

УДК 159.923.2+004.5

## 8. ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ ОПТИМИЗАЦИИ ПРОЦЕССА ОБРАБОТКИ ЗАЯВОК В АУДИТОРСКОЙ КОМПАНИИ

*Мартынюк Е.Д., студент гр. 072301*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Голда О.А. – ст. преподаватель каф. ЭИ*

**Аннотация.** Актуальность разработки проекта обоснована растущим спросом на сервисы, которые позволят проводить оптимизацию процессов корпоративного менеджмента. Подобный сервис сможет занять экономическую нишу в данной сфере, таким образом обеспечив свое развитие и получение прибыли.

**Ключевые слова.** Аудиторские услуги, обработка заявок, бизнес-процесс, база данных, веб-разработка, UX-элементы, дизайн, язык программирования NodeJS, архитектурные решения.

Разработка и внедрение веб-приложения для оптимизации процесса обработки заявок в аудиторской компании находят обоснование в изменяющихся требованиях организаций и отдельных лиц к управлению бизнес-процессами в условиях постоянного технологического прогресса и перехода к цифровым форматам ведения деятельности. Этот сдвиг стимулирует появление на рынке корпоративного менеджмента и продуктивности потребности в новых инновационных продуктах, способных оптимизировать организацию жизнедеятельности и использование ресурсов. Очевидно, что новая система должна соответствовать стандартам текущего периода развития цифровой эпохи, а именно: быть синхронизируемой, предоставлять быстрый доступ в любое время к любой нужной информации, сохранять приватность и безопасность личных данных пользователя, использовать продвинутые алгоритмы для вычисления и сбора требуемых данных, быть производительной и стабильной на уровне коммерческого B2B ПО, отвечать современным требованиям дизайна и проектирования ПО для масштабируемости, приносить уникальную ценность потребителю и содержать инновации, уникальные для конкретного программного продукта.

Таким образом, целью работы является совершенствование процесса обработки заявок в аудиторской компании путем разработки инструментальных средств поддержки процесса взаимодействия с заявками.

Объектом исследования являются процессы корпоративного менеджмента, формирования и отслеживания заявок. На основе вышеизложенного была поставлена следующая гипотеза: создание и внедрение ПО для сбора и автоматизации процессов, происходящих в корпоративном менеджменте, позволит ускорить, упростить, стабилизировать и оптимизировать обработку заявок в компании.

Пользовательский интерфейс системы реализуется на языке JavaScript(см. рисунок 1).

Основными требованиями к графическому интерфейсу являются:

- наличие анимации для каждого элемента экрана, который меняет состояние или положение;
- соответствие дизайну;
- использование компонентов;
- цветовая гамма должна быть выполнена, используя три основных брендовых цвета ПП;
- содержимое экранов должно быть адаптировано для планшетов и горизонтальной ориентации экрана, где это возможно технически;
- экраны должны сохранять и восстанавливать свое состояние;
- производительность отрисовки стабильна все время пользования приложением.

Навигация происходит путем нажатия на иконки в секции сверху( или сбоку) экрана. Таким образом, была проведена спецификация пользовательского интерфейса и определение основных назначений для навигации пользователя. Была сформирована дизайн-система и макеты интерфейса.

В соответствии с дизайн-стандартами, описанными выше, была реализована последовательность основных страниц приложения.

