**1 Выбор и обоснование средств проектирования и реализации**

Средства проектирования и реализации автоматизированной системы играют ключевую роль в обеспечении эффективности, надежности и удобства использования системы.

В качестве операционной системы для сервера была выбрана Windows Server. Она является стандартной операционной системой для серверов в корпоративной среде. Он предоставляет надежность, безопасность и широкие возможности для развертывания и управления веб-приложениями.

Для программирования серверной стороны были выбраны Node.js и Python. Node.js обеспечивает эффективное выполнение серверного кода на JavaScript, что упрощает разработку полноценных веб-приложений. Его асинхронная модель позволяет обрабатывать большое количество одновременных запросов, что важно для высоконагруженных систем. Python - это высокоуровневый и легко читаемый язык программирования, что упрощает разработку и поддержание кода. Он также обладает обширным сообществом разработчиков и множеством библиотек, что улучшает производительность и функциональность системы.

Для эффективного управления базой данных был использован PostgreSQL. Он представляет собой мощную открытую реляционную систему управления базами данных, которая отличается высокой степенью надежности, расширяемостью и соответствием стандартам SQL. Ее функциональные возможности и поддержка сложных типов данных делают PostgreSQL отличным выбором для приложений, требующих сложных запросов и обработки больших объемов данных. В контексте автоматизированной системы букинистического магазина, PostgreSQL обеспечивает эффективное хранение, быстрый доступ и управление информацией о книгах, клиентах и транзакциях. Важным преимуществом PostgreSQL является его открытый исходный код, что способствует гибкости в интеграции с различными технологиями и языками программирования, такими как Python и Node.js.

При разработке использовались такие web–технологии, как JavaScript, CSS, HTML. JavaScript обеспечивает динамическое взаимодействие на стороне клиента, а CSS и HTML используются для структурирования и стилизации веб-страниц. Их комбинация обеспечивает создание интерактивных и привлекательных пользовательских интерфейсов. Также была использована библиотека JacaScript – React. Она предназначена для разработки пользовательских интерфейсов. Ее компонентный подход упрощает создание и поддержку сложных интерфейсов, а также обеспечивает повторное использование кода.

Также была использована Figma, которая предоставляет возможности для дизайна интерфейсов и прототипирования. Это позволяет создавать макеты и прототипы, что улучшает визуализацию и понимание конечного результата.

**2 Проектирование архитектуры приложения**

При проектировании приложения была выбрана архитектура приложения вида MVC (рисунок 1), которая подразумевает разделение на три модуля: модель (model), представление (view), контроллер (controller). Модель отвечает за данные, методы работы с ними и структуру программы. Она реагирует на команды из контроллера и выдает информацию и/или изменяет свое состояние и передает данные в представление, которое отображает данные на уровне пользовательского интерфейса. Контроллер обеспечивает взаимодействие с системой. Он обрабатывает действия пользователя, проверяет полученную информацию и передает ее модели.

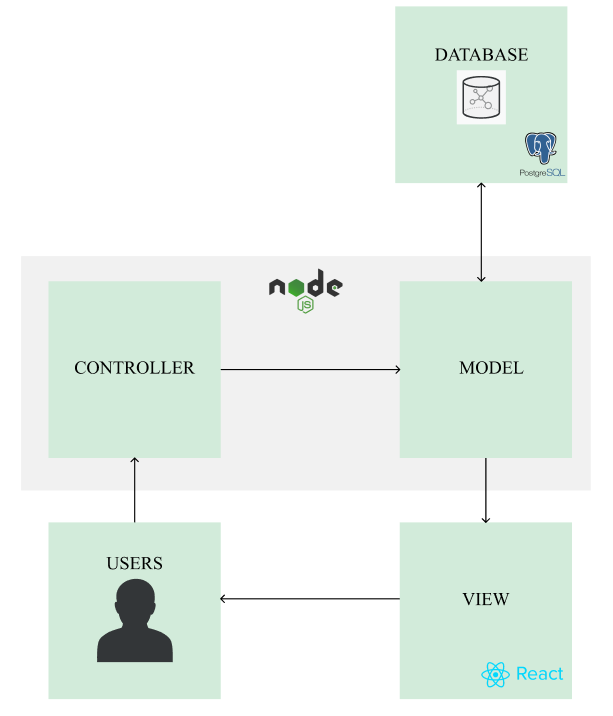


Рисунок – 1 Схема архитектуры приложения

**3 Проектирование хранилища данных**

В таблице 1 представлены основные сущности и их атрибуты.

Таблица 1 – Основные сущности

|  |  |
| --- | --- |
| Сущность | Атрифуты |
| Клиенты | id, ФИО, телефон, почта, пароль |
| Прием книг | id, id\_клиента, клиенты, комментарий, сохранность, помощь в перевозке, результат, дата запроса |
| Индивидуальный подбор книг | Id, интересы, id\_клиента, дата\_запроса |
| Список книг | id\_книги, название, id\_города, id\_издательства, год издания, количество страниц, id\_переплета, цена, OZON, Вконтакте, Инстаграм, id\_состояния |
| Прием книг | Id, id клиента, комментарий, сохранность, помощь в перевозке, дата запроса. |

На рисунке 2 показана логическая структура базы данных, которая показывает логическую организацию данных в виде множества типов записей и связей между ними.

