Домашняя работа по курсу “Управление требованиями и проектированием ИС”

Разработка требований для ООО «МедЛайн Клиника»

Выполняла:

Могушкова Фатима Магомедовна, ББИ231

Москва

2025

Оглавление

[2. Сфера деятельности компании «МедЛайн Клиника» 4](#_Toc216648750)

[2.1. Протокол интервью с заказчиком 4](#_Toc216648751)

[Основные вопросы и ответы 4](#_Toc216648752)

[2.2. Обзорная информация о компании 6](#_Toc216648753)

[2.3. Ограничения на основании регуляторных документов 7](#_Toc216648754)

[2.4. Информация о стейкхолдерах и организационной структуре 7](#_Toc216648755)

[2.5. Описание методов анализа 11](#_Toc216648756)

[**3. Сравнительный анализ с аналогичными решениями** 12](#_Toc216648757)

[**4. Анализ документации (открытые нормативные и публичные документы)** 12](#_Toc216648758)

[3. Основной перечень проблем и пути решения в ООО «МедЛайн Клиника» 12](#_Toc216648759)

[4. Модель жизненного цикла 14](#_Toc216648760)

[4.1. Выбор методологии жизненного цикла ПО 14](#_Toc216648761)

[4.2. Этапы жизненного цикла в Agile (Scrum) 16](#_Toc216648762)

[4.3. Сравнение методологии с другими методологиями 17](#_Toc216648763)

[4.4. Инструменты управления 17](#_Toc216648764)

[4.5. Заключение 21](#_Toc216648765)

[5. Функциональные требования 22](#_Toc216648766)

[5.1. FR1: Интеллектуальная онлайн-запись пациента 22](#_Toc216648767)

[5.2. FR2: Управление расписанием врачей 23](#_Toc216648768)

[5.3. FR3: Электронная медицинская карта пациента 23](#_Toc216648769)

[5.4. FR4: Уведомления и напоминания пациентам 24](#_Toc216648770)

[6. Нефункциональные требования 24](#_Toc216648771)

[6.1. Usability (Удобство использования) 25](#_Toc216648772)

[6.2. Reliability (Надёжность) 25](#_Toc216648773)

[6.3. Performance (Производительность) 25](#_Toc216648774)

[6.4. Supportability (Поддерживаемость) 26](#_Toc216648775)

[6.5. Security (Безопасность) 26](#_Toc216648776)

[7. Проектирование REST запроса 26](#_Toc216648777)

[7.1. Требование / пользовательская история 26](#_Toc216648778)

[7.3. API 27](#_Toc216648779)

[7.3.1. Описание запроса 27](#_Toc216648780)

[7.3.2. Входящее сообщение (Request) 27](#_Toc216648781)

[7.3.3. Пример запроса (JSON) 28](#_Toc216648782)

[7.3.4. Выходные данные 29](#_Toc216648783)

[7.3.5. Ответное сообщение 29](#_Toc216648784)

[7.3.6. Пример ответного сообщения 30](#_Toc216648785)

[7.3.7. Описание логики выполнения метода 30](#_Toc216648786)

[7.3.8. Спецификация в формате OpenAPI (YAML) 31](#_Toc216648787)

[8. Концептуальная модель данных 37](#_Toc216648788)

[8.1. СУБД 37](#_Toc216648789)

[8.2. Сущности 38](#_Toc216648790)

[8.3. Атрибуты сущностей 38](#_Toc216648791)

[8.4. ER-диаграмма 40](#_Toc216648792)

[8.5. Связи (для ER-диаграммы) 40](#_Toc216648793)

[8.6. скрипты для таблиц 41](#_Toc216648794)

[9. Диаграммы процессов 41](#_Toc216648795)

[9.1. BPMN-диаграмма процесса 41](#_Toc216648796)

[9.2 UML-диаграммы 42](#_Toc216648797)

[10. Источники 44](#_Toc216648798)

# 2. Сфера деятельности компании «МедЛайн Клиника»

## **2.1. Протокол интервью с заказчиком**

**Проект:** Разработка информационной системы управления записью пациентов и медицинскими данными  
**Компания-заказчик:** ООО «МедЛайн Клиника»  
**Дата проведения:** 12.11.2025 10:00 – 11:30  
**Формат:** Онлайн (видеоконференция)

**Участники:**

**Со стороны Заказчика (ООО «МедЛайн Клиника»):**

Иванова Елена Александровна, Директор клиники

Петров Алексей Сергеевич, Главный врач

**Со стороны Исполнителя (Проектная команда):**

Смирнов Дмитрий Игоревич, Системный аналитик

Кузнецова Анна Викторовна, Менеджер проекта

**Цель интервью**

Выявить и согласовать высокоуровневые бизнес-требования, цели и ключевые ограничения для проекта разработки информационной системы, направленной на автоматизацию процессов записи пациентов, хранения медицинских данных и повышения качества обслуживания в клинике.

