

Спецификация требований к разработке (SRS)

Процесс оплаты товара пластиковой картой
(покупка товара с использованием платежного терминала или оплатой через сайт компании)

2025 г.

Оглавление

1. Введение	3
1.1 Назначение	3
1.2 Соглашения, принятые в документах	3
1.3 Границы проекта	3
1.4 Ссылки	4
2. Общее описание	4
2.1 Цель проекта	4
2.1 Бизнес-потребность	4
2.3 Классы и характеристики пользователей	4
2.4 Операционная среда	5
2.5 Ограничения дизайна и реализации	5
2.6 Предположения и зависимости	5
3. Пользовательские требования	5
3.1 Пользовательские и технические истории	5
3.2 Варианты использования	6
4. Функции системы	6
4.1 Проведение оплаты банковской картой	6
4.2 Безопасность и защита данных	6
4.3 Интеграции с внешними системами	6
4.4 Уведомления и отчётность	6
4.4 Функциональные требования	7
5. Нефункциональные требования	8
6. Требования к данным	9
6.1 Модель данных	9
6.2 Отчёты	10
6.3 Получение, целостность, хранение и утилизация данных	10
7. Требования к внешним интерфейсам	10
7.1 Пользовательские интерфейсы	10
7.2 API	10
7.3 Интерфейсы оборудования	11
7.4 Коммуникационные интерфейсы	11
8. Атрибуты качества	11
8.1 Удобство использования	11
8.2 Производительность	11
8.3 Безопасность	12
8.4 Техника безопасности	12
9. Прочие требования и описания	12
9.1 Бизнес-процесс	12
9.2 Архитектура	12
9.3 Диаграмма последовательности	13
Приложение А. User Story	14
Приложение В. Use Cases	16
Приложение С. C4	22
Приложение D. Модель данных	27
Приложение Е. Диаграмма последовательности	30
Приложение F. BPMN	31

1. Введение

1.1 Назначение

Настоящий документ определяет требования к системе безналичной оплаты (по пластиковой карте) товаров с использованием платежного терминала или оплатой через сайт в торговой сети. Документ предназначен для:

- формализации требований к разрабатываемой системе,
- обеспечения общего понимания между заказчиком и командой разработки,
- использования в качестве основы для проектирования, реализации, тестирования и поддержки системы.

Данный документ предназначен для следующих заинтересованных сторон:

Со стороны бизнеса

1. Владелец торговой сети – заказчик, инициатор проекта.
2. Менеджер магазина – участник бизнес-процессов, отвечает за организацию работы на месте.
3. Бухгалтер – получает финансовую отчетность.

Со стороны команды

1. Менеджер проекта
2. Системный аналитик
3. Архитектор
4. Разработчики
5. Тестировщики
6. Сотрудники техподдержки

1.2 Соглашения, принятые в документах

В документе используются следующие обозначения:

- **Пользователь** – физическое лицо, совершающее покупку в торговой сети
- **Система** – программно-аппаратный комплекс, реализующий функциональность приёма безналичных платежей
- **Терминал** – устройство самообслуживания или кассовый модуль в магазине, принимающий оплату картой
- **US** – User Story
- **TS** – Technical Story

1.3 Границы проекта

Проект охватывает реализацию и описание требований к системе безналичной оплаты товаров в торговой сети, включая:

- оплату товаров в физическом магазине (через кассы и терминалы самообслуживания),
- оплату товаров онлайн (через сайт компании),
- поддержку различных способов оплаты: банковские карты (магнитная, чип, бесконтактная), СБП, QR-коды и другие.

Вне границ проекта остаются:

- разработка самого сайта или мобильного приложения
- настройка оборудования терминалов

1.4 Ссылки

- PCI DSS – стандарт безопасности данных платёжных карт
- ГОСТ 34.602-89 – Структура технического задания на создание АС
- ISO/IEC 25010 – Стандарты качества программного обеспечения
- Документация API
- Инструкции производителя POS-терминалов

2. Общее описание

2.1 Цель проекта

Цель проекта – внедрение системы безналичной оплаты в торговой сети (на кассе и на сайте), с последующей интеграцией с кассой, сайтом, 1С и бонусной системой.

2.1 Бизнес-потребность

Бизнес-потребность – увеличение скорости оплаты и объема транзакций, повышение удовлетворенности клиентов, снижение операционных издержек.

Приоритет внедрения:

1. Оплата на кассе через терминал
2. Онлайн-оплата на сайте

Срок реализации проекта: 3 месяца

Этапы:

- Этап 1: Терминалы и кассы – 1 месяц
- Этап 2: Онлайн-платежи – 1 месяц
- Этап 3: Интеграция и отчётность – 1 месяц

2.3 Классы и характеристики пользователей

В системе выделены следующие основные классы пользователей, отличающиеся ролями и функциями:

- Покупатель
Основной пользователь системы, который совершает покупку товара через терминал в магазине или онлайн через сайт. Вводит данные карты, выбирает способ оплаты, получает подтверждение и электронный чек. Не имеет доступа к внутренним функциям системы.
- Кассир
Работник магазина, осуществляющий обслуживание покупателей на кассе. Имеет доступ к терминалу, может помогать при ошибках оплаты, проверять статус платежей, инициировать возвраты и формировать отчеты для менеджера.
- Администратор
Ответственный за настройку и управление системой в магазине или на сервере. Управляет пользовательскими правами, контролирует работу оборудования и следит за бизнес-процессом оплаты.
- Бухгалтер
Пользователь, ответственный за финансовую отчетность. Работает с интеграцией в 1С, анализирует отчеты по операциям, контролирует корректность и полноту данных о транзакциях.

- ИТ-поддержка
Технический специалист, обеспечивающий бесперебойную работу системы оплаты. Отвечает за устранение технических сбоев, обновление программного обеспечения и поддержку оборудования.

2.4 Операционная среда

Онлайн-магазин (веб-интерфейс), терминалы самообслуживания и POS-терминалы.

2.5 Ограничения дизайна и реализации

- Использование только сертифицированных POS-терминалов
- Шифрование должно соответствовать стандарту TLS 1.2 или выше
- Интеграция только с зарегистрированными банками-эквайерами
- Устройства с поддержкой NFC и СБП должны быть протестированы на совместимость.

2.6 Предположения и зависимости

- Все магазины оснащены стабильным Интернет-соединением
- Банки-эмитенты поддерживают 3D Secure 2.0
- Платёжный шлюз предоставляет REST API
- Интеграция с 1С осуществляется через установленный коннектор

3. Пользовательские требования

3.1 Пользовательские и технические истории

В рамках проекта по реализации системы оплаты карты были сформулированы ключевые пользовательские истории (User Stories) и технические истории (Technical Stories), отражающие функциональные ожидания со стороны конечных пользователей и системные требования к платформе.

Пользовательские истории (User Stories)

- US-1: Выбор способа оплаты онлайн. Как покупатель, я хочу выбрать способ оплаты (при получении наличными или онлайн по карте или QR), чтобы быстро и удобно оплатить покупку на сайте магазина.
- US-2: Выбор способа оплаты в магазине. Как покупатель, я хочу выбрать способ оплаты (наличными, по карте, бесконтактно или по QR-коду), чтобы оплатить покупку в магазине.
- US-3: Получение чека. Как покупатель, я хочу получить чек на email, чтобы иметь подтверждение оплаты в случае потери бумажного.
- US-4: Обработка возврата. Как покупатель, я хочу вернуть деньги при возврате товара, чтобы не потерять средства.

Технические истории (Technical Stories):

- TS-1: Проверка карты. Как система, я хочу проверять корректность карты (номер, срок, CVV), чтобы избежать ошибочных транзакций.
- TS-2: Шифрование и передача. Как система, я хочу шифровать данные карты и передавать их в платёжный шлюз по защищённому каналу, чтобы обеспечить безопасность.
- TS-3: Интеграция с 1С. Как система, я хочу передавать данные об оплате в 1С для последующего бухгалтерского учёта и отчётности.

3.2 Варианты использования

В ходе развития проекта по реализации системы оплаты товаров по карте были выделены ключевые варианты использования (Use case), охватывающие основные бизнес-процессы взаимодействия покупателя с системой. В их число входят:

- [UC-1](#): Оплата картой через терминал – описывает процесс оплаты товара с использованием банковской карты через терминал в магазине.
- [UC-2](#): Оплата картой онлайн – охватывает сценарий онлайн-оплаты на сайте компании через ввод реквизитов карты.
- [UC-3](#): Возврат средств – описывает процедуру возврата денежных средств по транзакциям, выполненным по карте.
- [UC-4](#): Генерация электронного чека – регламентирует автоматическую отправку чека клиенту после успешной оплаты.
- [UC-5](#): История транзакций – обеспечивает возможность просмотра покупателем всех выполненных транзакций через личный кабинет.

