Техническое задание

на разработку *Мобильного приложения для онлайн-бронирования услуг груминга для животных*

Версия 1.1

Оглавление

[1 История изменений 5](#_Toc196678303)

[2 Введение 6](#_Toc196678304)

[2.1 Назначение 6](#_Toc196678305)

[2.2 Соглашения, принятые в документе 6](#_Toc196678306)

[A. Нумерация: 6](#_Toc196678307)

[2.3 Границы проекта 7](#_Toc196678308)

[2.4 Ссылки 11](#_Toc196678309)

[3 Общее описание 11](#_Toc196678310)

[3.1 Общий взгляд на продукт 11](#_Toc196678311)

[3.2 Классы и характеристики пользователей 12](#_Toc196678312)

[3.3 Операционная среда 15](#_Toc196678313)

[3.4 Ограничения дизайна и реализации 17](#_Toc196678314)

[3.5 Предположения и зависимости 18](#_Toc196678315)

[4 Функции системы 19](#_Toc196678316)

[4.1 Функция BOOKING.CREATE «Создание заказа» 20](#_Toc196678317)

[4.1.1 Функциональные требования 20](#_Toc196678318)

[4.2 Функция NOTIFY «Уведомления» 24](#_Toc196678353)

[4.2.1 Функциональные требования 24](#_Toc196678354)

[4.3 Функция BOOKING.STATUS «Управление статусами заказа» 25](#_Toc196678355)

[4.3.1 Функциональные требования 25](#_Toc196678356)

[4.4 Функция PAYMENT «Оплата» 26](#_Toc196678357)

[4.4.1 Функциональные требования 26](#_Toc196678358)

[4.5 Функция USER.PROFILE «Личный кабинет владельца» 27](#_Toc196678359)

[4.5.1 Функциональные требования 28](#_Toc196678360)

[4.6 Функция ADMIN «Панель администратора» 28](#_Toc196678361)

[4.6.1 Функциональные требования 28](#_Toc196678362)

[4.7 Функция BOOKING.RESCHEDULE «Перенос заказа» 29](#_Toc196678363)

[4.7.1 Функциональные требования 30](#_Toc196678364)

[4.8 Функция BOOKING.CANCEL «Отмена заказа» 30](#_Toc196678365)

[*4.8.1* Описание 30](#_Toc196678366)

[4.8.2 Функциональные требования 30](#_Toc196678367)

[9.1 Приложение A. Словарь терминов 31](#_Toc196678368)

# ****История изменений****

| **Версия** | **Дата** | **Автор** | **Изменения** |
| --- | --- | --- | --- |
| **1.1** | **10.02** | **Бурлакова А.В** | **Внесены изменения в разделы 2.1, 2.2, 2.3, 3.2, 3.3, 3.4, 4.2, 4.3,4.4,4.5, 4.6, 9.1** |
| **1.2** | **13.04** | **Бурлакова А.В** | **Внесены изменения в разделы 4.2, 4.3,4.4,4.5, 4.6, 9.1** |
| **1.3** | **19.04** | **Бурлакова А.В** | **Внесены изменения в разделы 2.1, 2.2, 2.3, 3.2, 3.3, 3.4** |

# ****Введение****

## ****Назначение****

Документ описывает **мобильное приложение для онлайн-бронирования услуг груминга для домашних животных**. Приложение предусматривает систему статусов (создано, забронировано, выполнено, просрочено, отменено), онлайн-оплату, push-уведомления, регистрацию пользователей и администраторов, а также набор возможностей по управлению расписанием и заказами.

На основании объёма и полноты требований (управление расписанием, система уведомлений, статусы заказов, логика возвратов и т.д.) можно сделать вывод, что спецификация охватывает **всю систему**: от клиентской части (приложение для владельца животного) до административной (личный кабинет администратора, управление расписанием и заказами). То есть это не только часть системы, а именно **целиком функциональность** приложения в рамках первоначальной реализации (MVP).

Данную редакцию можно считать первым релизом, поскольку тут описаны основные (ключевые) функции работы с бронированием, а более детальные интеграции и улучшения обычно появляются в последующих версиях.

Документ не ограничивается только одним модулем (например, только уведомлениями или только оплатой), а описывает работу и связь множества компонентов (бронь, оповещения, статусы, оплаты, личные кабинеты владельце животных и администраторов). Поэтому требования относятся ко всему приложению: и к мобильной части для владельцев животных, и к административной части.

**Типы читателей:**

* Разработчики (реализуют бизнес‑логику, API, уведомления и механизмы работы с БД и учётными записями)
* Тестировщики (проверяют статусы заказов, уведомления и возвраты, этапы бронирования)
* Менеджеры проектов (планируют и приоритизируют требования, релизы, сроки и ресурсы)
* Маркетологи (формируют позиционирование, ключевые функции и стратегию продвижения)
* Аналитики (моделируют бизнес-процессы, формируют документацию и помогают в принятии решений на основе данных)

## ****Соглашения, принятые в документе****

### Нумерация:

* Каждый пункт должен иметь уникальный идентификатор.
* При нумерации заголовков используются арабские числа каждая часть нумерации отделялась точкой, отражая иерархическую структуру документа. Например:
  + 2.2 Соглашения, принятые в документе.