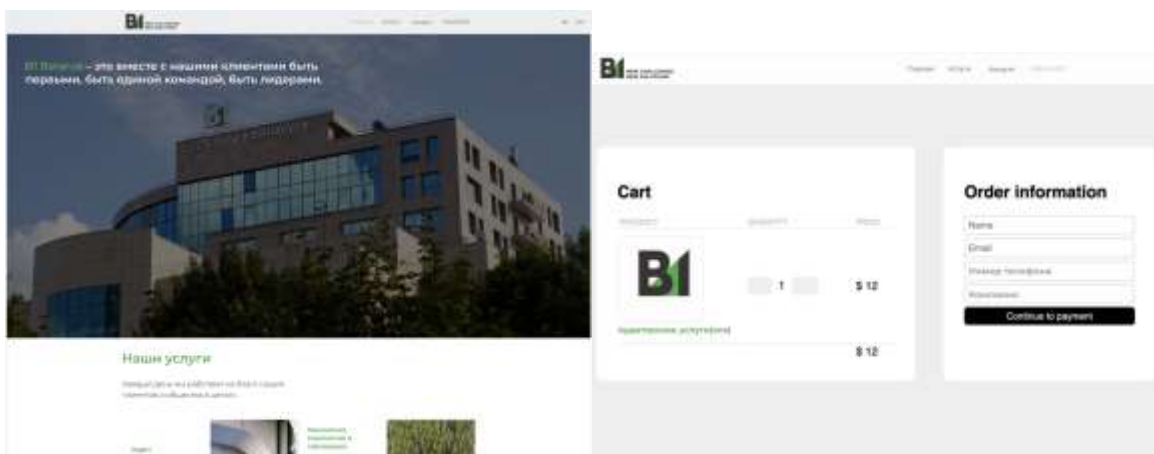


Рисунок 1 – Экраны «Главная» и «Корзина»

Помимо этого, проект организован в соответствии с принципами Clean Architecture, а также разбит на модули согласно последним стандартам Google.

Также в проекте использовалась внешняя система оплаты Stripe. Она позволяет компаниям и предпринимателям принимать платежи через интернет, мобильные устройства и другие каналы. Вот некоторые возможности Stripe: Stripe предлагает простой и удобный API, который позволяет легко интегрировать систему платежей в ваш веб-сайт или приложение; поддерживает различные способы оплаты, включая кредитные и дебетовые карты (Visa, Mastercard, American Express и другие), а также мобильные платежи и альтернативные методы оплаты, такие как Apple Pay и Google Pay; обеспечивает высокий уровень безопасности, следуя современным стандартам шифрования данных и соблюдая требования стандарта безопасности PCI DSS (Payment Card Industry Data Security Standard); Stripe поддерживает платежи в различных валютах и работает во многих странах по всему миру, что делает его идеальным решением для компаний с международной клиентской базой; предоставляет инструменты для отслеживания платежей, формирования отчётов и аналитики, что помогает компаниям понимать свою финансовую деятельность и принимать обоснованные решения. Использование Stripe может значительно упростить процесс приёма платежей онлайн для компаний всех размеров, начиная от стартапов и заканчивая крупными корпорациями.



Рисунок 2 – Экран «Оплата»

За время работы над проектом была реализована парадигма (поведенческая модель) работы приложения и определены стандарты, которым впоследствии будет соответствовать программный код приложения для оптимизации скорости разработки, стабильности, и производительности системы.

На рисунке 3 изображена диаграмма вариантов использования конечной системы. Ключевые варианты использования программного средства выглядят следующим образом:

- авторизация, как администратора;
- управление заказами, услугами, категориями;
- вход в личный профиль;
- оплата заказа;
- вывод статистик;
- редактирование данных компании;
- получение аналитики сайта;
- загрузка данных в бд;
- проверка состояния оплаты;
- загрузка чека оплаты;
- добавление комментария.