Каждый прецедент был разработан с учетом требований безопасности, бизнес-правил и пользовательского опыта, и обеспечивает прозрачность и надёжность процесса оплаты как для офлайн, так и для онлайн-покупок.

На основе всех выделенных вариантов использования была построена [Use Case диаграмма](#), отражающая взаимодействие участников в рамках указанных сценариев.

4. Функции системы

4.1 Проведение оплаты банковской картой

Система должна обеспечивать возможность оплаты товара банковской картой как на сайте компании, так и в торговых точках через терминалы. Это включает ввод данных карты, их обработку, проверку и проведение транзакции через платёжный шлюз, а также подтверждение оплаты.

Функциональные требования: ФТ1, ФТ2, ФТ3, ФТ4, ФТ5.

4.2 Безопасность и защита данных

Система должна обеспечивать безопасную обработку платёжных данных, включая подтверждение личности пользователя при оплате, защиту персональных данных и корректную обработку ошибок в процессе транзакции.

Функциональные требования: ФТ6, ФТ12, ФТ8.

4.3 Интеграции с внешними системами

Система должна быть интегрирована с внешними информационными системами, включая платёжные шлюзы, 1С и бонусную систему, с целью автоматизации бизнес-процессов и расширения функционала оплаты.

Функциональные требования: ФТ9, ФТ10.

4.4 Уведомления и отчётность

Система должна обеспечивать информирование как пользователей, так и менеджеров о результатах транзакций и возникших ошибках, а также сохранять историю платежей для последующего анализа.

Функциональные требования: ФТ7, ФТ11.

4.4 Функциональные требования

После встречи с заказчиком были выделены ключевые функциональные требования, отражающие поведение системы в процессе безналичной оплаты. Каждое требование сопровождается чёткими критериями приёма для верификации его выполнения.

№	Функция	Описание	Критерии приемки
ФТ1	Ввод данных карты	Система должна позволять пользователю вводить данные карты для оплаты на сайте	После ввода данных карты они передаются в платёжный шлюз, где подтверждается формат
ФТ2	Поддержка терминала оплаты	Система должна принимать оплату по карте через терминал на кассе	При оплате через терминал - успешно проходит транзакция, чек печатается, данные сохраняются.
ФТ3	Передача данных в платёжный шлюз	Система должна передавать данные транзакции в платёжный шлюз	Данные передаются в шлюз (Например, если шлюз возвращает статус 200 – считается успешно)
ФТ4	Получение чека	Система должна генерировать электронный чек и отправлять его пользователю после оплаты	Пользователь получает email с PDF-квитанцией в течение 1 минуты после успешной оплаты.
ФТ5	Уведомление об успешной оплате	Система должна отображать и отправлять уведомление об успешной транзакции	Система отображает сообщение: «Оплата прошла», и отправляет email/SMS уведомление.
ФТ6	Двухфакторная аутентификация	Система должна использовать 3D Secure или аналогичный метод подтверждения	После ввода данных карты запрашивается подтверждение через SMS/мобильный банк/3D Secure.
ФТ7	Сохранение истории платежей	Система должна сохранять историю транзакций для анализа и отчётности	История доступна в личном кабинете администратора и экспортируется в CSV или Excel.
ФТ8	Обработка ошибок и повтор запроса	При сбое оплаты система должна повторить попытку оплаты	При разрыве соединения транзакция повторяется до 3 раз, затем фиксируется как ошибка.
ФТ9	Интеграция с бонусной системой	Система должна поддерживать учёт бонусных баллов при покупке	При наличии бонусов - они списываются по желанию покупателя и отображаются в деталях оплаты.

№	Функция	Описание	Критерии приемки
ФТ10	Интеграция с 1С	Система должна отправлять данные по выручке в 1С	Данные об общей выручке и типах оплат (нал/безнал) передаются в 1С раз в сутки*.
ФТ11	Уведомление менеджеров о сбое оплаты	Система должна отправлять сообщения менеджерам при критических ошибках	В случае ошибки: уведомление с деталями приходит в Telegram и email-менеджерам в течение 1 минуты.
ФТ12	Согласие на обработку персональных данных	Система должна запрашивать согласие пользователя перед оплатой	Перед кнопкой оплаты пользователь должен поставить галочку «Согласен с обработкой персональных данных». Без галочки кнопка оплаты не активна.

5. Нефункциональные требования

Были выделены нефункциональные требования, отражающие характеристики производительности, надёжности, безопасности и удобства системы оплаты. Они определяют качество работы системы и дополняют функциональные возможности.

Производительность:

№	Описание	Критерий приёмки
НФТ1	Время обработки одной оплаты не более 15 секунд.	При нагрузочном тестировании 95% транзакций обрабатываются ≤ 15 секунд.
НФТ2	Поддержка нагрузки до 2500 операций в день.	Система успешно обрабатывает 2500 транзакций в течение 24 часов без деградации.
НФТ3	Максимум – 4 транзакции в минуту в офлайн-магазине.	При имитации 4 транзакций в минуту система не выдаёт ошибок и не тормозит.
НФТ4	Система должна масштабироваться при росте количества клиентов на 30% в год.	При моделировании прироста нагрузки на 30% система остаётся работоспособной.

Надёжность и отказоустойчивость:

№	Описание	Критерий приёмки
НФТ5	Работа кассовой части системы – минимум 10 часов в день без сбоев.	В течение 30 дней не фиксируется ни одного сбоя в работе терминала в дневное время.
НФТ6	Онлайн часть доступна 24/7 без сбоев.	Uptime веб-части $\geq 99,5\%$ за месяц, согласно мониторингу.
НФТ7	Резервирование каналов связи и ИБП при сбоях.	При имитации отказа основной линии система переключается на резервную без потери данных.
НФТ8	Ведение логов об ошибках и действиях пользователей за квартал.	Логи хранятся ≥ 90 дней, содержат время, IP, тип действия, код ошибки (если есть).

Безопасность:

№	Описание	Критерий приёмки
НФТ9	Передача данных по HTTPS + SSL с сертификатом X.509.	При проверке через браузер сертификат HTTPS действителен и имеет X.509-тип.
НФТ10	Маскирование данных карты.	На этапе отображения на сайте/в чеке отображается только 4 последних цифры карты.
НФТ11	Блокировка подозрительных IP-адресов.	Сверка с БД подозрительных IP. Блокировка подозрительной активности на 15 минут (несколько ошибок подряд).
НФТ12	Согласие пользователя на обработку данных.	Без отметки согласия кнопка «Оплатить» остаётся неактивной.

Удобство:

№	Описание	Критерий приёмки
НФТ13	Интерфейс адаптирован для Windows и мобильных браузеров.	При тестировании интерфейс корректно отображается в Chrome/Firefox/Android/IOS.
НФТ14	Интерфейс должен содержать логотип.	На главной и платёжной страницах отображается логотип компании.
НФТ15	Инструкция пользователя + документация для администратора.	В комплекте релиза содержится PDF-инструкция пользователя и admin-документация.
НФТ16	Быстрое обучение персонала по отказам и ошибкам.	Время обучения сотрудника <24 часов. Проверяется по внутреннему регламенту.

6. Требования к данным

6.1 Модель данных

Для хранения информации о покупателях, заказах, товарах и процессах оплаты, была смоделирована [ER - диаграмма](#). Основная цель модели – обеспечить целостность данных и поддержку функционала по приёму платежей по карте.

В разработанной модели были выделены следующие основные сущности:

1. Покупатель (Customer) – содержит данные клиента: ФИО, email и номер телефона.
2. Товар (Product) – включает наименование, цену и категорию товара.
3. Заказ (Orders) – фиксирует сведения о заказе: дата, сумма и привязка к клиенту.
4. Товары в заказе (Orders_product) – отображает состав заказа с указанием количества и стоимости каждого товара.
5. Метод оплаты (PaymentMethod) – описывает тип использованного способа оплаты (наличные, карта)
6. Платёж (Payment) – фиксирует факт оплаты конкретного заказа выбранным способом.
7. Корзина (Cart) – отображает корзину покупателя с указанием количества товара и стоимости каждого товара.
8. Товары в корзине (Cart_product) – отображает какие товары лежат в корзинах.
9. Терминал (Terminal) – отображает информацию о терминале оплаты.
10. Попытка платежа (PaymentAttempt) – хранит данные о попытках платежа.

Между основными сущностями реализованы связи:

- Один покупатель может иметь несколько заказов (1:M)
- Один заказ может содержать несколько товаров. Один товар может быть в нескольких заказах (M:M)
- Один заказ может быть оплачен одним способом (1:1)

На физическом уровне предоставлен [SQL-код](#) для создания базы данных. Он включает команды для создания таблиц, определения первичных и внешних ключей, а также атрибутов каждой сущности.