1. **Стиль текста и его значение**

* Заголовки разделов:  
  Используются крупный шрифт (14) и жирное начертание для обозначения разделов, например:
  + **"** **Соглашения, принятые в документе"**
* Заголовки подразделов

Используется шрифт (12) и жирное начертание для обозначения подразделов, например:

* **"** **Стиль текста и его значение"**

1. **Требования**

* Требования оформляются в разделе «X.Y.2 Функциональные требования» в виде таблицы из двух столбцов: «№» и «Требование». В столбце «№» указывается уникальный идентификатор вида <Код функции>.
* <Код действия>.<Порядковый номер> (например, PAYMENT.INITIALIZE.1), где код функции и код действия — латиницей в верхнем регистре, а порядковый номер — целое число, увеличивающееся при добавлении каждого нового требования.
* Тексты требований идут в соседней ячейке, выравниваются по левому краю. Идентификаторы требований и заголовок таблицы выделяются полужирным шрифтом, а при добавлении нового требования достаточно взять следующий свободный порядковый номер и добавить строку в конец таблицы.

## ****Границы проекта****

Программное обеспечение представляет собой мобильное приложение для онлайн-бронирования услуг груминга для домашних животных. Основное назначение продукта — упростить и автоматизировать процесс записи на услуги груминга, обеспечивая удобный и быстрый способ для владельцев животных записаться на обслуживание, а администраторам — эффективно управлять расписанием, статусами заказов и платежными операции.

Связь продукта:

* **С** **конечными пользователями**:  
  Мобильное приложение для онлайн-бронирования услуг груминга для домашних животных должно обладать удобным и понятным дизайном для выбора услуг, бронирования визитов с онлайн-оплатой и получения своевременных уведомлений, что улучшает комфорт пользователей, укрепляет их лояльность и привлекает новых клиентов, способствуя оптимизации бизнес-процессов.
* **С корпоративными целями:**  
  Приложение помогает автоматизировать бронирование заказов, снизить затраты и улучшить качество обслуживания. Это поможет увеличить клиентскую базу и рост доходов, а также улучшение положения салона на рынке, что соответствует корпоративным целям повышения эффективности и конкурентоспособности.
* **С стратегическими бизнес-целями:**  
  Приложение напрямую поддерживает стратегические бизнес-цели компании. Оно позволяет выйти на новые сегменты рынка благодаря современной платформе для онлайн-бронирования, привлекает новых клиентов и повышает лояльность постоянных пользователей. Автоматизация процесса записи снижает операционные затраты и ускоряет обслуживание, что помогает увеличить доходы. Кроме того, сбор и анализ пользовательских данных позволяет лучше понимать потребности клиентов и принимать обоснованные управленческие решения, способствуя развитию бизнеса и повышению конкурентоспособности компании.

**Высокоуровневая сводка главной функциональности выпуска**

* **Онлайн-бронирование услуг:**  
  Позволяет пользователям просматривать актуальное расписание грумеров, выбирать из полного перечня услуг (например, стрижка, мытьё, окрашивание и т.д.) с указанием цены и длительности исполнения, а также бронировать удобное время через мобильное приложение.
* **Система уведомлений**:  
  Автоматическая отправка push-уведомлений и email-сообщений. Владельцы животных получают напоминания за 24 часа до визита и уведомления об изменениях, а грумеры — сводку предстоящих заказов ежедневно.
* **Управление заказами:**Включает автоматическое и ручное изменение статусов заказа (таких как «создано», «забронировано», «выполнено», «просрочено», «отменено») в зависимости от действий пользователя или событий (например, неявка владельца). Администраторы могут управлять заказами через специальный интерфейс.
* **Платежная система:**  
  Поддерживает онлайн-оплату с возможностью внесения предоплаты (30% от общей суммы) или полной оплаты услуги. Автоматизированы процессы возврата средств в случае отмены или просрочки заказа согласно установленным правилам.
* **Личные кабинеты:**  
  Обеспечивают отдельный функционал для владельцев животных и администраторов. Владельцы могут управлять своими бронированиями, просматривать историю визитов и платежей. Для администраторов предусмотрена отдельная панель управления, где они могут настраивать расписание работы грумеров и отслеживать системные уведомления, а также выполнять другие задачи, связанные с контролем заказов и оперативным реагированием на возникающие проблемы, что обеспечивает высокое качество обслуживания*.*

**Цели проекта:**

* **Автоматизировать процесс онлайн‑бронирования услуг груминга**  
  Обеспечить владельцам домашних животных возможность быстро и удобно выбирать услугу, грумера и время визита через мобильное приложение.
* **Обеспечить кроссплатформенный интерфейс**Разработать адаптивный мобильный UI для iOS (iOS 13.0+) и Android (8.0+) с учётом принципов удобства и доступности. Ориентироваться на основные разрешения экранов: iOS: от 750×1334 (iPhone 8) до 1284×2778 (iPhone 14 Pro Max), Android: от 720×1280 до 1440×3200.
* **Реализовать систему управления расписанием и заказами**  
  Дать администраторам инструменты для планирования смен, изменения статусов («создано», «забронировано», «выполнено», «просрочено», «отменено») и оперативного реагирования на изменения.
* **Интегрировать безопасную онлайн‑оплату с поддержкой предоплаты и возвратов**  
  Подключить сертифицированные платёжные шлюзы, реализовать логику взимания 30% предоплаты, полного расчёта и автоматического возврата средств при отмене или просрочке.