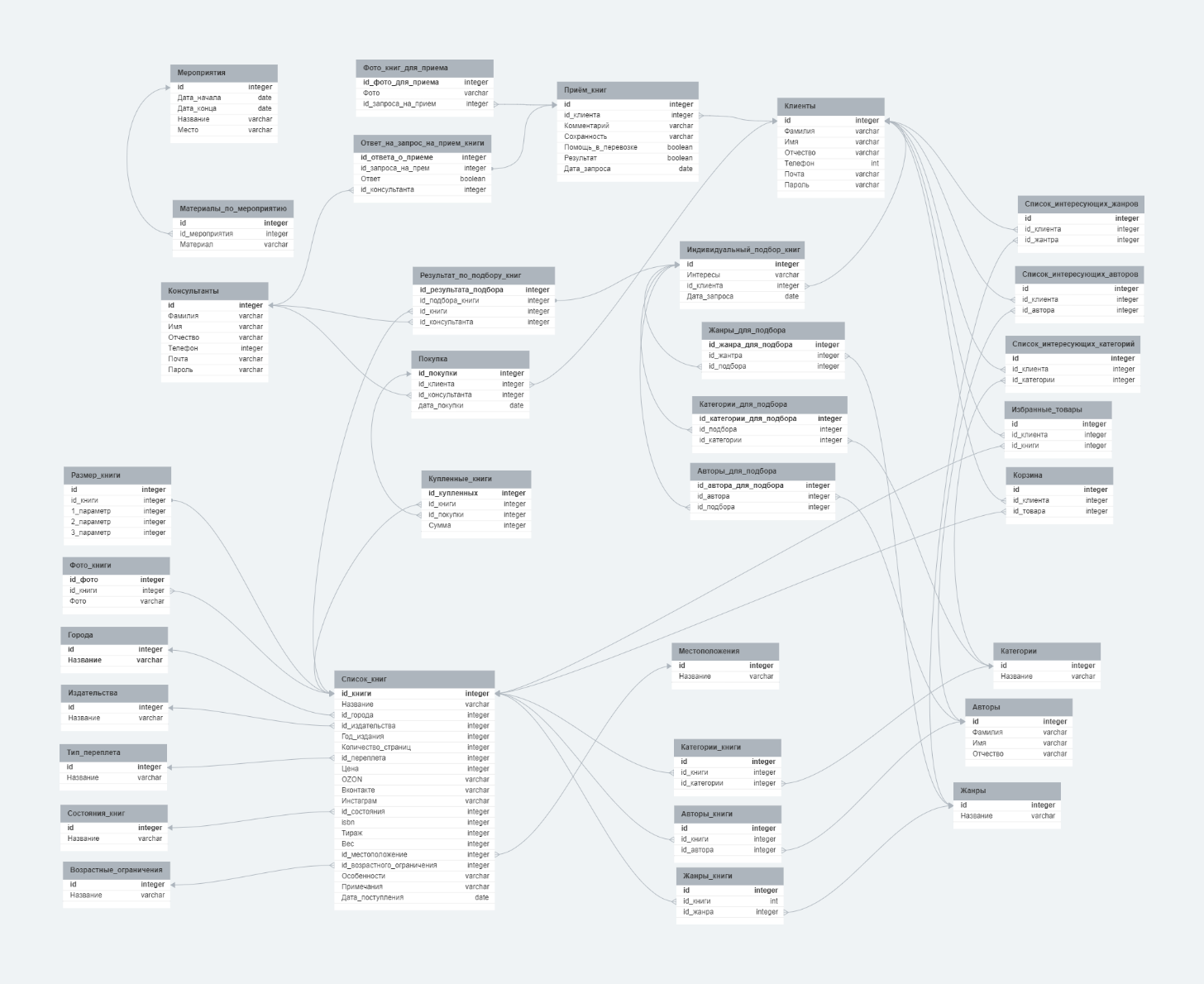


Рисунок – 2 Логическая модель данных

**4 Проектирование пользовательского интерфейса**

На рисунках 2 и 3 представлена схема верхнеуровневого определения экранов, сгруппированных по разделам.