6.2 Отчёты

- **Журнал транзакций:** включает ID заказа, способ оплаты и дату;
- **Отчёт по сбоям:** список неуспешных транзакций с кодами ошибок;
- **Отчёт по возвратам:** список возвратных операций с указанием причин и сумм;
- **Финансовый отчёт:** за период, сгруппированный по магазинам.

6.3 Получение, целостность, хранение и утилизация данных

- Данные о транзакциях хранятся не менее 5 лет.
- Все персональные и платёжные данные хранятся в зашифрованном виде.
- Удаление данных проводится по расписанию согласно политике хранения.
- Доступ к данным логируется и контролируется через RBAC.

7. Требования к внешним интерфейсам

7.1 Пользовательские интерфейсы

Онлайн-форма на сайте, экран терминала, интерфейс кассы.

7.2 API

Разработанный REST API предназначен для управления процессом оплаты товаров и услуг, включая создание платежей, генерацию чеков и работу с клиентской информацией. API обеспечивает взаимодействие между клиентским приложением (сайтом или POS-системой) и платёжной системой, включая шлюз и банки.

Основные сущности

- Payment (Платёж) – хранит информацию о транзакции: уникальный идентификатор, ID клиента, сумму, валюту и статус;
- Receipt (Чек) – содержит информацию о платеже и email для отправки электронного чека;
- Customer (Клиент) – информация о клиенте: ID, имя, телефон и email.

Основные операции API

Управление платежами (/payments)

- Создание платежа – POST запрос для инициации нового платежа с передачей данных клиента, суммы и валюты.
- Получение информации о платеже – GET запрос по ID платежа.
- Обновление статуса платежа – PATCH запрос для изменения статуса (например, после успешной авторизации).
- Удаление платежа – DELETE запрос по ID платежа.

Работа с чеками (/receipts)

- Создание и отправка чека – POST запрос для генерации электронного чека и отправки его клиенту на email.
- Получение чека по ID – GET запрос для получения данных чека.

Управление клиентами (/customers)

- Создание клиента – POST запрос для регистрации нового клиента.
- Получение информации о клиенте – GET запрос по ID.
- Обновление данных клиента – PUT запрос для изменения информации.
- Удаление клиента – DELETE запрос по ID.

Безопасность

Для доступа к API используется авторизация с использованием JWT-токенов в заголовке Authorization (Bearer).

Сценарий взаимодействия

1. Покупатель инициирует оплату на сайте или в POS, вводит или прикладывает карту.
2. Продавец создает платежный документ и передает данные в Платежный шлюз (Gateway) через HTTPS с шифрованием и токенизацией.
3. Шлюз отправляет платежную информацию в банк-эквайер (Acquirer API), далее запрос проходит через МПС (НСПК), затем в банк-эмитент (Issuer API).
4. Банк-эмитент проверяет данные карты, средства, CVV, 3D Secure и авторизует платеж.
5. Статус успешной или неуспешной оплаты возвращается обратно по цепочке и отображается покупателю.
6. При успешной оплате генерируется и отправляется чек покупателю.

Просмотреть YAML файл можно по [этой ссылке](#).

7.3 Интерфейсы оборудования

- POS-терминалы
- QR-сканеры для оплаты по СБП.
- Серверное оборудование для хостинга платёжного шлюза.
- Принтеры чеков для касс.

7.4 Коммуникационные интерфейсы

- HTTPS/TLS соединения с платёжным шлюзом.
- Веб-сервис (REST API) для связи между сайтом, шлюзом и процессингом.
- Интеграция с 1С по стандарту XML/HTTP.
- Внутренняя сеть магазина с выходом в интернет.

8. Атрибуты качества

8.1 Удобство использования

Простая форма оплаты, короткие сценарии, мобильная адаптация.

8.2 Производительность

Время авторизации ≤ 5 сек, отправка чека ≤ 10 сек.

8.3 Безопасность

SSL, PCI DSS, 3D Secure, проверка на мошенничество.

8.4 Техника безопасности

- POS-терминалы должны быть установлены в зонах, защищённых от постороннего наблюдения.
- Регулярное обновление прошивки терминалов.
- Ограничение физического доступа к серверам.
- Инструкции по реагированию при компрометации оборудования.

9. Прочие требования и описания

9.1 Бизнес-процесс

Для описания бизнес-процесса оплаты товара картой в торговой сети, как в магазине через терминал, так и на сайте компании была разработана [BPMN диаграмма](#). Цель диаграммы – визуализировать основные этапы прохождения транзакции от инициации до завершения, включая проверки и подтверждение оплаты.

Основные участники процесса:

- Клиент – инициирует оплату
- Платёжный шлюз – логика обработки и маршрутизации транзакции
- Эмитент/Эквайер/МПС – обработка и авторизация транзакции.

9.2 Архитектура

Для того чтобы показать, как происходит оплата с помощью карты, была с отражена работа платежного шлюза на диаграмме C4.

1. [Диаграмма контекста](#) - показывает, как система платёжного шлюза вписывается в окружающий бизнес-ландшафт. Показывает взаимодействие человека с системой и внешние системы: Банк-Эквайер, НСПК и Банк-Эмитент.
2. [Диаграмма контейнров](#) - раскрывает архитектуру системы платёжного шлюза на уровне сервисов.
3. Диаграмма компонентов – дает детали для каждого контейнера в виде совокупности компонентов. Основные сервисы:
 - a. [Сервис валидации](#)
 - b. [Сервис шифрования](#)
 - c. [Сервис логирования](#)

При решении задачи по приёму оплаты по карте через сайт и терминал важны надёжность, масштабируемость и безопасность. Поэтому были выделены основные технологий и архитектурные решения:

1. Spring Boot - используются несколько обособленных сервисов (валидация карты, шифрование, логирование, маршрутизация), и Spring Boot подходит для разделённых микросервисов. Также работа с RabbitMQ, валидация по банковским стандартам и интеграция с внешними API (эквайер, эмитент). Spring Boot имеет нужные решения так как можно использовать Spring AMQP, Spring Validation, RestTemplate/WebClient и т.п.
2. RabbitMQ – исходя из требований по обработке 1000 покупателей в день с приростом 30% в год, что является меньше 1 сообщения в минуту. Поэтому, RabbitMQ лучше справляется с точечными событиями и небольшими объемами, которые должны быть точно доставлены. Kafka была бы избыточной и RabbitMQ легче в поддержке, после передачи проекта заказчикам.

3. Сервис шифрования – OpenSSL используется из-за расширения SSL/TLS, используемого в веб-протоколе HTTPS для обеспечения безопасности.
4. Сервис Логирования - Elasticsearch используется для хранения логов, так как она обеспечивает быстрый поиск и аналитику. Также, быстро обрабатывает большие объемы данных
5. JSON - Для внутренних сервисов быстрее передает данные, чтобы быстро завершить оплату.
6. TSL - обязательный стандарт безопасности.
7. ISO 8583 - Стандарт межбанковского взаимодействия.

9.3 Диаграмма последовательности

Составлена [диаграмма последовательности](#), описывающая взаимодействие между основными участниками процесса безналичной оплаты, начиная с формирования платёжного документа продавцом и заканчивая ответом банка-эмитента об успешной или неуспешной авторизации операции.

В процессе участвуют следующие сущности:

- Покупатель - инициирует оплату, прикладывая карту или вводя данные;
- Продавец (POS-система или сайт) - формирует платёжный документ, инициирует транзакцию;
- Платёжный шлюз - обрабатывает платёжные данные, выполняет шифрование и маршрутизацию;
- Банк-Эквайер - перенаправляет запрос в платёжную систему;
- НСПК - определяет банк-эмитент и передаёт ему запрос;
- Банк-эмитент - проводит авторизацию и возвращает результат.

Основные шаги последовательности:

1. Формирование платежа: продавец создаёт платёжный документ в своей системе.
2. Инициация транзакции: покупатель прикладывает карту или вводит данные; система продавца отправляет запрос на оплату в платёжный шлюз.
3. Обработка и маршрутизация: платёжный шлюз шифрует данные и перенаправляет их эквайеру, далее - через НСПК к банку-эмитенту.
4. Авторизация: банк-эмитент проверяет корректность данных карты, наличие средств и правила безопасности (CVV, 3DS и др.), затем возвращает код результата:
 - a. При успешной авторизации - средства резервируются, транзакция подтверждается;
 - b. При неуспешной авторизации - транзакция отменяется.
5. Ответ по цепочке: результат возвращается от эмитента к продавцу через НСПК, эквайера и шлюз.
6. Отображение результата покупателю: при успешной авторизации - подтверждение и чек, при отказе - сообщение об ошибке.

Таким образом, диаграмма отражает как основной сценарий (успешная оплата), так и альтернативный (отказ в авторизации), включая типы взаимодействий (HTTPS, ISO 8583), шифрование, коды авторизации и логирование.