Обеспечить безопасность платёжных данных через:

* использование протокола HTTPS для шифрования передаваемой информации,
* с соответствие требованиям Положения Банка России № 757-П (о защите информации при осуществлении переводов денежных средств),
* отсутствие хранения реквизитов карт в системе приложения (вся работа — через платёжного провайдера с шифрованием).
* Рекомендуемые платёжные шлюзы для России: ЮKassa (Яндекс.Касса), Тинькофф Касса, Сбербанк, CloudPayments, Robokassa.
* Внедрить систему уведомлений для всех участников процесса  
  Настроить push‑уведомления и email‑рассылки: напоминания владельцам за 24 часа до визита, ежедневные сводки для грумеров, уведомления об изменениях и отказах.
* **Внедрить систему уведомлений для всех участников процесса**Настроить push‑уведомления и email‑рассылки: напоминания владельцам за 24 часа до визита, ежедневные сводки для грумеров, уведомления об изменениях и отказах.
* **Собрать инструменты аналитики и отчётности**Интегрировать систему метрик (Yandex Metrica, Google Analytics) для мониторинга конверсии бронирований, лояльности клиентов и загрузки грумеров.
* **Обеспечить стандарты безопасности и соответствие ФЗ‑152**Гарантировать защиту данных пользователей (TLS 1.2), хранение персональных данных в соответствии с законодательством РФ.

## Ссылки

*Пока отсутствуют.*

# ****Общее описание****

Приложение для онлайн-бронирования услуг груминга для домашних животных предназначено для использования на мобильных устройствах (iOS, Android) и интегрируется с внешними системами (например, платежными шлюзами и уведомлениями). Ожидается, что его основными пользователями станут владельцы животных, использующие приложение для записи на услуги, а также администраторы, ответственные за управление расписанием и контроль заказов. Также учитываются известные ограничения (например, зависимость от стабильного интернет-соединения и аппаратных характеристик устройств), а также делаются предположения о доступности сторонних сервисов и технологий, необходимых для функционирования продукта.

## **Общий** взгляд на продукт

Данный продукт представляет собой совершенно новое мобильное приложение для онлайн-бронирования услуг груминга для домашних животных. Оно было инициировано в рамках стратегии цифровой трансформации компании, направленной на улучшение взаимодействия с клиентами и повышение эффективности внутренних процессов. Основной задачей приложения является упрощение процесса выбора, записи и оплаты услуг груминга, что способствует повышению удовлетворённости клиентов и укреплению их лояльности.

Приложение позиционируется как самостоятельное решение, являющееся новым членом растущего семейства цифровых продуктов компании. Оно не является обновлением существующей системы или заменой предыдущего приложения, а представляет собой разработку с нуля. Новый продукт задуман как ключевой элемент комплексной цифровой экосистемы, которая включает в себя и другие сервисы компании, направленные на повышение качества обслуживания клиентов.

Продукт тесно интегрируется с несколькими внешними системами.

* Во-первых, он использует API платёжных систем, что обеспечивает безопасную и надёжную обработку онлайн-оплат — как полной, так и в виде предоплаты.
* Во-вторых, приложение связывается с системами уведомлений: приложение интегрируется с сервисами отправки уведомлений, такими как FCM для push-уведомлений и SMTP-серверы или специализированные сервисы для email-рассылок (например, SendGrid), чтобы эффективно доставлять сообщения пользователям.
* В-третьих, для администраторов разработана отдельная панель управления, позволяющая настраивать расписание работы грумеров, отслеживать поступающие заявки и системные уведомления, а также управлять другими аспектами операционной деятельности. В приложении будет реализовано разграничение прав, чтобы добавить персональные функции для администраторов.

## ****Классы и характеристики пользователей****

В данном приложении для онлайн-бронирования услуг груминга для домашних животных предусмотрены следующие классы пользователей, каждый из которых имеет уникальные характеристики и права доступа. Эти классы позволяют разделить пользователей в зависимости от их ролей и уровня доступа к функционалу системы.