Рисунок 3 – Схема верхнеуровнего определения экранов группируем по разделам

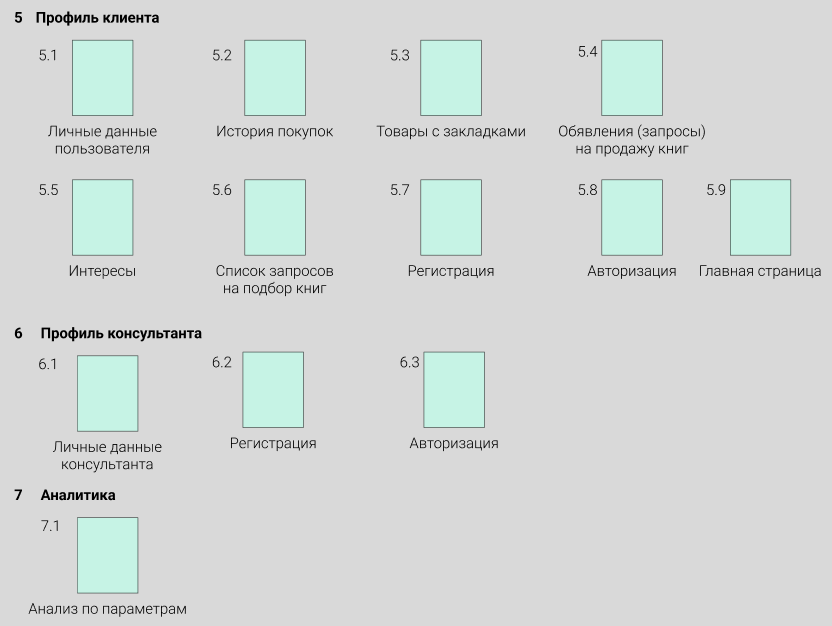
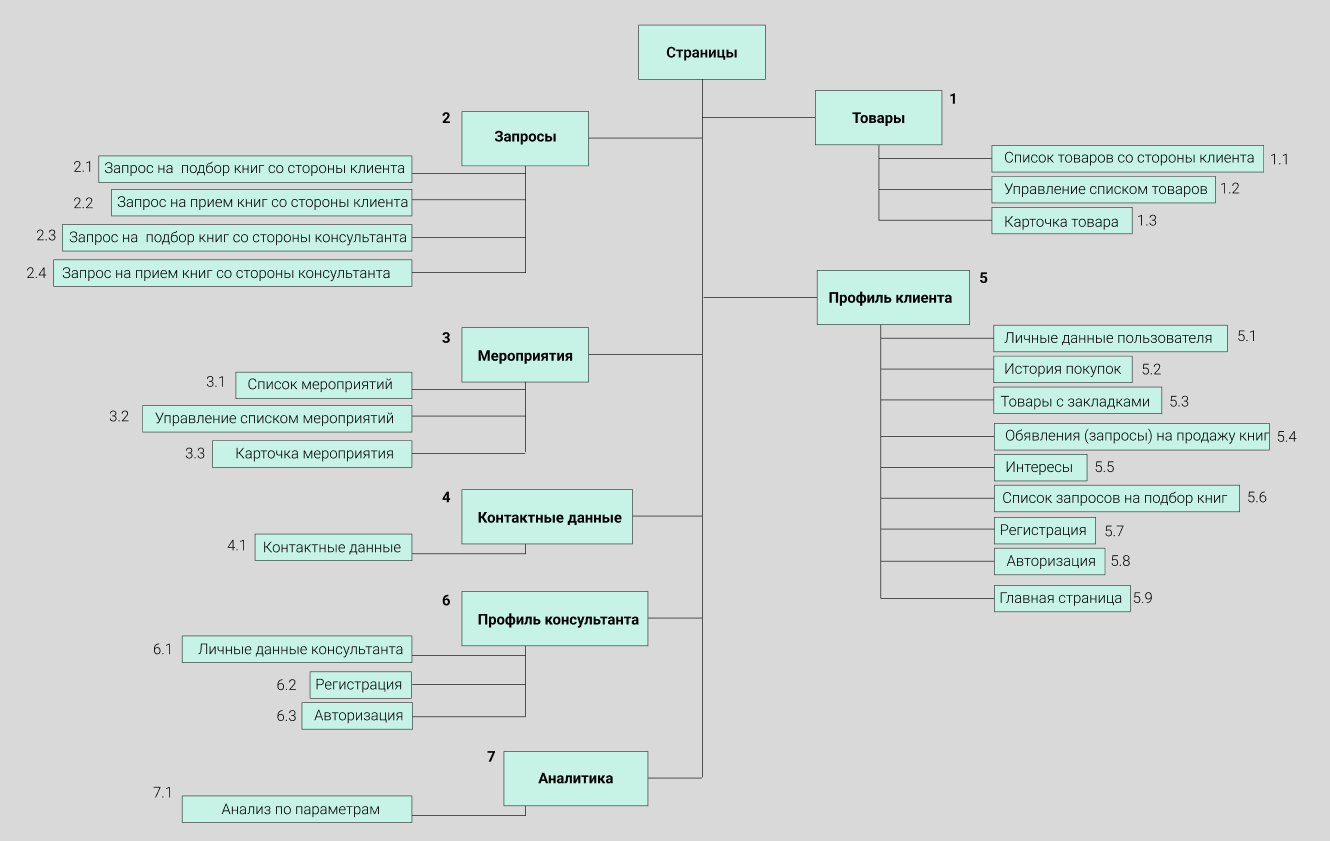


Рисунок – 4 Схема верхнеуровнего определения экранов группируемых по разделам

На рисунке 5 изображена схема иерархии верхнеуровнего определения экранов.

 Рисунок – 5 Схема иерархии верхнеуровнего определения экранов

На рисунках 6, 7 и 8 описано назначение экранов.