Приложение А. User Story

ID	Краткое наименование	Описание	Предусловия	Критерии приемки	ФТ
US-1	Выбор способа оплаты онлайн	Как покупатель, я хочу выбрать способ оплаты (при получении наличными или онлайн по карте или QR), чтобы быстро и удобно оплатить покупку на сайте магазина	<p>1. Товар есть в наличии и добавлен в корзину*.</p> <p>2. Доступна оплата при получении наличкой или онлайн по карте или QR</p>	<p>1. Способ оплаты выбирается из 3 предложенных опций</p> <p>2. Оплата происходит в соответствии с выбранным способом</p>	ФТ1 ФТ12
US-2	Выбор способа оплаты в магазине	Как покупатель, я хочу выбрать способ оплаты (оплата наличкой, через терминал для оплаты по карте, бесконтактной оплате или по QR-коду), чтобы быстро и удобно оплатить покупку в магазине.	<p>1. Товар есть в наличии.</p> <p>2. Есть терминал для оплаты по карте, бесконтактной оплате или по QR-коду.</p> <p>3. Есть возможность принять наличку</p>	<p>1. Способ оплаты выбирается из 4 предложенных опций</p> <p>2. Оплата происходит в соответствии с выбранным способом.</p>	ФТ2
US-3	Получение чека	Как покупатель, я хочу получить чек на email, чтобы иметь подтверждение оплаты в случае потери бумажного	Оплата прошла успешно	<p>1. Чек генерируется и отправляется в формате PDF</p> <p>2. Чек приходит в течение 1 минуты</p> <p>3. Содержит детали оплаты (время, сумма, наименование)</p>	ФТ4
US-4	Обработка возврата	Как покупатель, я хочу вернуть деньги при возврате товара, чтобы не потерять средства.	Товар возвращён и оформлен возврат	<p>1. Деньги возвращаются в зависимости от способа оплаты (нал/безнал)</p> <p>2. В системе фиксируется факт возврата</p> <p>3. История обновляется</p>	ФТ7

ID	Краткое наименование	Описание	Предусловия	Критерии приемки	ФТ
TS-1	Проверка карты	Как система, я хочу проверять корректность карты, чтобы избежать ошибочных транзакций	Данные карты получены	1. Проверка номера, срока*, CVV 2. Некорректные данные - ошибка 3. Проверка длится не более 2 сек	ФТ1
TS-2	Шифрование и передача	Как система, я хочу шифровать данные карты и передавать их в платёжный шлюз, чтобы обеспечить безопасность	Данные карты прошли проверку	1. Передача по HTTPS + SSL 2. Данные зашифрованы	ФТ3 ФТ6
TS-3	Интеграция с 1С	Как система, я хочу отправлять данные об оплате в 1С, чтобы обеспечить бухгалтерскую отчётность	Произведена оплата	1. Данные по оплате передаются раз в сутки 2. Учитываются нал/безнал 3. Отчёт формируется в CSV	ФТ10

Приложение В. Use Cases

Use Case-1

UC-1. Оплата картой через терминал	
Описание	Покупатель оплачивает товар с помощью банковской карты через терминал в магазине
Участники	Покупатель, Продавец, Платежный шлюз, Банк-Эквайер
Факторы запуска	Покупатель выбрал товар и решает оплатить товар Терминал находится в рабочем состоянии и подключен к сети
Триггер	Покупатель выбрал способ оплаты по карте и прикладывает карту к терминалу
Постусловия	Средства списаны с карты покупателя. Чек с информацией об оплате печатается и, по желанию, генерируется и отправляется покупателю по электронной почте
Основные действия	<ol style="list-style-type: none">1. Покупатель выбирает товар2. Покупатель выбирает оплатить товар по карте через терминал3. Терминал отправляет запрос на проверку данных карты.4. Платежный шлюз передает запрос в банк-эквайер.5. Банк-эквайер проверяет подлинность карты и наличие средств.6. Платежный шлюз отправляет результат в терминал.7. Покупатель получает подтверждение успешной оплаты.8. Чек генерируется и отправляется клиенту на электронную почту
Исключения	Терминал не может считать данные с карты (например: неисправность терминала). Недостаточно средств на карте. Ошибка в процессе связи с банком.
Альтернативы	Использовать другую карты для оплаты. Использовать другой способа оплаты (например: наличные, QR).
Бизнес - правила	Все виды карт принимаются (Visa, MasterCard, Мир и т. д.). Оплата должна быть завершена в течение 15 секунд. Комиссия за транзакцию оплачивается продавцом.
Специальные требования	Генерация чека в электронном виде.
Закрывает ФТ	ФТ2, ФТ3, ФТ4, ФТ5, ФТ6

Use Case-2

UC-2. Оплата картой онлайн	
Описание	Покупатель оплачивает товар через сайт с использованием банковской карты
Участники	Покупатель, Платежный шлюз, Банк-Эквайер
Факторы запуска	Покупатель выбрал товар на сайте и решает оплатить товар Сайт поддерживает оплату через банковскую карту
Триггер	Покупатель нажимает кнопку "Оплатить" на странице оформления заказа
Постусловия	Средства списаны с карты покупателя. Чек с информацией об оплате генерируется и отправляется клиенту на электронную почту
Основные действия	<ol style="list-style-type: none">1. Клиент оформляет заказ на сайте и выбирает способ оплаты онлайн по карте2. Система направляет покупателя на страницу для ввода данных карты.3. Покупатель вводит данные карты (номер, срок действия, CVV).4. Платежный шлюз отправляет запрос в банк-эквайер для проверки.5. Банк-эквайер проверяет карту и наличие средств.6. Платежный шлюз уведомляет сайт о результате проверки.7. Покупатель получает уведомление об успешной оплате.8. Чек генерируется и отправляется покупателю.
Исключения	Ошибка ввода данных карты. Недостаточно средств на карте. Ошибка при подключении к банку.
Альтернативы	Повторный ввод данных карты
Бизнес - правила	3D Secure должен быть включен для каждой транзакции. Оплата должна быть завершена в течение 15 секунд. Комиссия за транзакцию оплачивается продавцом.
Специальные требования	Генерация чека в электронном виде.
Закрывает ФТ	ФТ1, ФТ3, ФТ4, ФТ5, ФТ6, ФТ12

Use Case-3

UC-3. Возврат средств	
Описание	Покупатель хочет получить возврат средств после оплаты по карте
Участники	Покупатель, Продавец, Платежный шлюз, Банк-Эквайер
Факторы запуска	Покупатель выбрал товары, за которые хочет получить возврат. Есть функционал возврата
Триггер	Покупатель оформляет возврат средств по товару
Постусловия	Средства возвращены на карту клиента Покупатель получает уведомление о возврате средств
Основные действия	<ol style="list-style-type: none">1. Покупатель обращается с запросом на возврат2. Продавец подтверждает возврат в системе3. Платежный шлюз инициирует возврат4. Банк-эквайер передает запрос на возврат5. Возврат одобрен6. Покупатель получает подтверждение о возврате.
Исключения	Возврат невозможен (например, по причине превышения срока возврата) Проблемы с возвратом средств из-за технических ошибок
Альтернативы	Отказ от возврата средств.
Бизнес - правила	Возврат средств возможен только в течение 14 дней с момента получения товара
Специальные требования	Процесс возврата должен быть завершен в течение 14 дней с момента оформления*
Закрывает ФТ	ФТ7