|  |  |  |  |
| --- | --- | --- | --- |
| Пользователь | Характеристика | Особенности | Ограничения |
| Владельцы животного | Это основной класс пользователей приложения, который использует сервис для бронирования услуг груминга для своих домашних животных. | Могут:  1. Просматривать доступные услуги (стрижка, мытьё, окрашивание и т.д.).  2. Выбирать подходящее время для записи.  3. Оформлять и оплачивать заказ (в том числе с возможностью предоплаты).  4. Получать push- и email-уведомления о статусе записи, изменениях в расписании или напоминания о визите.  5. Переносить или отменять запись в пределах установленных сроков.  6. Управлять личными данными и настройками уведомлений через личный кабинет. | 1. Пользователи этого класса не имеют доступа к администрированию системы или управления расписанием грумеров. |
| Грумеры | сотрудники, предоставляющие услуги груминга. Они могут работать с приложением через отдельный интерфейс, предназначенный для управления своими записями и графиком работы. | Могут:  1. Просматривать свой график работы.  2. Уведомляться о предстоящих записях через уведомления.  3. Обновлять информацию о выполнении услуги (например, пометить запись как «выполнено»).  4. Получать уведомления о просроченных заказах или изменениях в расписании.  5. Вносить изменения только в своё расписание. | Грумеры не могут изменять статус заказов .  Их доступ ограничен только работой с собственными записями и выполнениям услуг. |
| Администраторы | имеют доступ ко всей административной панели системы. Это привилегированный класс пользователей, который управляет расписанием грумеров, отслеживает заказы и взаимодействует с клиентами. | Могут:  1. Управлять расписанием грумеров, распределяя их рабочие смены.  2. Просматривать, изменять и отменять заказы, включая изменение статуса заказа на «выполнено», «просрочено» и т.д.  3. Отслеживать ошибки системы и уведомления о сбоях.  4. Фильтровать заказы по различным параметрам (например, по дате, грумеру, статусу заказа).  5. Просматривать персонализированную информацию о пользователях и заказах для дальнейшего анализа.  6. Иметь доступ ко всем функциям управления сервисом и могут иметь доступ ко всем функциям управления сервисом в рамках административной панели: управлять расписанием, заказами, пользователями и отслеживать процесс бронирования и оказания услуг. | 1.Ограничен доступ к персональным данным клиентов.  2. Нет доступа к техническим настройкам:  конфигурации серверов и баз данных  программному код, логике работы приложения  внутренним настройкам системы (например, включение/выключение функций, обновление версий, управление лицензиями)  процессам резервного копирования и восстановления данных.  параметрам безопасности (например, настройку шифрования, протоколы авторизации) |
| Техническая поддержка | отвечают за поддержку и обслуживание системы, исправление багов и обеспечение стабильной работы инфраструктуры. | Могут:  1. Иметь доступ к внутренним логам системы для диагностики и решения проблем.  2. Работать с базой данных, проводят обновления и исправления.  3. Участвовать в обновлении версий системы, устанавливать патчи безопасности и контролировать стабильность работы серверов и сопутствующих сервисов.  4. Проводить профилактические работы: мониторинг производительности и устранение ошибок. | 1.Не взаимодействуют напрямую с заказами или пользователями.  2. Не имеют права изменять бизнес-логику системы, процесс бронирования, алгоритмы работы сервисов и интерфейсы приложения.  3. Не изменяют структуру базы данных: нельзя создавать, удалять или изменять таблицы, поля, связи между таблицами. |

Привилегированные пользователи:

* Администраторы — это основной привилегированный класс пользователей, который имеет полный доступ ко всем аспектам работы системы, включая управление заказами, настройку расписания и администрирование пользователей.
* Технические специалисты — имеют высокий уровень доступа, связанный с обеспечением технической поддержки и стабилизации работы системы, но не участвуют в бизнес-процессах напрямую.

## ****Операционная среда****

Аппаратная платформа и устройства пользователей:  
Приложение будет функционировать на мобильных устройствах с поддержкой следующих характеристик:

* Смартфоны и планшеты с поддержкой Android (от версии Android 8.0 и выше).
* Устройства Apple под управлением iOS 13.0 и выше.
* Минимальные системные требования: 2 ГБ ОЗУ, стабильное интернет-соединение (Wi-Fi / LTE / 4G).

Операционные системы:

* Клиентская часть (мобильное приложение):
  + Android (8.0+), iOS (13.0+)

Географическое размещение пользователей:

* Приложение рассчитано на использование в пределах России и, потенциально, в других русскоязычных странах СНГ.
* Серверная инфраструктура:
  + Серверы и базы данных планируется развернуть в облачной инфраструктуре (например, Yandex Cloud, Selectel или VK Cloud) с центрами обработки данных на территории РФ.
  + Это обеспечивает соблюдение законодательства о хранении и обработке персональных данных (ФЗ-152).

Базы данных:

* Основная СУБД — PostgreSQL или MySQL (по результатам проектирования).
* Размещение в защищённой облачной среде (с резервным копированием и системой мониторинга).
* Доступ к базе ограничен только авторизованными сервисами.

Совместимость с другими компонентами: Приложение и административная система должны быть совместимы со следующими внешними и внутренними компонентами:

* Платёжные системы: ЮKassa, Сбербанк Онлайн, Tinkoff Pay, Google Pay.
* Сервисы уведомлений:
  + Push: Firebase Cloud Messaging (FCM)
  + Email: SendGrid, SMTP-сервер почтового провайдера, возможна интеграция с Яндекс 360 или Mailgun.
* Интеграция с внутренними CRM и системами отчётности (при необходимости).

Требования к сети и подключению:

* Все соединения между клиентом и сервером должны быть защищены через TLS 1.2+.
* API-сервер должен обеспечивать устойчивую работу при 10 000 до 20 000 запросах в сутки.
* Система должна выдерживать одновременную работу до 1000 активных пользователей без деградации производительности при:
* авторизации
* просмотре списка услуг и расписания
* оформлении или отмене бронирования
* получении уведомлений о статусе заказов

Языки программирования:

* Клиентская часть (мобильное приложение): Dart (с использованием Flutter).