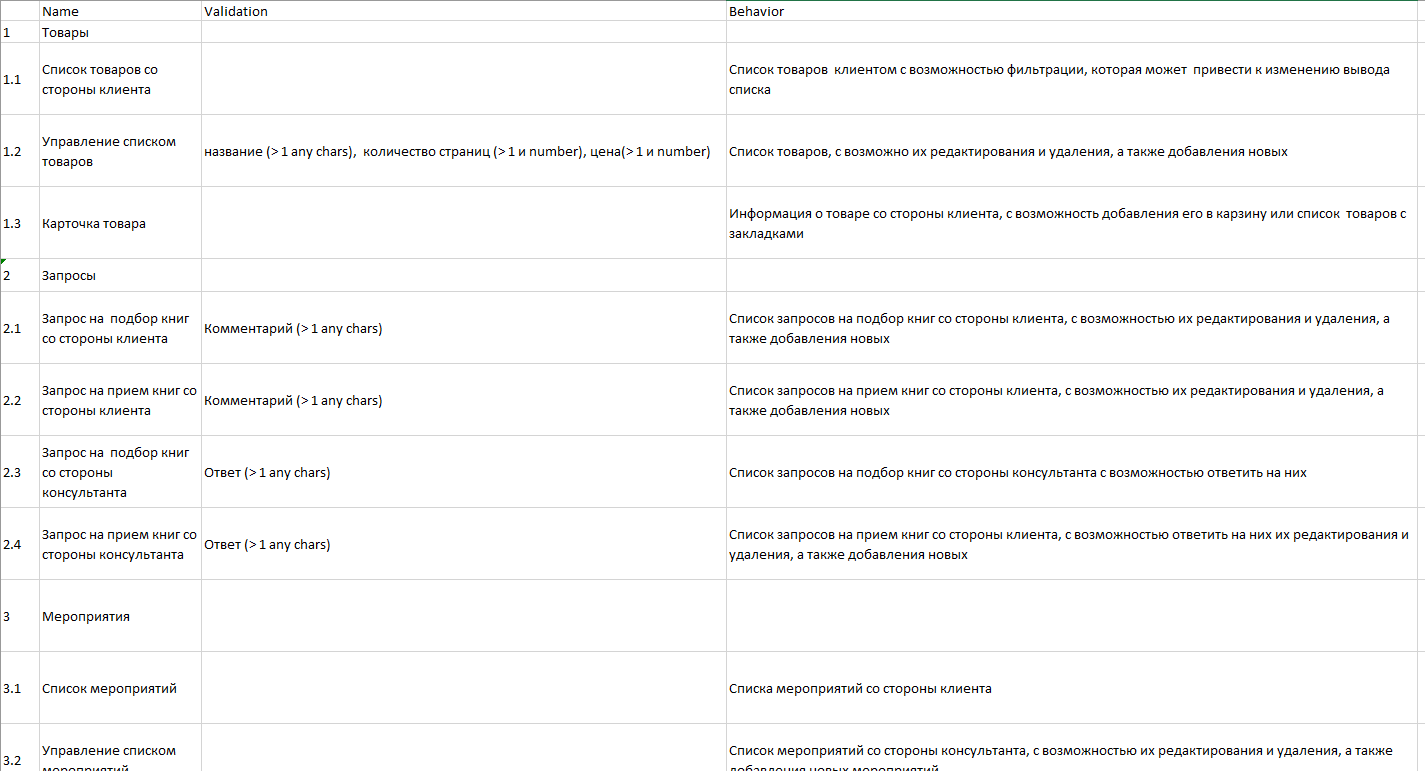


Рисунок – 6 Назначение экранов

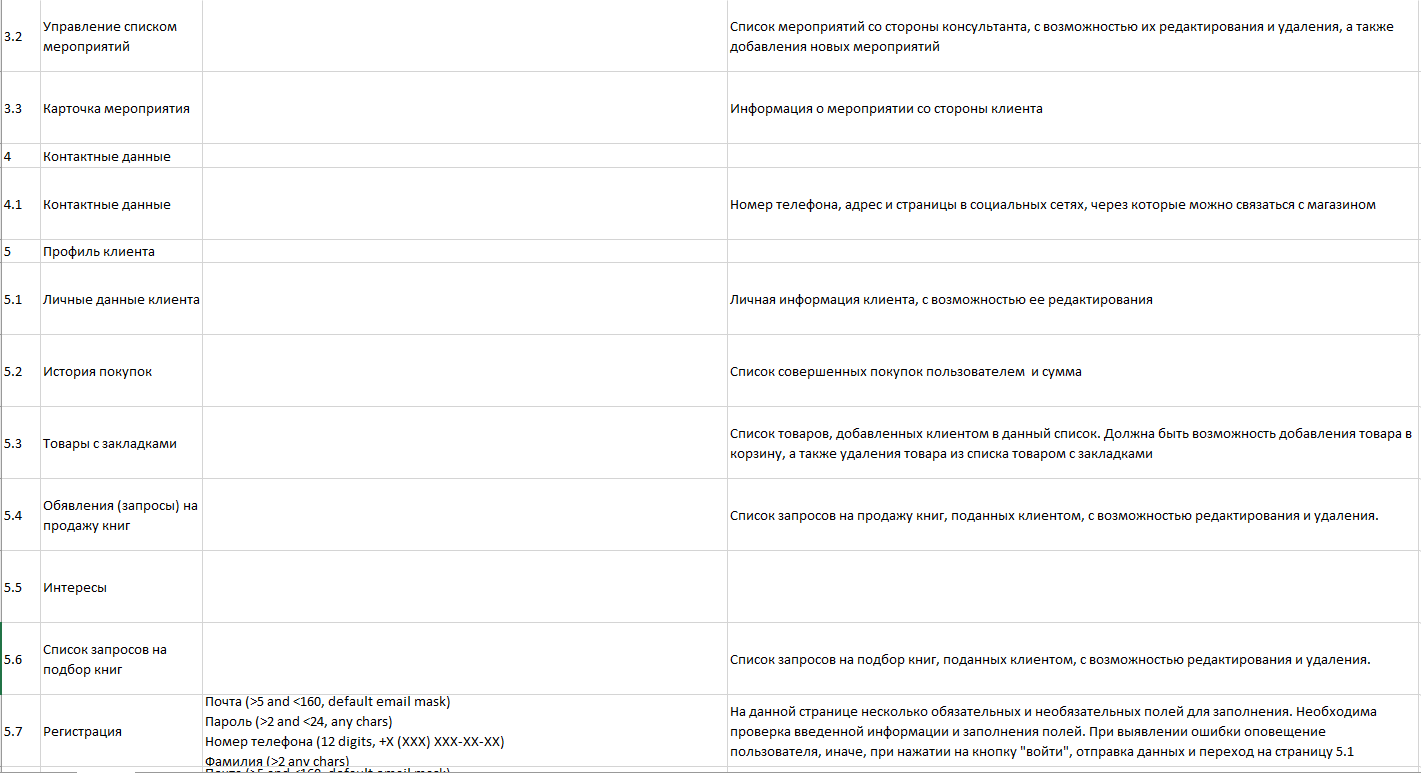


Рисунок – 7 Назначение экранов

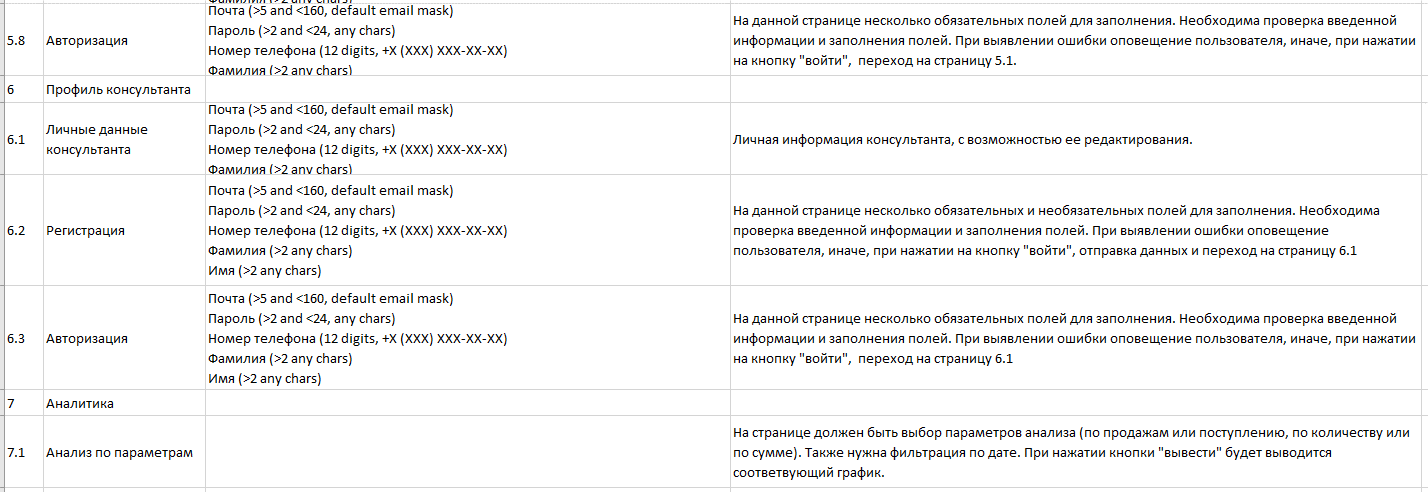


Рисунок – 8 Назначение экранов

Регистрация в системе предусмотрена для учета пользователей и разделения ролей. Регистрация клиентов позволит создавать персонализированные профили, учитывая их предпочтения, историю покупок и другие данные. Это способствует улучшению пользовательского опыта и предоставлению клиентам более релевантных и интересующих их предложений. Регистрация позволяет клиентам отслеживать статусы своих заказов и получать уведомления об изменениях. Это повышает прозрачность и доверие к магазину со стороны клиентов.

На рисунке 9 представлен макет главной страницы. Он состоит из навигационного меню, раздела с объявлениями, а также подборки редких и подарочных изданий, с автографами, а также антиквариат.