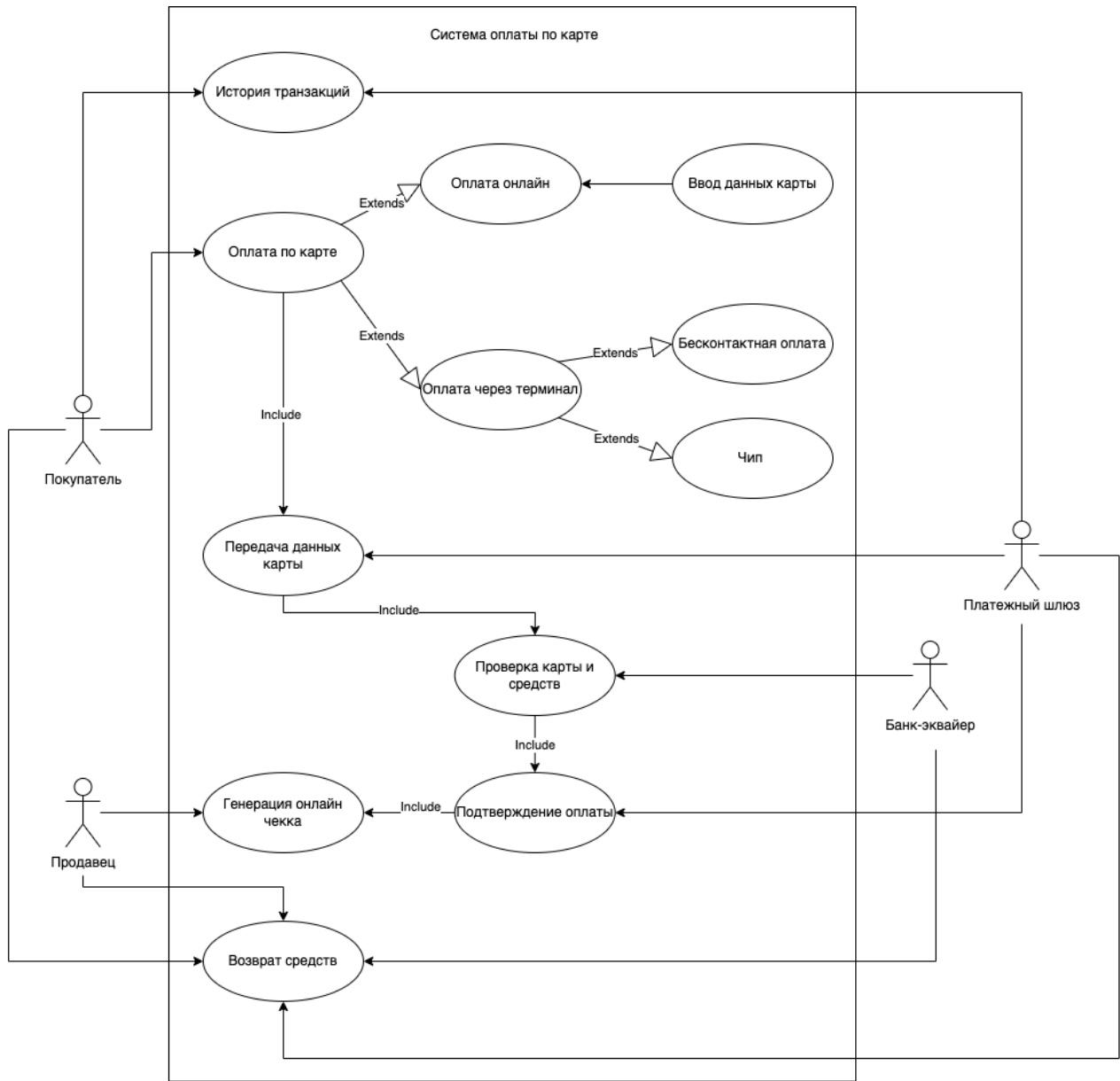
Use Case-4

UC-4. Генерация электронного чека	
Описание	Покупатель хочет получить электронный чек на почту
Участники	Покупатель, Продавец, Система
Факторы запуска	Покупатель завершил процесс оплаты товара по карте Платеж прошел успешно, и система готова отправить чек.
Триггер	Система получает подтверждение об успешной оплате
Постусловия	Электронный чек с данными отправлен клиенту на указанный email или через SMS.
Основные действия	<ol style="list-style-type: none">1. Система фиксирует успешный платёж.2. Система формирует данные чека.3. Уведомление с деталями покупки отправляется клиенту.4. Электронный чек прикладывается к письму или SMS.5. Клиент получает чек и подтверждение.
Исключения	Неправильные данные покупателя (email или номер телефона) Ошибка при генерации чека
Альтернативы	Возможность сохранять чек в ЛК на сайте Отказаться от получения чека
Бизнес - правила	Электронный чек должен содержать обязательные данные (дата, сумма, наименование) Отправляются только при успешной оплате.
Специальные требования	Уведомления должны отправляться в течение 5 минут после успешной оплаты. Чек должен быть доступен в электронном виде (PDF)
Закрывает ФТ	ФТ4

Use Case-5

UC-5. История транзакций	
Описание	Покупатель может просматривать историю своих платежей
Участники	Покупатель, Система
Факторы запуска	Покупатель решает посмотреть историю своих транзакций
Триггер	Покупатель входит в свою учетную запись и выбирает "История транзакций"
Постусловия	Покупатель видит список всех своих транзакций
Основные действия	<ol style="list-style-type: none">1. Покупатель авторизуется в системе.2. Выбирает раздел "История транзакций".3. Система отображает список транзакций клиента
Исключения	Нет записей в истории транзакций. Ошибка отображения
Альтернативы	Покупатель может обратиться в службу поддержки для получения истории транзакций
Бизнес - правила	История транзакций доступна только авторизованным пользователям.
Специальные требования	Обновляется в течение 5 минут после покупки товара
Закрывает ФТ	ФТ7