Серверная часть (backend): Python (например, фреймворк FastAPI или Django)

## ****Ограничения дизайна и реализации****

|  |  |  |  |
| --- | --- | --- | --- |
| **Технологический стек** | **Требования к перечню возможных применяемых технологий** | **Обязательность** | **Примечания по отклонению** |
| Язык программирования и фреймворки | Flutter, React (кроссплатформенные фреймворки) | обязательное требование | использование Swift/Kotlin возможно только для отдельных платформ с обоснованием |
| Выбор СУБД | PostgreSQL, MySQL | обязательное требование | исключается SQLite, MongoDB — недостаточная производительность |
| Безопасность данных | TLS, проверенные криптобиблиотеки, соответствие ФЗ-152: OpenSSL, LibreSSL, PyCA Cryptography. | обязательное требование | запрещено использование неподтверждённых или устаревших решений |
| Интеграция платёжных систем | API ЮKassa, Tinkoff Pay, Сбербанк Онлайн | обязательное требование | запрещено использование сторонних агрегаторов без сертификации |
| Пользовательский интерфейс | Адаптивный дизайн, поддержка разных экранов | обязательное требование | фиксированные размеры и неадаптивные решения не допускаются |
| Географические ограничения | Хранение данных на территории РФ (Yandex Cloud, Selectel) | обязательное требование | запрещено использование облаков без дата-центров в РФ (AWS, Google Cloud и др.) |
| Поддержка устаревших ОС | Совместимость минимум с Android 8.0 и iOS 13.0 | обязательное требование | нельзя использовать библиотеки без поддержки указанных версий ОС |

## ****Предположения и зависимости****

*Предположения, касающиеся функциональности и разработки системы:*

|  |  |  |
| --- | --- | --- |
| **Категория** | **Предположение** | **Потенциальный риск** |
| **Поддержка платформ** | Пользователи используют устройства с Android 8.0+ и iOS 13.0+ | Использование более старых устройств приведёт к ошибкам или невозможности установить приложение |
| **Платёжные системы** | Интеграция с ЮKassa, Сбербанк Онлайн и Tinkoff Pay будет доступна и стабильна | Изменения API или условий могут вызвать сбои в транзакциях и потребность в переработке интеграции |
| **Сервисы уведомлений** | Firebase Cloud Messaging и SendGrid будут стабильно функционировать и поддерживаться | Сбои, изменения условий или прекращение поддержки повлияют на доставку уведомлений |
| **Серверные технологии** | PostgreSQL/MySQL обеспечат нужную производительность | Возможные проблемы с масштабируемостью и нагрузкой потребуют переработки архитектуры |
| **Законодательные требования** | Приложение соответствует ФЗ-152 и требованиям хранения данных на территории РФ | Изменения в законодательстве могут потребовать изменений в архитектуре обработки и хранения данных |
| **Интернет-соединение** | У пользователей есть стабильный доступ к интернету | При медленном соединении работа приложения замедляется или становится нестабильной |

*Зависимости от внешних факторов или компонентов:*

|  |  |  |
| --- | --- | --- |
| **Тип зависимости** | **Описание зависимости** | **Риск при нарушении** |
| **Облачные провайдеры** | Использование Yandex Cloud, Selectel для хранения и обработки данных | Технические сбои, рост цен или смена условий могут повлиять на доступность сервиса |
| **API внешних сервисов** | Использование внешних API (платежи, уведомления, аналитика и др.) | Изменения в API или прекращение поддержки приведут к необходимости переработки логики приложения |
| **Сторонние библиотеки и фреймворки** | Использование Flutter, React Native, PostgreSQL и др. | Обновления или смена лицензии могут потребовать адаптацию или переход на другие инструменты |
| **Устройства пользователей** | Устройства должны соответствовать минимальным требованиям ОС | Устаревшие устройства могут вызвать проблемы совместимости и нестабильную работу |
| **Интернет-соединение** | Нужна постоянная связь с сервером и сервисами | Нестабильное соединение приводит к задержкам, ошибкам и невозможности выполнять операции в реальном времени |

# ****Функции системы****

Перечень функций, которые должна поддерживать система:

* BOOKING – Работа с заказами
* BOOKING.CREATE – Создание заказа
* BOOKING.STATUS – Управление статусами заказа
* BOOKING.VIEW – Просмотр деталей заказа
* BOOKING.CANCEL – Отмена заказа
* BOOKING.RESCHEDULE – Перенос заказа
* NOTIFY – Уведомления
* NOTIFY.OWNER\_PUSH – Push уведомление владельцу
* NOTIFY.OWNER\_EMAIL – Email уведомление владельцу
* NOTIFY.GROOMER\_EMAIL – Email уведомление грумеру
* NOTIFY.RETRY – Повторная отправка уведомлений
* NOTIFY.LOG – Журнал доставки уведомлений
* PAYMENT – Оплата
* PAYMENT.INITIALIZE – Инициация оплаты
* PAYMENT.CONFIRM – Подтверждение оплаты
* PAYMENT.REFUND – Возврат средств
* PAYMENT.FAILURE – Обработка ошибок платежей
* USER – Личный кабинет владельца
* USER.REG – Регистрация и восстановление доступа
* USER.PROFILE – Просмотр/редактирование профиля
* USER.CABINET – История заказов и платежей
* USER.SERVICES – Просмотр и выбор услуг
* USER.SCHEDULE – Просмотр расписания грумеров
* ADMIN – Панель администратора
* ADMIN.ORDERS – Управление всеми заказами
* ADMIN.SCHEDULE – Управление расписанием грумеров
* ADMIN.ERRORS – Мониторинг ошибок (платежи, сети, уведомления)
* ADMIN.USERS – Управление аккаунтами
* ADMIN.SERVICES – Управление перечнем услуг
* ADMIN.PRICING – Настройка цен и акций
* ADMIN.NOTIFICATIONS – Просмотр логов уведомлений
* ADMIN.PERM – Управление ролями и правами доступа
* SECURITY – Безопасность
* SECURITY.TLS – Шифрование трафика
* SECURITY.AUTH – Контроль прав доступа

## Функция BOOKING.CREATE «Создание заказа»

Владелец животного выбирает услугу, грумера, дату и время визита, при необходимости добавляет дополнительные опции и комментарии. Перед подтверждением заказа система показывает итоговую стоимость и запрашивает согласие с условиями. После подтверждения заказ получает уникальный номер, а выбранное время закрепляется за владельцем.