**Основные вопросы и ответы**

**1. Опишите основные проблемы в текущем процессе записи и обслуживания пациентов.**

В настоящий момент запись пациентов осуществляется через несколько каналов: телефон, сайт и мессенджеры. Данные фиксируются разрозненно, часть информации ведётся вручную. Это приводит к накладкам в расписании врачей, дублированию записей и увеличению нагрузки на администраторов.

**2. Какие цели являются приоритетными для бизнеса при внедрении новой информационной системы?**

a. Оптимизация операционных процессов: снижение нагрузки на администраторов на 25–30%.  
b. Повышение качества сервиса: сокращение времени ожидания пациента при записи и визите.  
c. Рост лояльности пациентов: за счёт удобной онлайн-записи и напоминаний о приёмах.

**3. Кто основные пользователи системы и какой функционал для них критически важен?**

a. Пациенты: онлайн-запись на приём, просмотр истории посещений, получение уведомлений.  
b. Администраторы: управление расписанием врачей, контроль загрузки, быстрая обработка заявок.  
c. Врачи: доступ к медицинской карте пациента, результатам анализов и истории назначений.  
d. Руководство клиники: аналитика загрузки врачей и ключевые показатели эффективности.

**4. С какими системами должна быть интегрирована информационная система?**

Необходима интеграция с лабораторными системами для получения результатов анализов, а также с бухгалтерской системой для учёта оказанных услуг и формирования отчётности. Решение должно быть совместимо с существующей ИТ-инфраструктурой клиники.

**5. Какие нормативные ограничения необходимо учесть при разработке системы?**

Ключевыми являются требования законодательства в области защиты персональных и медицинских данных. Все данные пациентов должны храниться и обрабатываться с соблюдением требований конфиденциальности и разграничения доступа.

**Высокоуровневые бизнес-требования:**

* **BR-001:** Система должна обеспечивать онлайн-запись пациентов на приём с учётом актуального расписания врачей.
* **BR-002:** Система должна предоставлять врачам доступ к полной медицинской истории пациента в рамках их полномочий.
* **BR-003:** Система должна обеспечивать автоматические уведомления пациентов о предстоящих приёмах и изменениях в расписании.

**Ограничения:**

* **C-001:** Решение должно соответствовать требованиям ФЗ №152 «О персональных данных».
* **C-002:** Обработка медицинских данных должна осуществляться в соответствии с ФЗ №323 «Об основах охраны здоровья граждан».
* **C-003:** Срок внедрения пилотной версии системы — не более 5 месяцев.
* **C-004:** Бюджет проекта не должен превышать 13 млн руб.

**Договорённости:**

1. Заказчик предоставляет описание текущих бизнес-процессов и шаблоны документов до 25.11.2025.
2. Ответственным за согласование требований со стороны Заказчика назначен Петров А.С.
3. Проектная команда подготавливает перечень пользовательских сценариев (Use Cases) и направляет их на согласование до 10.12.2025.

## 2.2. Обзорная информация о компании

ООО «МедЛайн Клиника» — частная многопрофильная медицинская организация, оказывающая платные амбулаторно-поликлинические услуги населению. Клиника функционирует на рынке коммерческой медицины и ориентирована на обслуживание пациентов среднего и выше среднего ценового сегмента.

Основными направлениями деятельности клиники являются консультации врачей-специалистов, проведение лабораторных и инструментальных исследований, профилактические осмотры, а также сопровождение пациентов в рамках комплексных программ диагностики и лечения. Клиника оказывает услуги как физическим лицам, так и корпоративным клиентам по программам добровольного медицинского страхования (ДМС).

На рынке платных медицинских услуг ООО «МедЛайн Клиника» конкурирует с другими частными клиниками и медицинскими центрами, делая акцент на качестве сервиса, профессионализме медицинского персонала и удобстве взаимодействия с пациентами. Существенным фактором конкурентоспособности является скорость обслуживания, минимизация времени ожидания и персонализированный подход к каждому пациенту.