Use Case Диаграмма



Приложение С. С4
Диаграмма контекста

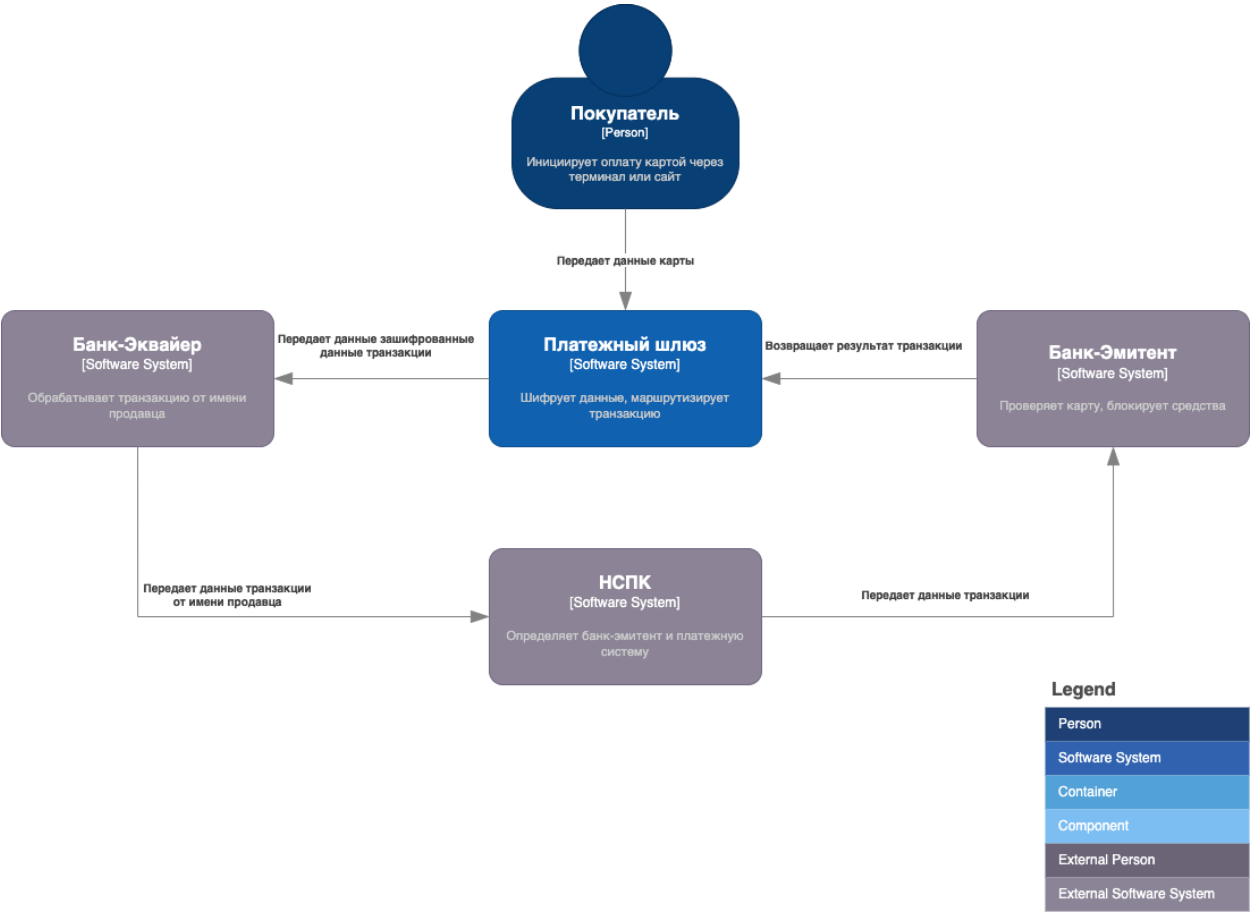


Диаграмма контейнеров

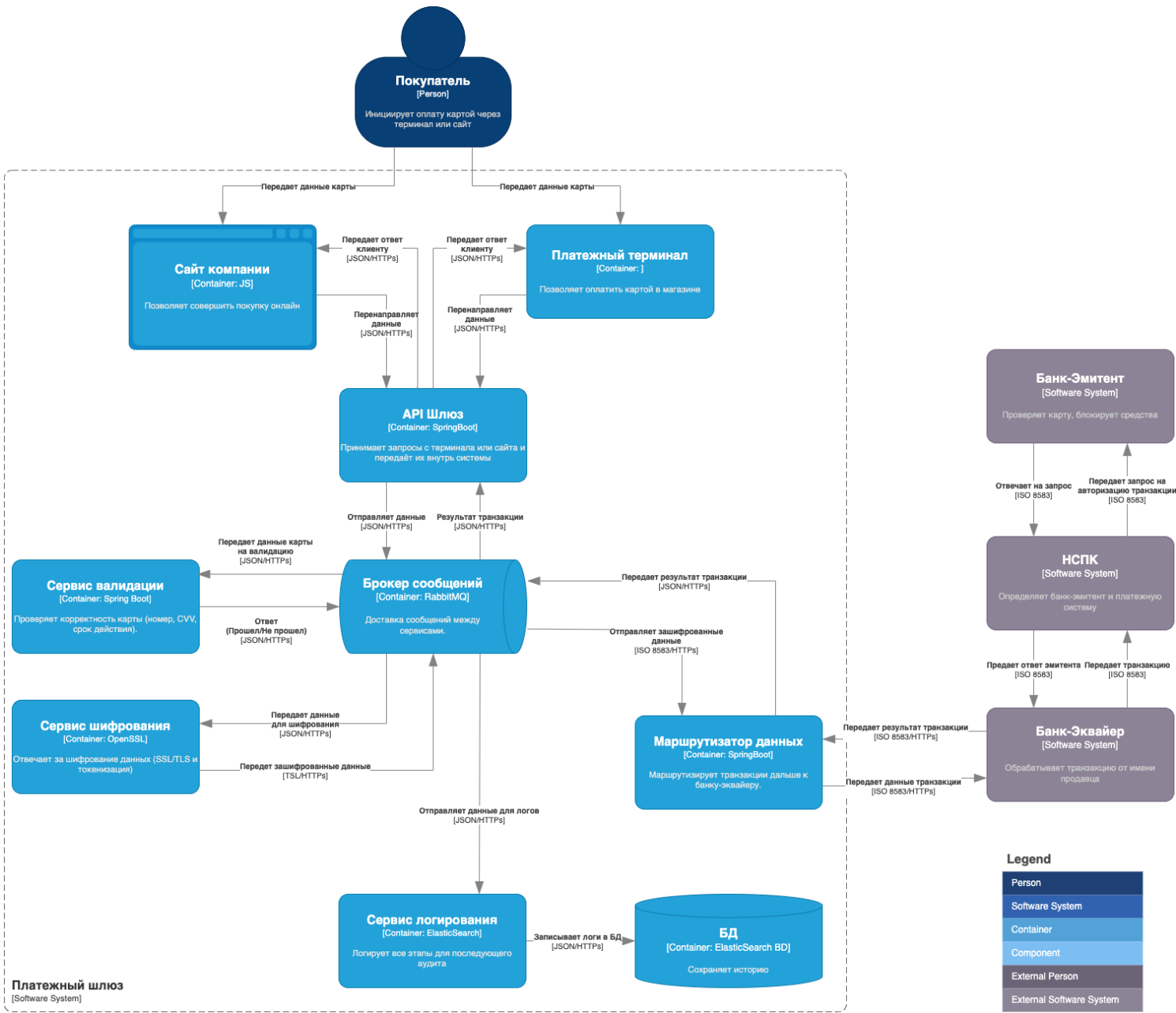


Диаграмма компонентов. Валидация