**Приоритет:** Высокий

### Функциональные требования

|  |  |
| --- | --- |
| № | Требование |
| BOOKING.CREATE.1 | Система должна отображать полный каталог услуг: стрижка шерсти, мытьё животных, вычёсывание подшерстка, тримминг, стрижка когтей, шлифовка когтей, чистка ушей, чистка зубов, гигиеническая стрижка, обработка параанальных желез, выведение колтунов, укладка шерсти, декоративное окрашивание, SPA-процедуры, антипаразитарная обработка, ароматизация шерсти, подготовка к выставке, индивидуальный подбор косметики и консультация по уходу. |
| BOOKING.CREATE.2 | Система должна показывать список грумеров по выбранной услуге с рейтингом, опытом и описанием. |
| BOOKING.CREATE.3 | Система должна выводить доступные временные слоты выбранного грумера, учитывая длительность услуги на странице оформления заказа. |
| **BOOKING.CREATE.4** | Система должна предотвращать двойное бронирование, запрещая выбор слота, уже занятого в другом заказе того же грумера. |
| **BOOKING.CREATE.5** | Перед подтверждением заказа система должна показывать детальную смету: итоговая цена, перечень выбранных услуг, размер предоплаты, ориентировочная длительность услуг, способ оплаты. |
| **BOOKING.CREATE.6** | Система должна позволять владельцу оставить комментарий к заказу (особые пожелания, описание поведения или медицинских ограничений питомца). |
| **BOOKING.CREATE.7** | При попытке оформить заказ без согласия система должна открывать модальное окно с полным текстом условий. |
| **BOOKING.CREATE.8** | После создания заказа система должна присваивать уникальный номер заказу. |
| **BOOKING.CREATE.9** | Система должна давать возможность выбрать дополнительные опции (например, стрижка шерсти носа, удаление колтунов) прямо при оформлении заказа. |
| **BOOKING.CREATE.10** | Система должна предоставлять владельцу животного возможность просматривать доступное расписание грумеров с фильтрацией по дате, времени и типу услуги. |
| **BOOKING.CREATE.11** | Система должна давать владельцу животного добавлять выбранные услуги в корзину. |
| **BOOKING.CREATE.12** | Система должна предотвращать повторное добавление одной и той же услуги. |
| **BOOKING.CREATE.13** | Система должна предоставлять пользователю доступ к содержимому корзины. |
| **BOOKING.CREATE.14** | Система должна автоматически пересчитывать итоговую сумму при каждом изменении содержимого в корзине. |
| **BOOKING.CREATE.15** | Система должна автоматически очищать корзину после успешного оформления заказа. |
| **BOOKING.CREATE.16** | Система должна сохранять состояние корзины локально на устройстве, чтобы восстановить его при следующем запуске. |
| **BOOKING.CREATE.17** | Система должна информировать пользователя, если выбранные услуги не могут быть оказаны по следующим причинам:   * Технический сбой или ошибка при бронировании * Отсутствие нужных материалов или средств (например, краски для окрашивания, специальных шампуней) * Ограничения по породе, размеру или состоянию животного (например, особые медицинские противопоказания) * Несоответствие времени оказания услуги внутренним регламентам салона (например, услуга требует больше времени, чем доступно в смену) * Закрытие салона в выбранную дату (праздничные дни, санитарная обработка и пр.) * Превышение лимита бронирований на конкретный день * Невозможность оказания комплекса услуг вместе (Декоративное окрашивание и антипаразитарная обработка, SPA-процедуры и тримминг, антипаразитарная обработка и чистка зубов, глубокое мытьё лечебным шампунем и декоративное окрашивание, выведение колтунов и декоративная стрижка, ароматизация шерсти и SPA-уход.) * Отсутствие свободных временных слотов в расписании грумеров * Недоступность выбранного грумера (например, отпуск, болезнь) |
| **BOOKING.CREATE.18** | Система должна предоставлять владельцу животного возможность выбора даты и времени оказания услуги на странице оформления заказа. |
| **BOOKING.CREATE.19** | Система должна блокировать недоступные даты и время. |
| **BOOKING.CREATE.20** | Система должна автоматически обновлять список доступных временных слотов при изменении выбранной услуги или грумера. |
| **BOOKING.CREATE.21** | Система должна холдировать выбранный владельцем животного временной слот на 30 минут после его резервирования (переход заказа в статус **«создано»).** |
| **BOOKING.CREATE.22** | Система должна автоматически снимать резервирование при отмене заказа |
| **BOOKING.CREATE.22** | Система должна закреплять дату и время за заказом владельца животного при переходе заказа в статус «забронировано». |

## Функция NOTIFY «Уведомления»

Автоматическая отправка уведомлений владельцу(по событиям) и грумеру (по расписанию).