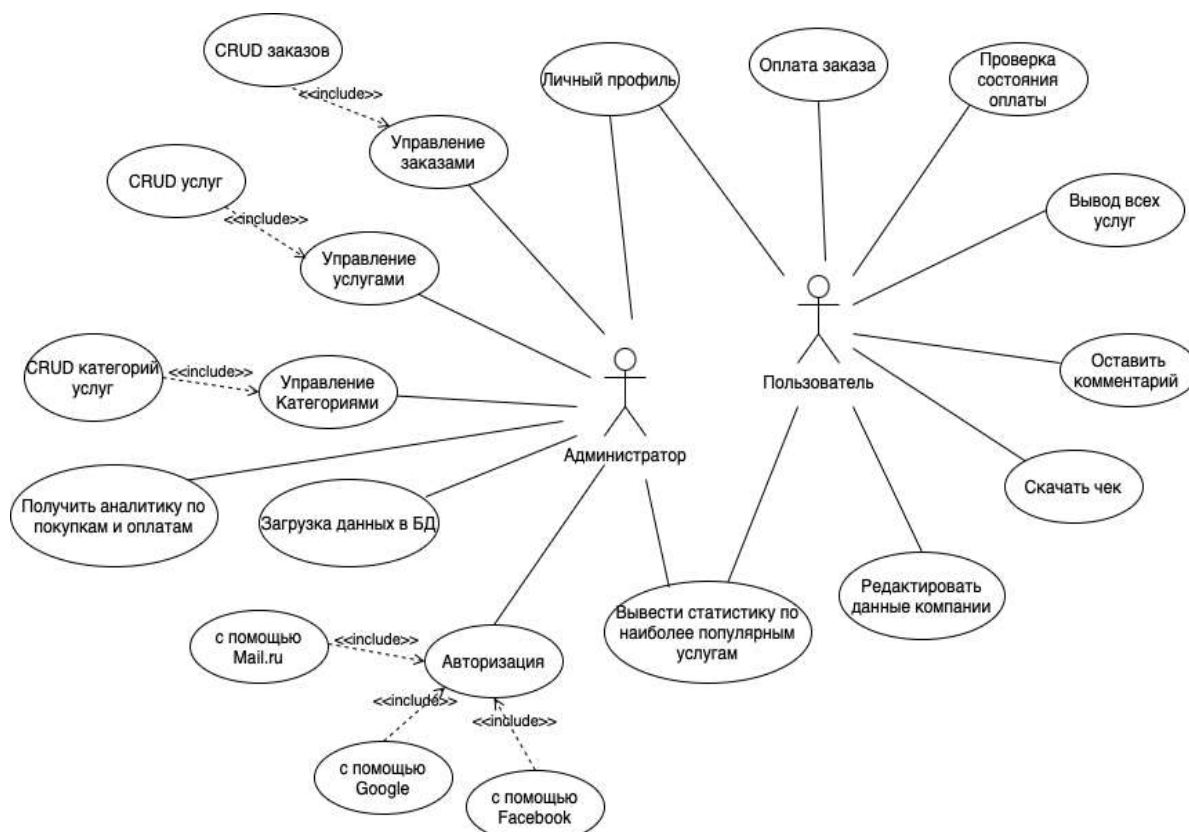


Рисунок 3 – Диаграмма вариантов использования программного средства

На рисунке 4 представлена диаграмма базы данных серверного приложения.