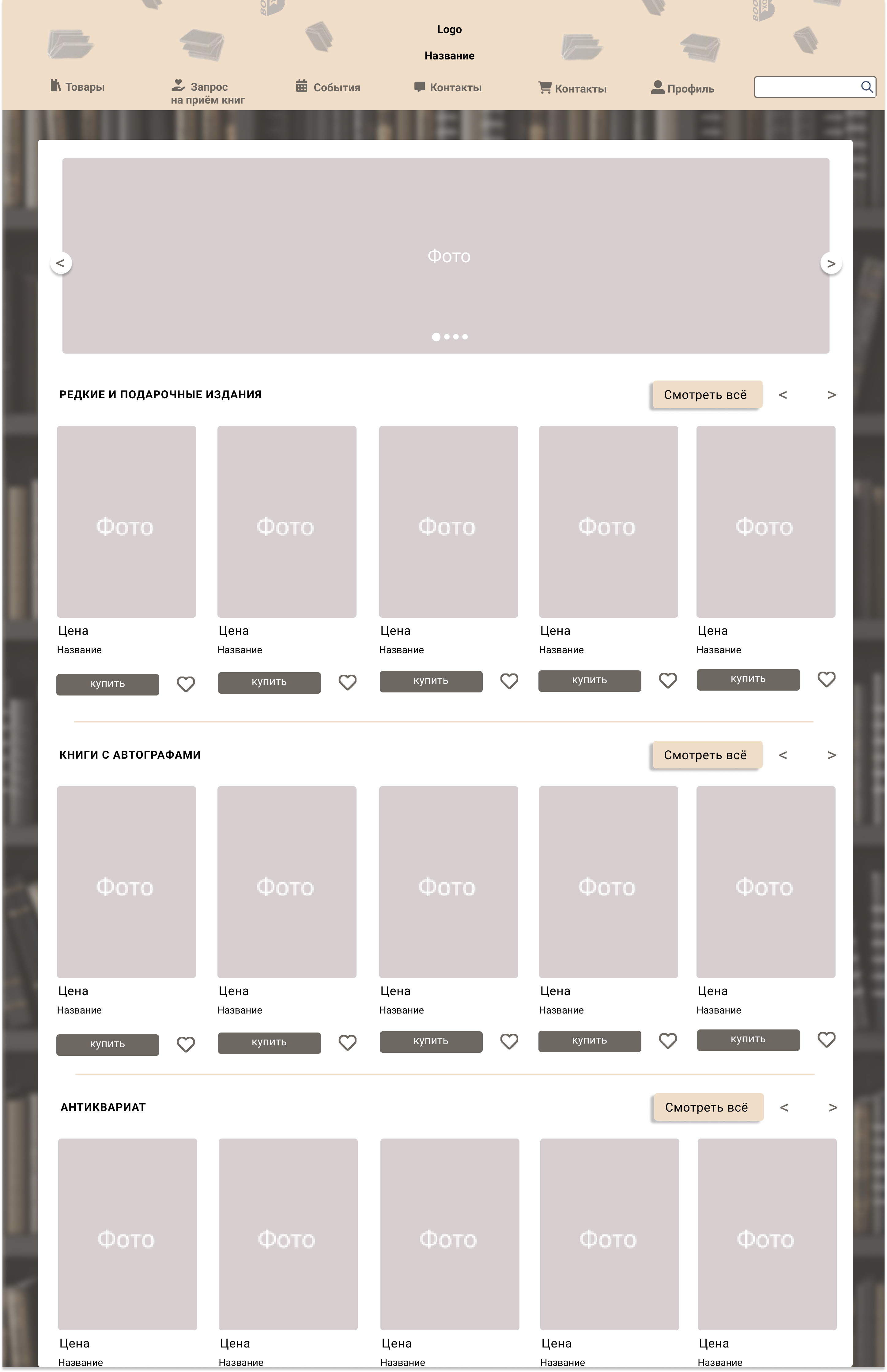


Рисунок 9 – Главная страница

Форма авторизации клиента (рисунок 10) откроются, при попытке неавторизированным пользователем перейти на одну из таких вкладок, как «Запрос на прием», «Корзина» или «Профиль». Чтобы перейти к форме регистрации (рисунок 11), необходимо нажать на подсказку «Не зарегистрированы». Для обратного перехода надо нажать на подсказку «Зарегистрированы».

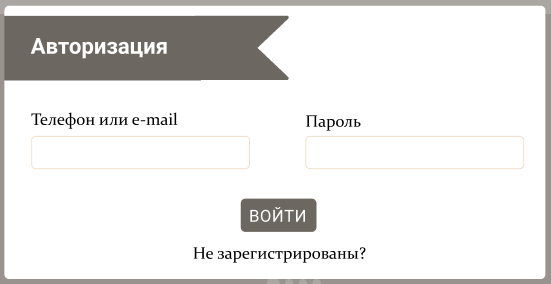


Рисунок 10 – Форма авторизации клиентов

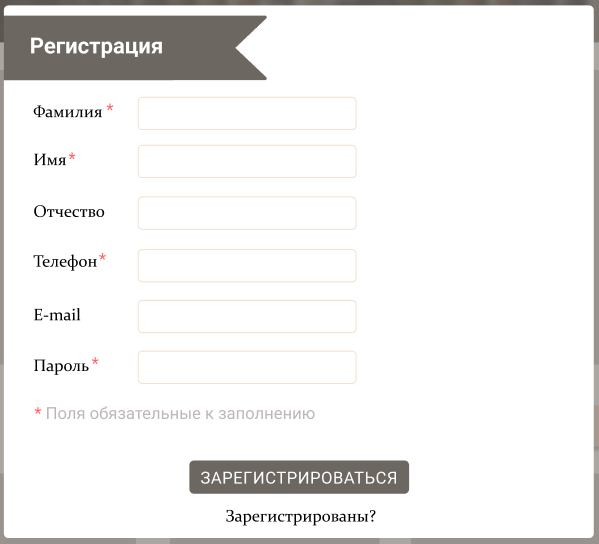


Рисунок 11 – Форма регистрации клиентов

Формы регистрации и авторизации консультантов (рисунки 12 и 13 соответственно) открывается при первом входе на страницу. Навигация между формами аналогична упомянутой выше.

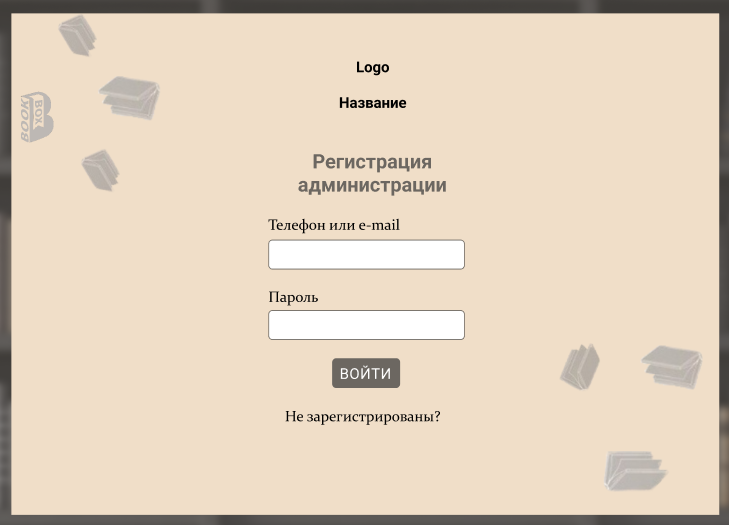


Рисунок 12 – Форма авторизации консультантов

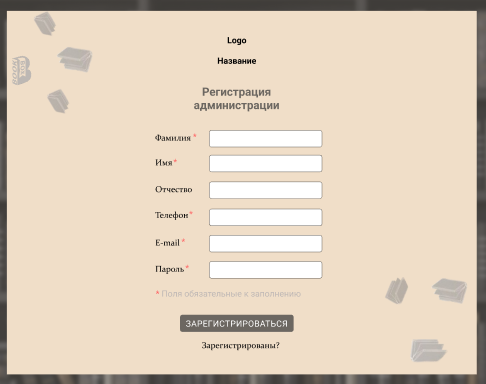


Рисунок 13 – Форма регистрации консультантов

На рисунке 14 представлен макет учета запросов на прием книг.

