер Н.В.) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Зан Д.С.) |

Иркутск 2021

**1 Общие сведения**

Наименование работы: Приложение «Интернет-магазин книг»

Заказчик: ГБПОУИО «Иркутский Авиационный техникум»

Исполнитель: студент Иркутского Авиационного техникума, группы ПКС-17-3, Зан Д.С.

**2 Назначение и цели создания системы**

Назначение разработки – упрощение процесса оказания услуг магазина книг, накопление клиентской базы. Приложение должно содержать информацию о клиентах, заказах, книгах.

Приложение должна решать следующие задачи:

* хранение информации о товарах, операциях над товарами;
* добавление, удаление и редактирование данных;
* распределение прав учётных записей;
* реализация функции подбора книг на основе совершенных покупок и просмотров.

**3 Требования к системе в целом**

## 3.1 Требования к структуре и функционированию системы

В разрабатываемом приложении должно быть реализовано:

1. Раздел «Заказы»:

* изменение статуса не готовых заказов;
* просмотр не готовых заказов.

1. Раздел «Корзина»:

* просмотр товаров пользователя, выбранных для покупки;
* возможность покупки;
* увеличение количества выбранного товара.

1. Раздел «Обзор»:

* просмотр товаров по разным категориям;
* просмотр одного товара;
* возможность добавления товара в корзину.

1. Раздел «Продукты»:

* просмотр товаров;
* добавление, редактирование, удаление данных товара.

1. Раздел «Магазин»:

* просмотр товаров;
* применение фильтров к списку товаров;
* по страничный вывод товаров;
* просмотр товаров одного автора, жанра, издательства, возрастного ограничения;
* возможность добавления товара в корзину.

1. Раздел «Роли»:

* создание новой роли;
* посмотреть права доступа роли;
* редактировать права доступа роли;
* удалить роль.

1. Раздел «Пользователи»:

* зарегистрировать нового пользователя информационной системы;
* редактировать пароль, почту, логин, роль пользователей;
* просмотреть данные пользователя;
* удалить пользователя.

1. Личный кабинет:

* изменение личных данных;
* изменение данных авторизации;
* просмотр истории заказов;
* просмотр логируемых данных пользователей.

1. Регистрация:

* возможность регистрация пользователя.

1. Вход:

* возможность авторизации пользователя в приложение.

**3.2 Требования к надежности**

Для обеспечения надежности необходимо проверять корректность получаемых данных. Входные данные поступают в виде значений из клавиатуры. Эти значения отображаются в полях ввода программы.

**3.3 Требования к безопасности**

Для обеспечения безопасности в приложении, необходимо реализовать разграничение прав доступа для клиентов и администрации сайта.

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

Минимальные системные требования сервера:

1. процессор: Intel Core i5 7400/ AMD Ryzen 5 1400;
2. оперативная память: 4 Гб;
3. Место на жестком диске: 4 Гб.

Рекомендуемые системные требования сервера:

1. процессор: AMD FX-8350/ INTEL Xeon E5-2603;
2. оперативная память: 8 Гб;
3. Место на жестком диске: 6 Гб.

Требования клиента:

* Браузер: Chrome, Safari.

Требования сервера:

* Поддержка –PHP 8.0, MYSQL 8.0, Apache 2.4.
* Операционная система: Windows, Unix-подобные.

**4 Требования к документированию**

Основным документом, регламентирующим разработку, является техническое задание.

Для облегчения взаимодействия с программным продуктом, необходимо наличие руководства пользователя и системного администратора.

**5 Состав и содержание работ по созданию системы**

В таблице 1 представлены плановые сроки начала и окончания работы по выполнению дипломного проекта.

Таблица 1 – Плановые сроки по выполнению дипломного проекта

|  |  |
| --- | --- |
| Этапы выполнения дипломного проекта | Сроки |
| Провести предпроектное исследование. | 23.04.21 |
| Составить техническое задание на разработку в соответствии с ГОСТ. | 27.04.21 |
| Провести проектирование программного продукта. | 04.05.21 |
| Реализовать программный продукт. | 18.05.21 |
| Выполнить тестирование и отладку программного обеспечения. | 20.05.21 |
| Представить внедрение и сопровождение программного продукта. | 25.05.21 |
| Разработать документы для программного продукта. | 29.05.21 |
| Расчитать стоимость разработки и внедрения программного продукта. | 30.05.21 |
| Составить пояснитиельную записку. | 01.06.21 |