**Приоритет:** Высокий

### Функциональные требования

|  |  |
| --- | --- |
| № | Описание |
| NOTIFY.OWNER\_PUSH.1 | Система должна отправлять владельцу push‑уведомление о успешной оплате заказа. |
| NOTIFY.OWNER\_EMAIL.2 | Система должна отправлять владельцу push‑уведомление при отмене заказа с указанием причины и информации о возврате средств. |
| NOTIFY.PAYMENT\_REMIND.1 | Система должна отправлять владельцу push‑уведомление за 10 мин до истечения 30‑минутного таймера оплаты, если оплата ещё не поступила. |
| NOTIFY.OWNER\_PUSH.3 | Система должна отправлять владельцу push‑уведомление за 24 ч до начала услуги. |
| NOTIFY.OWNER\_PUSH.4 | При автоматическом переходе заказа в статус «просрочено» система должна отправлять владельцу push‑уведомление о пропуске записи. |
| NOTIFY.GROOMER\_EMAIL.1 | Система должна отправлять грумеру email‑уведомление со списком записей на следующий день каждый будний день в 20:00 МСК. |
| NOTIFY.GROOMER\_EMAIL.2 | В случае сбоя рассылки email грумеру система должна перед каждой попыткой система проверять актуальность уведомления. |
| NOTIFY.GROOMER\_EMAIL.3 | Система должна повторять отправку email- уведомления каждые 5 минут, не более 12 попыток подряд |
| NOTIFY.GROOMER\_EMAIL.4 | При устаревании события уведомление должно быть отменено без отправки. |
| NOTIFY.RETRY.1 | В случае недоставки push-уведомления система должна повторять попытку через 10 с (не более 3 раз). |
| NOTIFY.UNSUBSCRIBE.1 | Система должна предоставлять пользователю возможность отписаться от push-уведомлений через настройки. |
| NOTIFY.LOG.2 | Система должна считать push‑уведомление доставленным при подтверждении от ОС и отображении в журнале. |

## Функция BOOKING.STATUS «Управление статусами заказа»

Обеспечивает автоматические и ручные переходы между статусами заказа.

**Приоритет:** Высокий

### Функциональные требования

|  |  |
| --- | --- |
| **№** | **Требование** |
| BOOKING.STATUS.1 | Система должнаменять статус на **«забронировано»** сразу после получения успешного подтверждения оплаты. |
| BOOKING.STATUS.2 | **Изменение статуса заказа на "выполнено" осуществляется вручную администратором при подтверждении факта оказания услуги.** |
| BOOKING.STATUS.3 | Система должнаавтоматически переводить в **«просрочено»**, если клиент не пришёл в течение 10 мин после времени начала услуги. |
| BOOKING.STATUS.4 | Система должнаавтоматически отменять заказ, если предоплата не поступила в течение 30 мин после создания. |
| BOOKING.STATUS.6 | Система должнаобеспечивать корректный переход между статусами: «создано», «забронировано», «выполнено», «просрочено», «отменено». |
| **BOOKING.STATUS.7** | Администратор должен иметь возможность отката следующих статусов заказа на предыдущий:   * Из "выполнено" → обратно в "забронировано" или "в процессе" (если ошибка при завершении услуги). * Из "отменено" → обратно в "забронировано" (если отмена была по ошибке). * Из "просрочено" → обратно в "забронировано" (если клиент всё же пришёл с опозданием). |
| BOOKING. STATUS.8 | Система должна при нажатии «Забронировать услугу» создавать заказ со статусом **«создано».** |

## Функция PAYMENT «Оплата»

Управляет платёжными сценариями, обработкой ошибок и автоматическими возвратами.

**Приоритет:** Высокий

### Функциональные требования

|  |  |
| --- | --- |
| № | Требование |
| PAYMENT.INITIALIZE.1 | При создании заказа система должнаинициировать платеж в стороннем шлюзе. |
| PAYMENT.CONFIRM.1 | При получении ответа от платёжной системы (успех или ошибка) система должна зафиксировать результат в базе данных: сумму, тип оплаты (предоплата или полная оплата), статус оплаты, дату, время и идентификатор транзакции. |
| PAYMENT.REFUND.1 | Если заказ был автоматически отменён из-за несвоевременной оплаты, система должна вернуть пользователю 100% ранее внесённых средств (то есть если пользователь, например, внёс предоплату, но не оплатил заказ вовремя полностью — заказ отменяется, и вся внесённая сумма возвращается ему автоматически). |
| PAYMENT.REFUND.2 | При статусе «просрочено» и полной оплате система должна возвращать 50 % стоимости всего заказа. |
| PAYMENT.REFUND.3 | Система должназапретить возврат средств, если заказ перешёл в «просрочено» и была внесена только предоплата. |
| PAYMENT.FAILURE.1 | Система должнапри ошибке платёжного шлюза информировать пользователя и предложить повторную попытку. |
| PAYMENT.INITIALIZE.2 | Система должна запускать таймер 30 минут на оплатупосле перехода заказа в статус **«создано».** |
| PAYMENT.INITIALIZE.3 | Система должна предлагать выбор: внести предоплату (30 %) или полную оплату. |

## Функция USER.PROFILE «Личный кабинет владельца»

Обеспечивает владельцу доступ к управлению своими данными, заказами и настройкам.