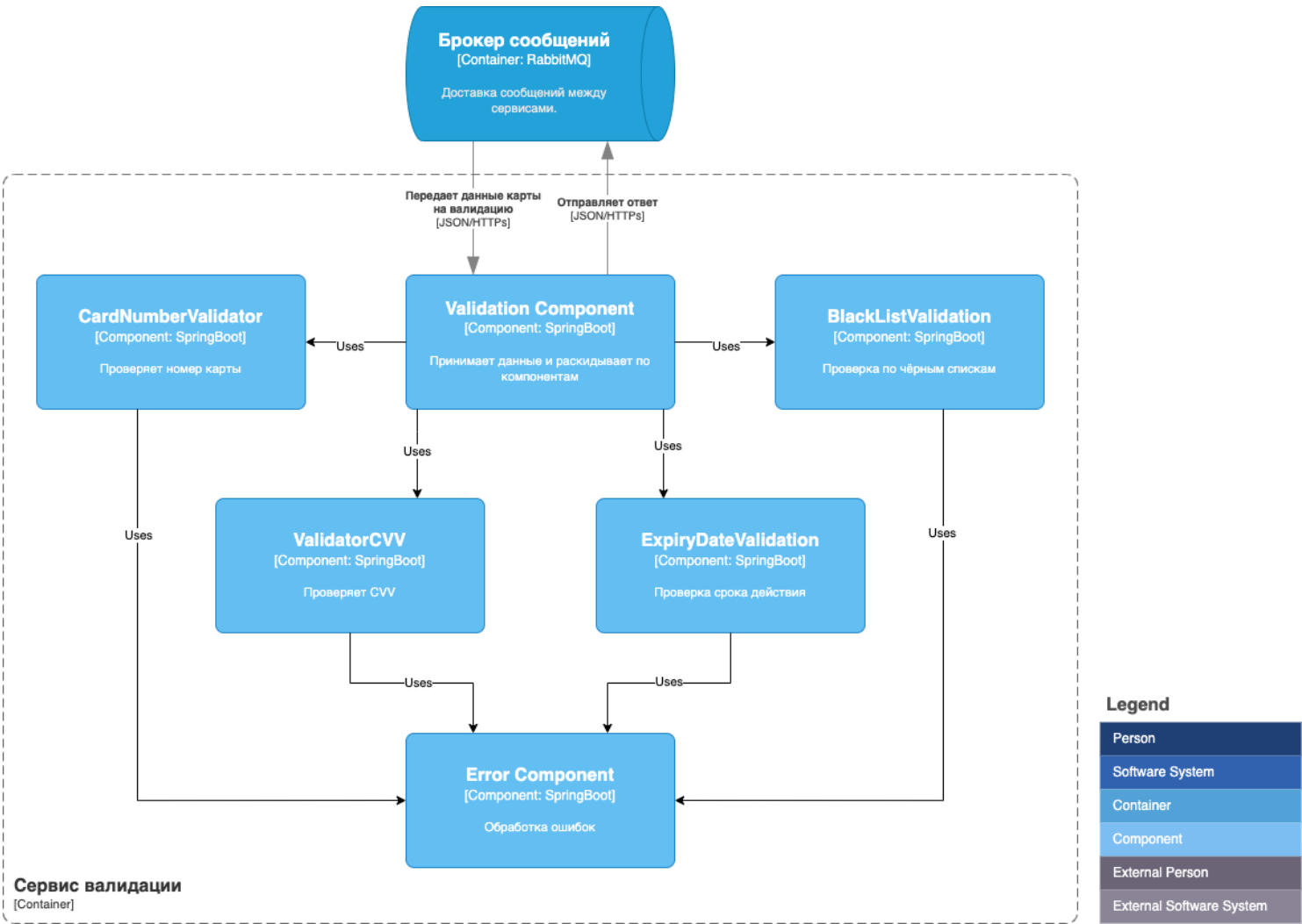


Диаграмма компонентов. Шифрование

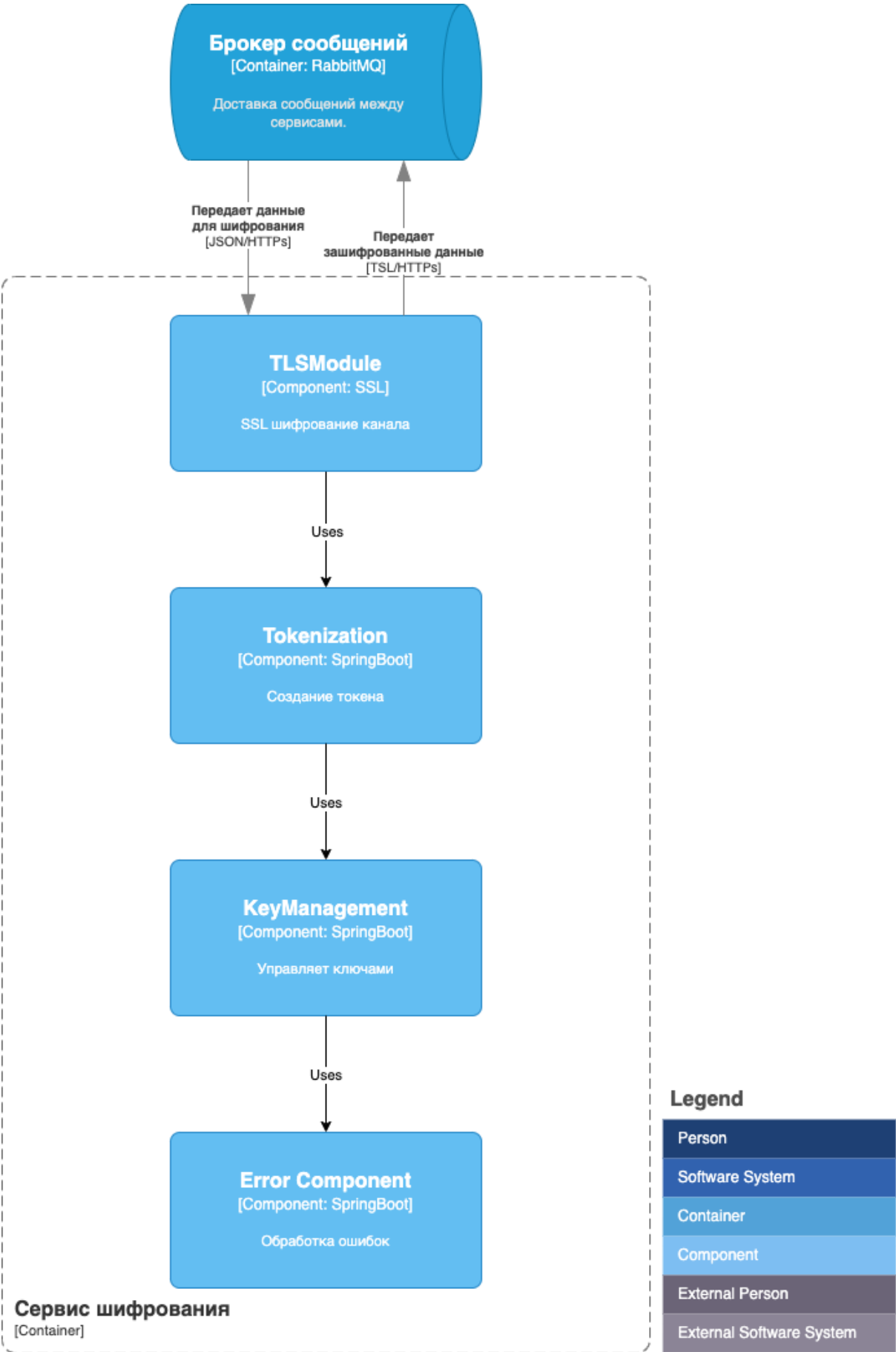
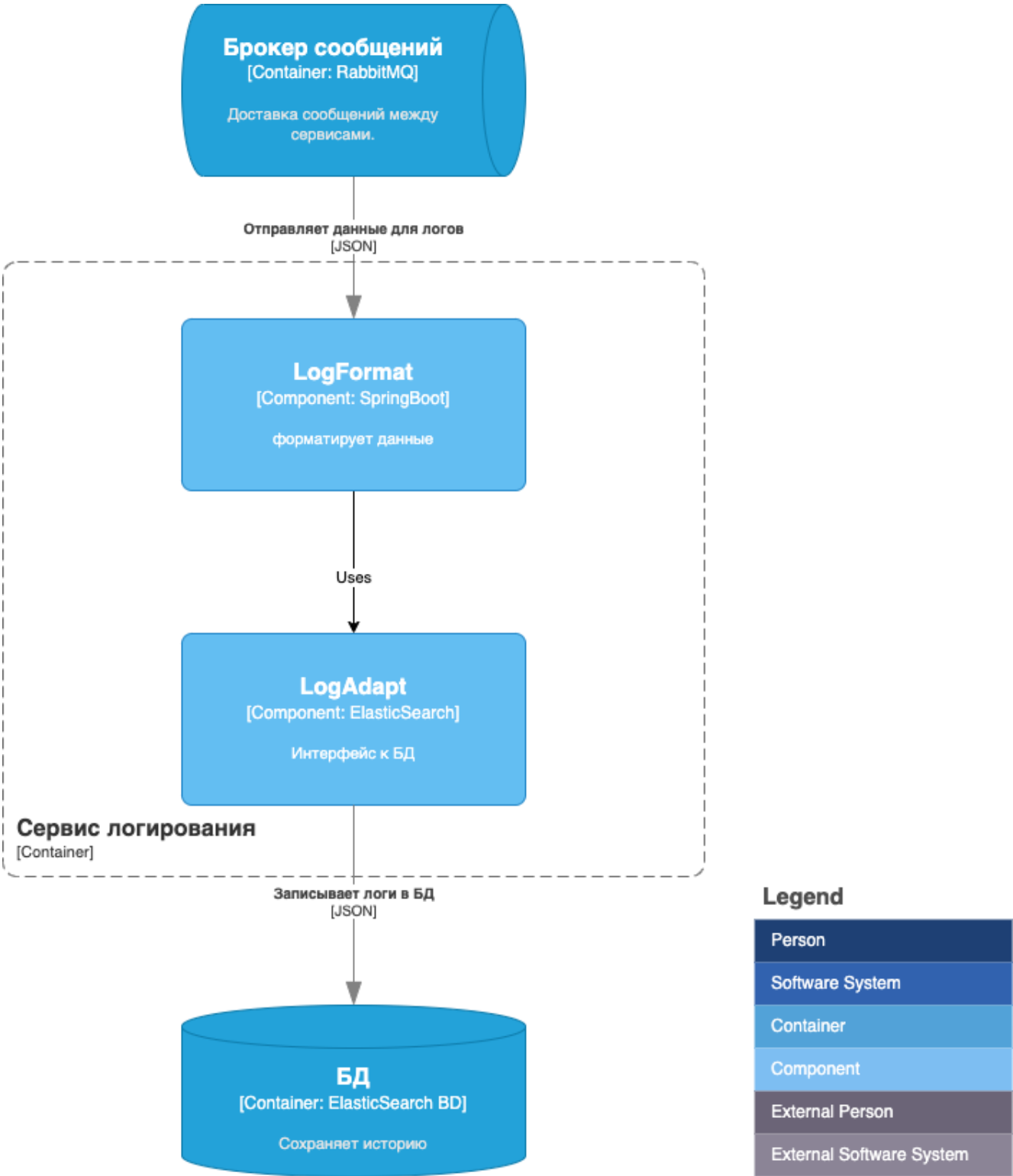
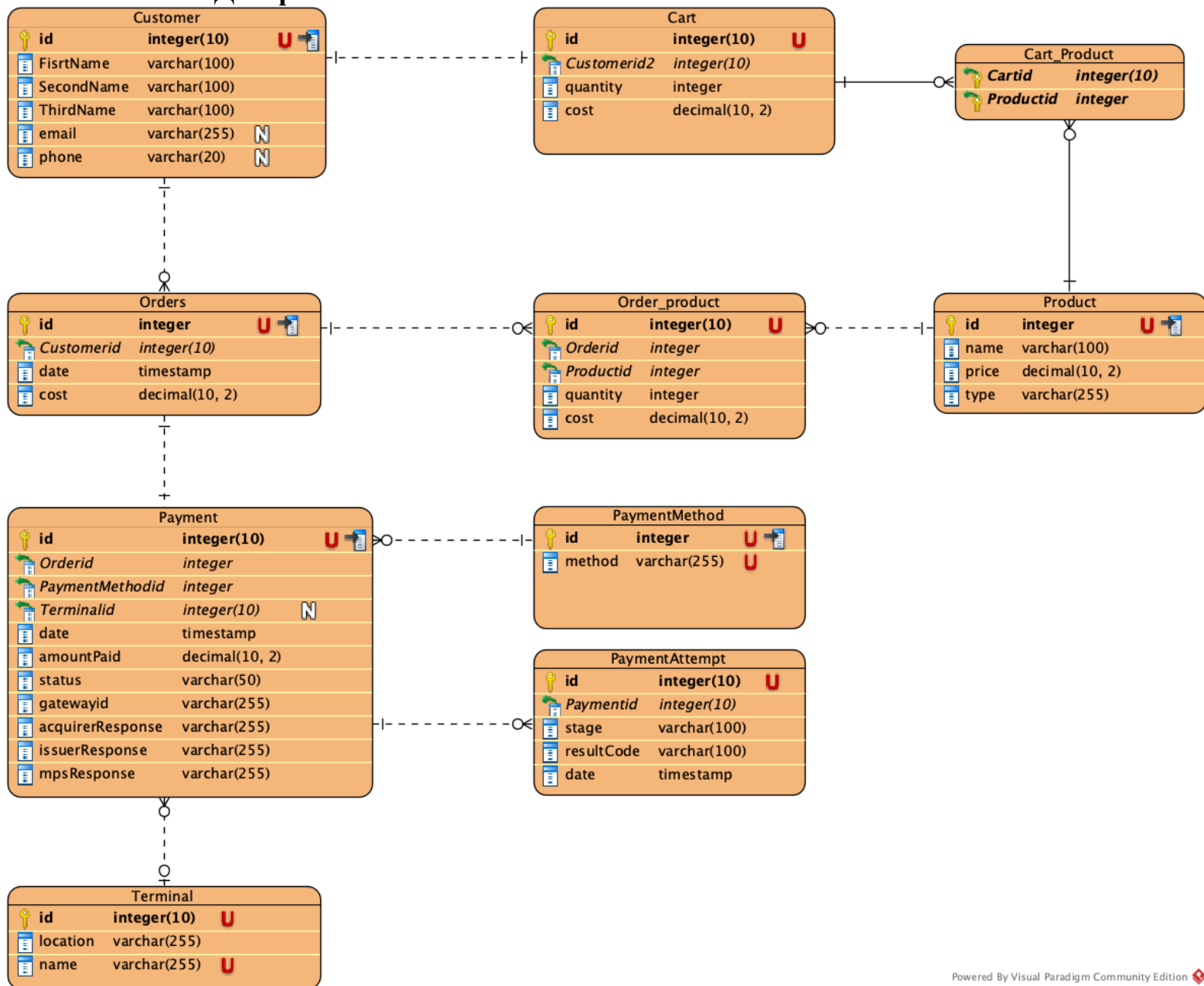