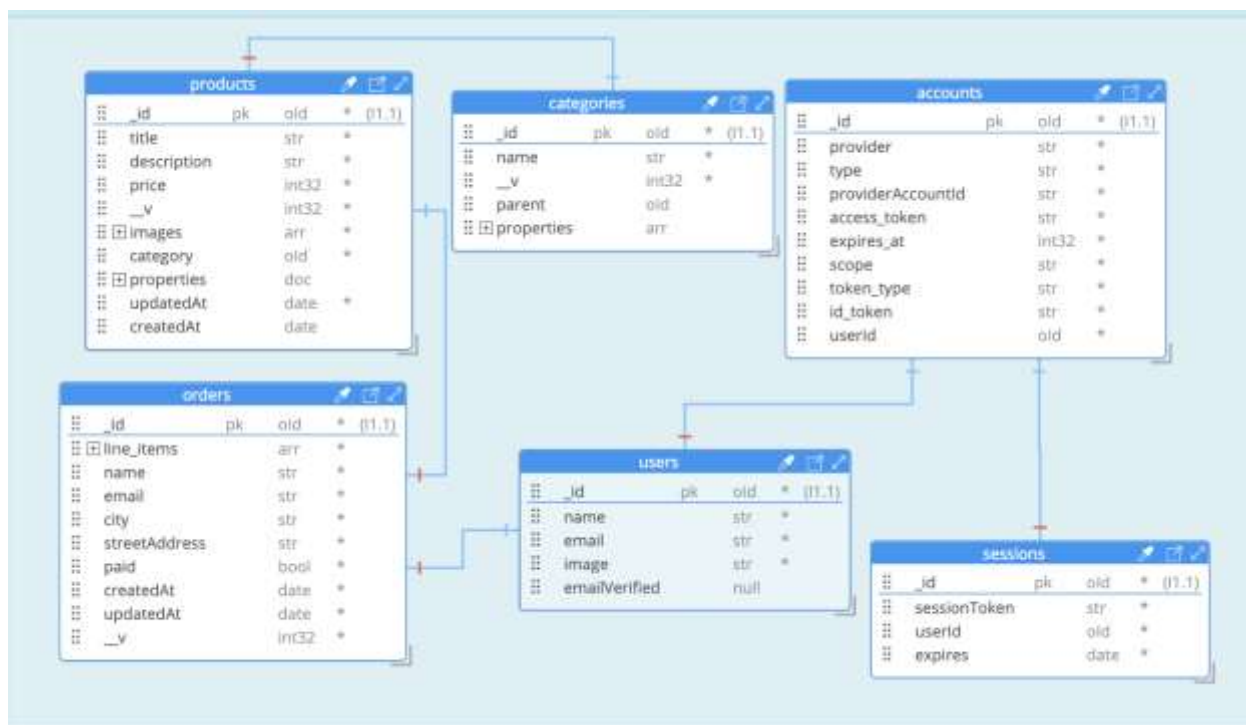


Рисунок 4 – Схема базы данных удаленного сервера

Подобное решение позволит реализовать удаленную синхронизацию данных пользователя, резервное копирование и поддержку нескольких устройств.

Смена конфигурации устройства, а также общее управление зависимостями внутри приложения поддерживается с помощью принципа DI реализуемого с помощью фреймворка NodeJS.

Архитектура серверного приложения следует принципам поддержки неограниченной масштабируемости и единства с принципами работы мобильного приложения. Сервер выполнен на базе архитектуры MVC, что дает возможность оставить его монолитной структурой, а в дальнейшем перейти к использованию микросервисов, вынеся общий код в распространяемую мультиплатформенную библиотеку. Это обеспечит расширяемость кодовой базы. Производительность обеспечивается асинхронным основанным на рабочих потоках механизмом выполнения операций (корутинах).

Таким образом, для реализации требуемых спецификаций, описанных ранее, была разработана уникальная архитектура приложения, а также использованы рекомендуемые архитектурные практики и технологии.

Одной из причин подобных решений стало наличие значительного количества взаимодействующих друг с другом элементов, модулей, систем, исполняемых компонентов. Коммуникация приложения осуществляется не только с удаленным сервером, но и со сторонними сервисами, в основном предоставленными Google, Amazon, Stripe. Коммуникация с базой данных, репликация и синхронизация в реальном времени в фоновом режиме работы устройства также повлияли на комплексность системы. Диаграмма развертывания представлена на рисунке 5.

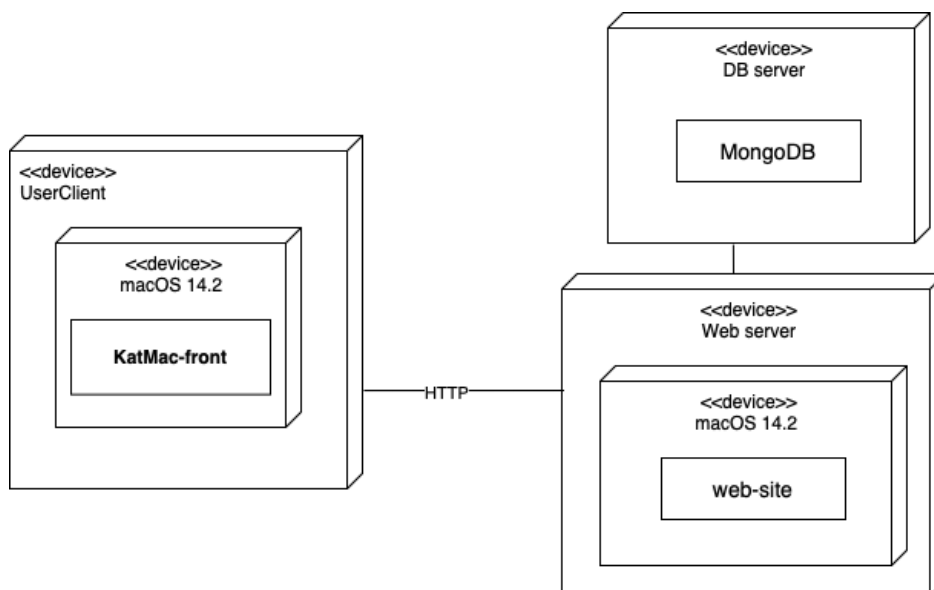


Рисунок 5 – Диаграмма развертывания

Для демонстрации алгоритмов работы системы была выбрана последовательность оплаты покупки при инициации транзакции из приложения. На рисунке 6 приведена диаграмма последовательности процесса добавления компании и заказа этой компании в бд. Как можно видеть, пользователь принимает минимальное участие в процессе, однако выполнение запроса требует значительных объемов коммуникации систем приложения, API приложения.