**Приоритет:** Средний

### Функциональные требования

|  |  |
| --- | --- |
| **№** | **Требование** |
| USER.PROFILE.1 | Система должна осуществлять регистрацию с обязательным подтверждением либо email, либо номера телефона на выбор пользователя. |
| USER.PROFILE.2 | Система должна давать владельцу животного право редактировать контактные данные через личный кабинет: имя, номер телефона, email, город и комментарии или примечания для связи (например, "звонить только в рабочее время") |
| USER. PROFILE.3 | Система должна давать владельцу животного право просматривать историю заказов с фильтрацией по статусам. |
| |  | | --- | | USER.PROFILE.4 | | В личном кабинете система должна отображать персонализированные push‑ по предстоящим и прошедшим бронированиям. |
| **USER.PROFILE.5** | Система должна позволять владельцу менять пароль с подтверждением текущего пароля. |
| **USER.PROFILE.6** | Система должна поддерживать двухфакторную аутентификацию с использованием SMS. |

## Функция ADMIN «Панель администратора»

Позволяет администратору управлять всеми аспектами системы..

**Приоритет:** Средний

### Функциональные требования

|  |  |
| --- | --- |
| № | Требование |
| ADMIN.ORDERS.1 | Система должна давать администратору право просматривать заказы по дате, грумеру, статусу и другим параметрам. |
| ADMIN.ORDERS.2 | Система должна давать администратору право фильтровать заказы по дате, грумеру, статусу и другим параметрам. |
| ADMIN.SCHEDULE.1 | Система должна давать администратору право управлять расписанием грумеров и закрывать/открывать слоты. |
| ADMIN.USERS.1 | Система должна давать администратору право управлять аккаунтами владельцев и грумеров (активация, блокировка). |
| ADMIN.PRICING.1 | Система должна давать администратору право настраивать цены на услуги, скидки и акции. |
| ADMIN.ORDERS.3 | Система должна предоставлять администратору доступ ко всем заказам с возможностью просмотра и фильтрации по статусам («создано», «забронировано», «выполнено», «просрочено», «отменено»). При этом в основном списке бронирований отображаются только актуальные статусы заказов (например, «забронировано»), а не все заказы подряд. |
| ADMIN.ORDERS4 | Система должна давать администратору право менять статус забронированного заказа на “выполнено” в списке забронированных заказов при явке владельца животного в своё забронированное время. |

## Функция BOOKING.RESCHEDULE «Перенос заказа»

Владелец животного может перенести забронированную услугу на другое доступное время не позднее чем за 72 ч до визита.

**Приоритет:** Средний

### Функциональные требования

|  |  |
| --- | --- |
| № | Требование |
| BOOKING.RESCHEDULE.1 | Система должна позволять владельцу перенести заказ не позднее чем за 72 ч до назначенного времени. |
| BOOKING.RESCHEDULE.2 | После переноса автоматически система должна уведомлять владельца и грумера о новом времени. |
| BOOKING.RESCHEDULE.3 | Система должна обновлять слот в расписании грумера: освобождать старый и резервировать новый. |
| BOOKING.RESCHEDULE.4 | Если попытка переноса позже, чем за 72 ч — система должна показывать владельцу сообщение об ошибке и не разрешать операцию. |

## Функция BOOKING.CANCEL «Отмена заказа»

### Описание

Владелец или система могут отменить заказ автоматически или по запросу, при этом время возвращается в пул доступных слотов.

**Приоритет:** Высокий

### Функциональные требования

|  |  |
| --- | --- |
| **№** | **Требование** |
| BOOKING.CANCEL.1 | Система должна позволять владельцу отменять заказ не позднее чем за 72 ч до визита. |
| BOOKING.CANCEL.2 | Система должна автоматически отменять заказ, если предоплата (30 %) не внесена в течение 30 мин после создания. |
| BOOKING.CANCEL.3 | При отмене система должна освобождать забронированный слот и обновлять расписание грумера. |
| BOOKING.CANCEL.4 | После отмены система должна отправлять владельцу push‑уведомление с причинной отмены. |
| BOOKING.CANCEL.5 | Система должна автоматически инициировать возврат средств согласно правилам в разделе «Оплата» (см. PAYMENT.REFUND.\*). |

## ****9.1 Приложение A. Словарь терминов****

*FCM (Firebase Cloud Messaging)* — облачный сервис от Google для доставки push‑уведомлений на мобильные устройства.

*SMTP (Simple Mail Transfer Protocol)* — стандартный протокол передачи электронной почты между почтовыми серверами.

*CRM (Customer Relationship Management)* — система управления взаимоотношениями с клиентами, используемая для хранения истории заказов, контактов и аналитики по работе с пользователями.

*TLS (Transport Layer Security)* — протокол защиты сетевых соединений, обеспечивающий шифрование и целостность передаваемых данных.

*JWT (JSON Web Token)* — компактный URL‑безопасный формат для передачи заявлений (claims) между двумя сторонами, используемый для аутентификации и передачи информации о пользователе в виде зашифрованного токена.

*OAuth2 (Open Authorization 2.0)* — протокол авторизации, позволяющий сторонним приложениям получать ограниченный доступ к ресурсам пользователя на сервере без передачи учётных данных, через механизм выдачи и использования токенов доступа.