Диаграмма компонентов. Логирование



Приложение D. Модель данных ER Диаграмма



Powered By Visual Paradigm Community Edition

SQL код по созданию таблиц

-- Таблица Покупателей

```
CREATE TABLE Customer (
    id SERIAL PRIMARY KEY,
    FirstName VARCHAR(100) NOT NULL,
    SecondName VARCHAR(100) NOT NULL,
    ThirdName VARCHAR(100) NOT NULL,
    email VARCHAR(255),
    phone VARCHAR(20)
);
```

-- Таблица Товаров

```
CREATE TABLE Product (
    id SERIAL PRIMARY KEY,
```

```

    name VARCHAR(100) NOT NULL,
    price DECIMAL(10,2) NOT NULL,
    type VARCHAR(255) NOT NULL
);

-- Таблица Заказов
CREATE TABLE Orders (
    id SERIAL PRIMARY KEY,
    Customer_id INTEGER,
    date TIMESTAMP NOT NULL,
    cost DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (Customer_id) REFERENCES Customer(id)
);

-- Таблица товаров в заказе
CREATE TABLE Orders_product (
    id SERIAL PRIMARY KEY,
    Orders_id INTEGER,
    Product_id INTEGER,
    quantity INTEGER NOT NULL,
    cost DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (Orders_id) REFERENCES Orders(id),
    FOREIGN KEY (Product_id) REFERENCES Product(id)
);

-- Таблица Методов оплаты
CREATE TABLE PaymentMethod (
    id SERIAL PRIMARY KEY,
    method VARCHAR(255)
);

-- Таблица Терминалов
CREATE TABLE Terminal (
    id SERIAL PRIMARY KEY,
    location VARCHAR(255),
    device_id VARCHAR(100),
    software_version VARCHAR(50)
);

-- Таблица Платежей
CREATE TABLE Payment (
    id SERIAL PRIMARY KEY,
    Orders_id INTEGER UNIQUE,
    PaymentMethod_id INTEGER,
    Terminal_id INTEGER,
    amount DECIMAL(10,2) NOT NULL,
    status VARCHAR(50), -- initiated, success, failed
    gateway_transaction_id VARCHAR(100),
    acquirer_response_code VARCHAR(20),
    issuer_response_code VARCHAR(20),
    date TIMESTAMP NOT NULL,
    FOREIGN KEY (Orders_id) REFERENCES Orders(id),

```

```
FOREIGN KEY (PaymentMethod_id) REFERENCES PaymentMethod(id),  
FOREIGN KEY (Terminal_id) REFERENCES Terminal(id)  
);
```

-- Таблица Попыток оплаты

```
CREATE TABLE PaymentAttempt (  
  id SERIAL PRIMARY KEY,  
  payment_id INTEGER REFERENCES Payment(id),  
  attempt_number INTEGER,  
  stage VARCHAR(50), -- acquirer, mps, issuer  
  result_code VARCHAR(20),  
  result_message TEXT,  
  date TIMESTAMP NOT NULL  
);
```

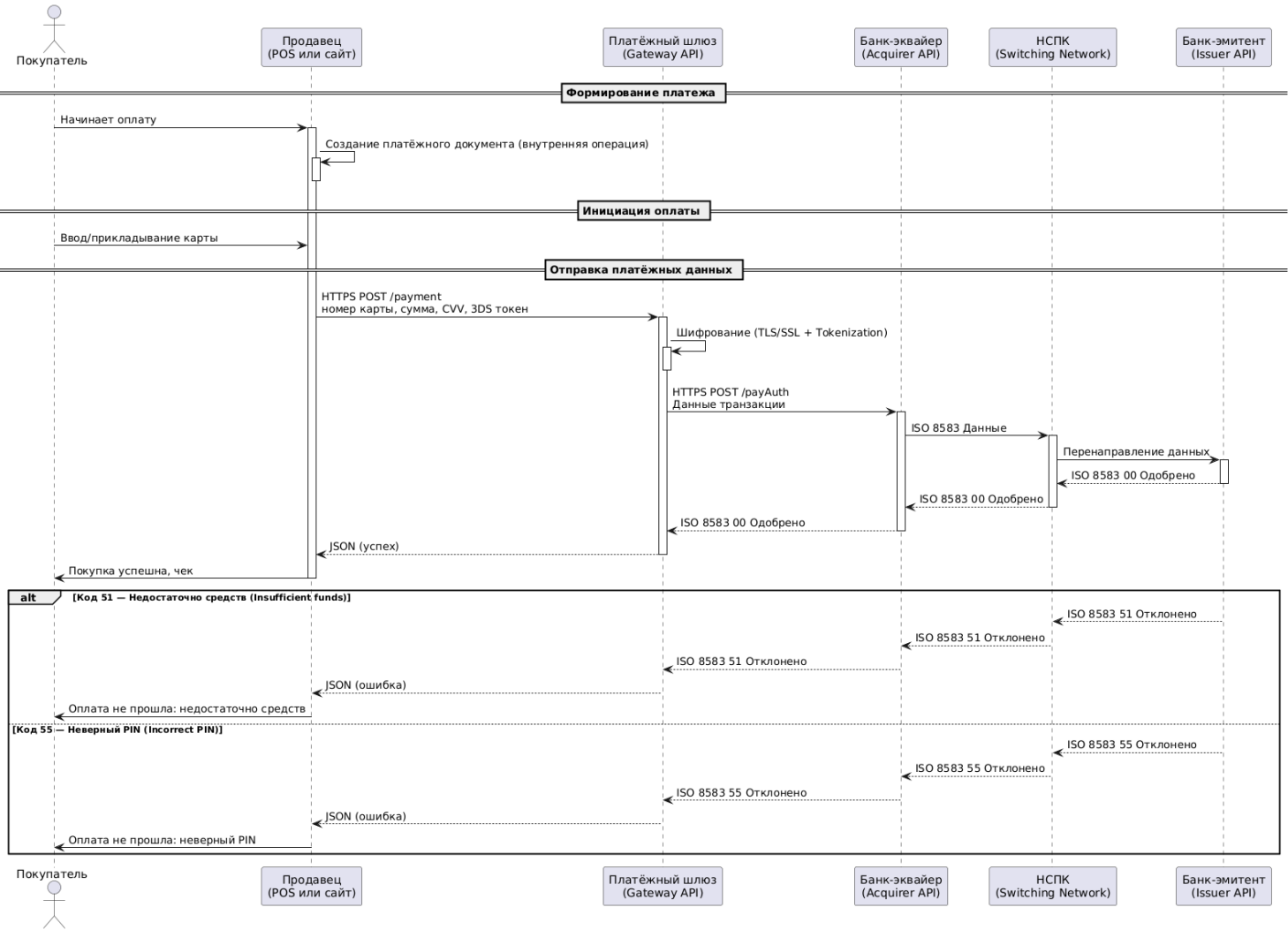
-- Таблица Корзин

```
CREATE TABLE Cart (  
  id SERIAL PRIMARY KEY,  
  Customer_id INTEGER UNIQUE,  
  quantity INTEGER NOT NULL,  
  cost DECIMAL(10,2) NOT NULL,  
  FOREIGN KEY (Customer_id) REFERENCES Customer(id)  
);
```

-- Таблица Товаров в корзине

```
CREATE TABLE Cart_Product (  
  Cart_id INTEGER,  
  Product_id INTEGER,  
  FOREIGN KEY (Cart_id) REFERENCES Cart(id),  
  FOREIGN KEY (Product_id) REFERENCES Product(id)  
);
```

Приложение Е. Диаграмма последовательности



Приложение F. BPMN