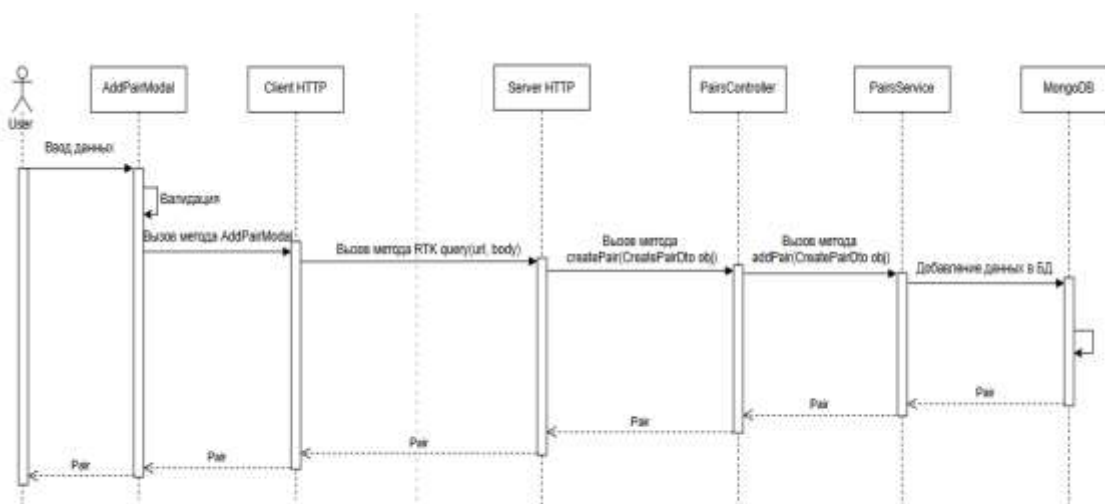


Рисунок 6 – Диаграмма последовательности добавления пары “Компания-заказ”

Подводя итог, была реализована парадигма (поведенческая модель) работы приложения и определены стандарты, которым впоследствии будет соответствовать программный код приложения для оптимизации скорости разработки, стабильности, и производительности системы.

Было проведено как автоматизированное, так и ручное тестирование ключевых компонентов приложения, а также реализована как удаленная, так и локальная валидация содержимого, генерируемого пользователями, что удовлетворяет установленным спецификациям. Описанный функционал был реализован в полном объеме как с точки зрения функциональных, технических и бизнес-требований, так и с точки зрения пользовательского интерфейса приложения.

Внедрение программного обеспечения для обработки заявок обещает значительные преимущества для аудиторской компании. Ускорение, упрощение, стабилизация и оптимизация обработки заявок будут результатом такой инновации. Автоматизация позволит не только улучшить эффективность работы сотрудников, но и сделать компанию более гибкой и конкурентоспособной на рынке. Такой подход оптимизирует процессы, увеличивает прозрачность и позволяет оперативно реагировать на изменения во внешней среде. Как следствие, это приведет к повышению уровня обслуживания клиентов и улучшению общих показателей эффективности компании.

*60-я Юбилейная Научная Конференция Аспирантов, Магистрантов и Студентов БГУИР,  
Минск 2024*

**Список использованных источников:**

1. Лукашевич, Н.П. Теория и практика самоменеджмента: учеб. пособие / Н.П. Лукашевич. - 2-е изд., испр. - К.: МАУП, 2006. - 360 с.: ил.
2. *Android Developers* [Электронный ресурс]. – Режим доступа: <https://developer.android.com>. – Дата доступа: 01.04.2023.
3. *Attenuating Neural Threat Expression with Imagination* / Marianne Cumella Reddan, Tor Dessart Wager 4, Daniela Schiller [Электронный ресурс]. – Режим доступа: [https://www.cell.com/neuron/fulltext/S0896-6273\(18\)30955-3](https://www.cell.com/neuron/fulltext/S0896-6273(18)30955-3). – Дата доступа: 02.04.2023.

УДК 336.49