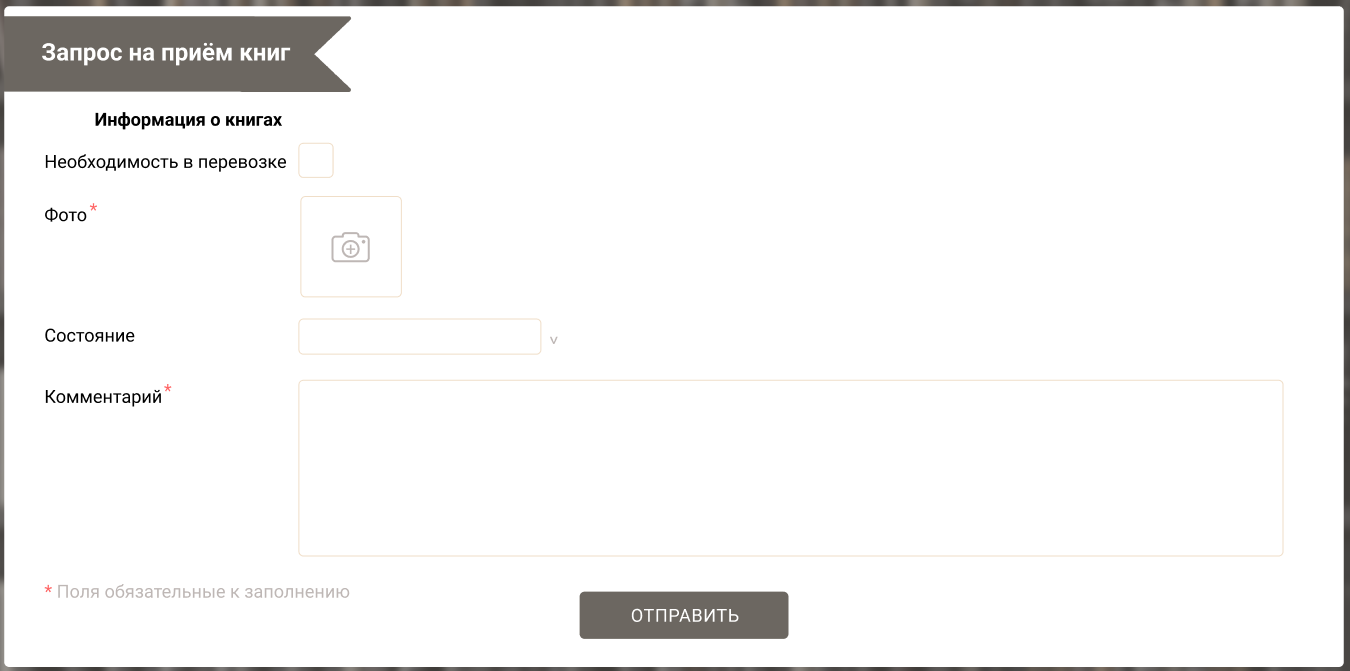


Рисунок 14 – Макет учета запросов на прием книг

Для открытия в меню данного документа необходимо в главном меню выбрать пункт «Запрос на прием книг». Документ смешенной формы. Он состоит из трех полей, два из которые надо заполнять вручную, а один выбирается из списка, выгруженного из базы данных. Некоторые поля обязательны к заполнению. Для оповещения об этом пользователя возле таких полей стоит специальный знак контрастного цвета, а также присутствует примечание, объясняющее значение символа. Сохранение информации в базу данных осуществляется только после нажатия кнопки «Отправить».

На рисунке 15 представлен макет главного меню для консультанта.

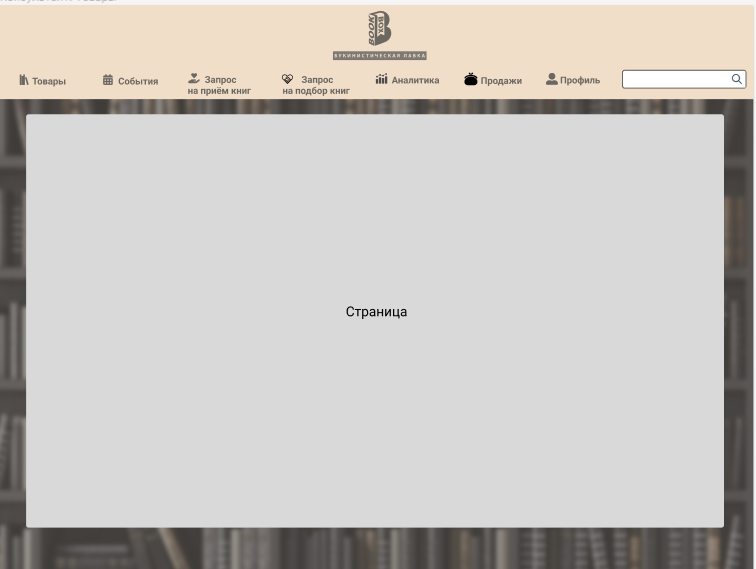


Рисунок 15 – Макет главного меню

На рисунке 16 представлен показан макет анализа продаж и поступлений.