Ключевые бизнес-процессы клиники включают:

* запись пациентов на приём к врачам;
* планирование и управление расписанием медицинского персонала;
* оказание медицинских услуг и проведение диагностических исследований;
* хранение и обработку медицинской информации;
* информирование пациентов о приёмах, результатах исследований и рекомендациях врачей;
* финансовые расчёты и формирование отчётности.

В настоящее время автоматизация данных процессов реализована частично и фрагментарно. Используются разрозненные программные решения и ручные операции, что приводит к дублированию данных, увеличению нагрузки на административный персонал и рискам ошибок при работе с медицинской информацией. Отсутствие единой информационной системы затрудняет оперативный доступ врачей и администраторов к актуальным данным о пациентах и расписании.

С учётом текущих тенденций цифровизации здравоохранения и роста ожиданий пациентов к уровню сервиса, внедрение единой информационной системы рассматривается руководством ООО «МедЛайн Клиника» как необходимый шаг для повышения операционной эффективности, улучшения качества обслуживания и укрепления позиций клиники на рынке платных медицинских услуг.

**Источники:** официальный сайт медицинских клиник, данные бизнес-реестров, материалы отраслевых медицинских порталов и агрегаторов отзывов.

## **2.3. Ограничения на основании регуляторных документов**

При проектировании и внедрении информационной системы для ООО «МедЛайн Клиника» необходимо учитывать ряд законодательных и нормативных ограничений, связанных с обработкой персональных и медицинских данных, а также с обеспечением информационной безопасности.

**Ключевые регуляторные ограничения, влияющие на проект, приведены ниже:**

* **Федеральный закон №152-ФЗ «О персональных данных» (ст. 5, ст. 19)**  
  Устанавливает требования к законности обработки персональных данных, обеспечению их конфиденциальности и защите от несанкционированного доступа. В рамках проекта это требует реализации механизмов разграничения прав доступа пользователей, шифрования данных, а также хранения персональных данных пациентов на защищённых серверах.
* **Федеральный закон №323-ФЗ «Об основах охраны здоровья граждан в Российской Федерации» (ст. 91)**  
  Определяет требования к защите врачебной тайны и медицинской информации. Влияние на проект выражается в необходимости строгого контроля доступа к медицинским данным, журналирования действий пользователей системы и обеспечения сохранности истории медицинских записей.
* **Федеральный закон №149-ФЗ «Об информации, информационных технологиях и о защите информации» (ст. 16)**  
  Регламентирует общие требования к защите информации, включая предотвращение утечек, уничтожения или искажения данных. Для проекта это означает необходимость внедрения мер информационной безопасности, резервного копирования и защиты от внешних и внутренних угроз.
* **Приказы и методические рекомендации ФСТЭК и ФСБ РФ в области защиты информации**  
  Устанавливают требования к построению систем защиты информации для информационных систем, обрабатывающих персональные данные. В рамках проекта необходимо учитывать классификацию информационной системы и применять соответствующие организационные и технические меры защиты.

## **2.4. Информация о стейкхолдерах и организационной структуре**

В рамках проекта разработки и внедрения информационной системы управления записью пациентов и медицинскими данными были выявлены ключевые стейкхолдеры, заинтересованные в результатах проекта и оказывающие влияние на его реализацию.