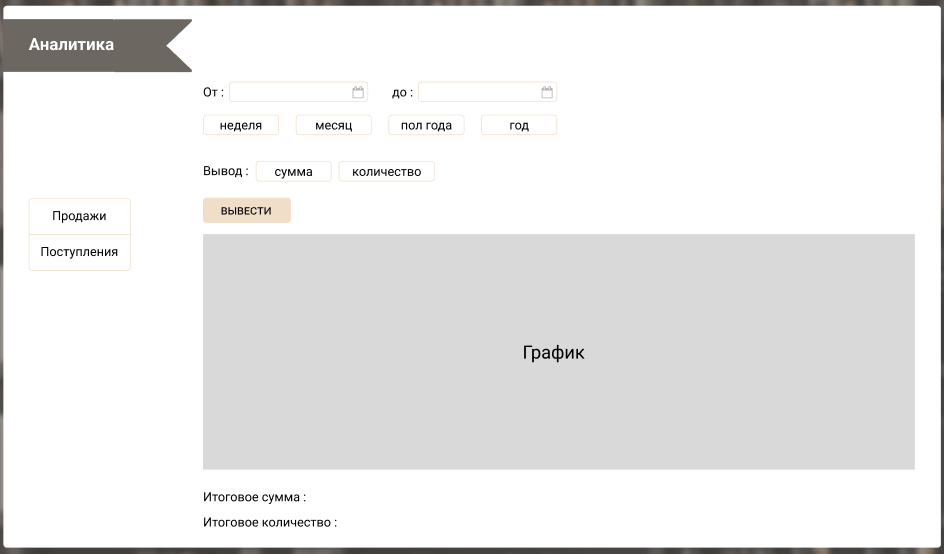


Рисунок 16 – Макет для анализа продаж и поступлений

Для открытия в меню данного документа необходимо в главном меню консультанта выбрать пункт «Аналитика». Документ смешенной формы. Он состоит из двух полей, а один выбирается из выводимого календаря. Выходной документ представлен в формате страницы сайта с графиком и кратким описанием результата анализа. Для просмотра выходного документа необходимо нажать на кнопку «Вывести».

На рисунке 17 изображена карта экранов со стороны клиента

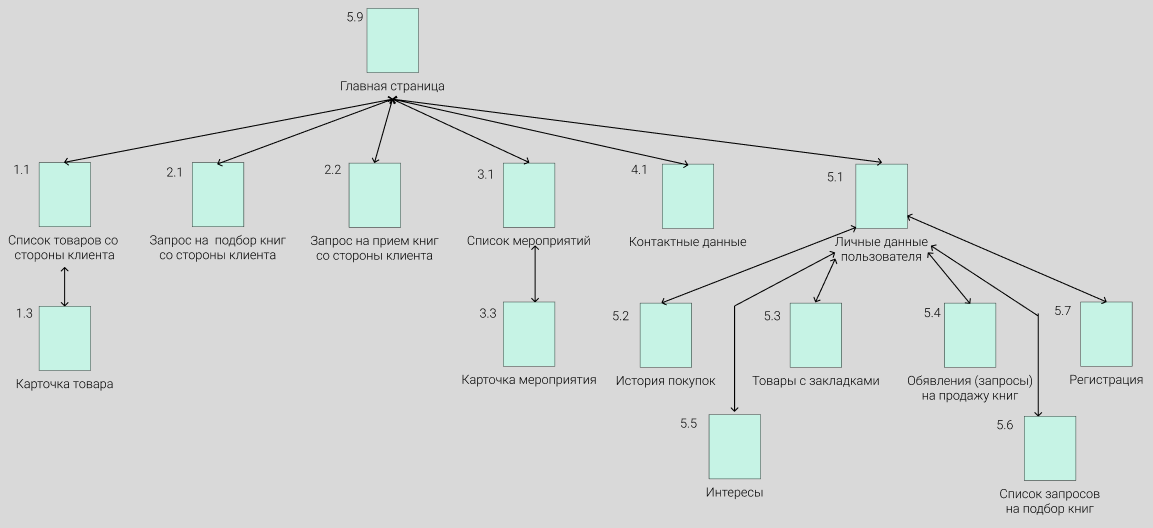


Рисунок 17 – Карта экранов со стороны клиента

На рисунке 18 изображена карта экранов со стороны консультанта

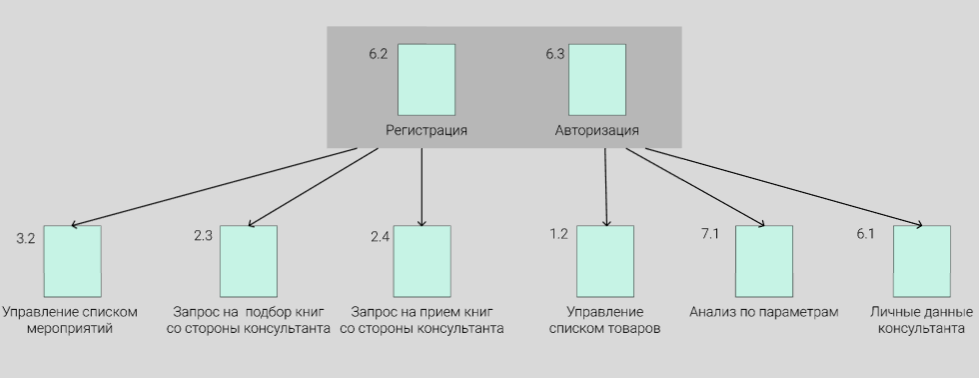


Рисунок 18 – Карта экранов со стороны консультанта