|  |  |
| --- | --- |
| **Роль / Должность** | **Описание вклада и интересов в проекте** |
| **Генеральный директор клиники** | Определяет стратегические цели проекта, утверждает бюджет, сроки и ключевые решения. Заинтересован в повышении операционной эффективности клиники, росте выручки и укреплении конкурентных позиций на рынке платных медицинских услуг. |
| **Главный врач** | Является владельцем медицинских бизнес-процессов. Заинтересован в обеспечении высокого качества оказания медицинской помощи, удобстве доступа врачей к медицинским данным и соблюдении требований законодательства в сфере здравоохранения. |
| **Заместитель директора по ИТ / ИТ-специалист** | Отвечает за техническую реализуемость решения, интеграцию информационной системы с существующей ИТ-инфраструктурой, а также за вопросы информационной безопасности и сопровождения системы. |
| **Администраторы клиники** | Являются ключевыми пользователями системы. Заинтересованы в упрощении процессов записи пациентов, снижении количества ошибок в расписании и сокращении рутинных операций. |
| **Врачи клиники** | Используют систему в повседневной работе. Заинтересованы в быстром и удобном доступе к медицинской карте пациента, истории посещений и результатам исследований. |
| **Проектная команда (менеджер проекта, аналитики, разработчики, тестировщики)** | Реализует проектные задачи, формирует требования, разрабатывает и внедряет информационную систему, обеспечивает техническую и бизнес-экспертизу, взаимодействует с заказчиком и конечными пользователями. |
| **Пациенты (пользователи системы)** | Конечные пользователи сервиса онлайн-записи. Заинтересованы в удобстве записи на приём, своевременных уведомлениях, прозрачности информации о визитах и услугах клиники. |
| **Внешние подрядчики и партнёры** | Могут быть привлечены для поставки программных решений, интеграции лабораторных систем, технической поддержки и консультаций по вопросам информационной безопасности. |

Организационная структура

Изображение выглядит как текст, диаграмма, чек, снимок экрана

Автоматически созданное описание

ООО «МедЛайн Клиника» имеет функционально-иерархическую организационную структуру, характерную для частных медицинских организаций среднего масштаба. Управление клиникой осуществляется через разделение ответственности между медицинским, административным и ИТ-блоками.

**1. Генеральный директор (CEO)**

Высшее должностное лицо клиники.

**Функции:**

* определение стратегии развития клиники;
* утверждение бюджета и ключевых проектов;
* принятие решений о внедрении информационных систем;
* контроль финансово-экономической деятельности.

**2. Медицинский директор**

Подчиняется генеральному директору.

**Функции:**

* управление медицинской деятельностью клиники;
* контроль качества оказания медицинских услуг;
* утверждение медицинских регламентов и протоколов;
* взаимодействие с врачами и контроль расписаний приёма.

**3. Административно-управленческий блок**

3.1. Главный администратор клиники

**Функции:**

* организация работы регистратуры и call-центра;
* контроль записи пациентов и клиентского сервиса;
* координация работы администраторов;
* взаимодействие с пациентами по организационным вопросам.

3.2. Администраторы клиники

**Функции:**

* запись пациентов на приём;
* работа с обращениями и запросами клиентов;
* консультации по услугам клиники;
* ведение данных пациентов в информационной системе.

**4. Медицинский персонал**

4.1. Врачи-специалисты

**Функции:**

* проведение приёма пациентов;
* ведение медицинской документации;
* работа с расписанием приёмов;
* взаимодействие с системой записи и карточками пациентов.

4.2. Средний медицинский персонал (медсёстры, ассистенты)

**Функции:**

* сопровождение приёма пациентов;
* подготовка кабинетов;
* помощь врачам;
* фиксация данных в системе (при необходимости).

**5. Финансово-экономический блок**

5.1. Финансовый менеджер / бухгалтерия

**Функции:**

* ведение финансового и налогового учёта;
* контроль оплат услуг;
* формирование отчётности;
* взаимодействие с платёжными системами и страховыми компаниями.

**6. ИТ-блок (внутренний или аутсорсинг)**

6.1. ИТ-менеджер / внешний ИТ-подрядчик

**Функции:**

* сопровождение информационных систем клиники;
* обеспечение информационной безопасности;
* поддержка пользователей;
* взаимодействие с разработчиками CRM / MedCRM.

## 2.5. Описание методов анализа

**1. Анализ открытых источников**

Основным методом сбора информации стал анализ открытых источников, включающий:

* официальный сайт медицинской клиники;
* сайты и агрегаторы медицинских услуг (например, сервисы записи к врачам);
* государственные бизнес-реестры (ЕГРЮЛ);
* публикации отраслевых СМИ;
* агрегаторы отзывов пациентов.

Данный метод позволил получить общее представление о деятельности клиники, перечне предоставляемых медицинских услуг, позиционировании на рынке платных медицинских услуг, а также об уровне цифровизации процессов взаимодействия с пациентами.

На основании анализа открытых источников были выделены ключевые бизнес-процессы клиники, такие как запись пациентов на приём, взаимодействие с администраторами, информирование пациентов и организация работы врачей.

Результаты анализа легли в основу обзорного описания компании, её рынка и основных направлений деятельности.

**2. Анализ пользовательского опыта на основе публичных отзывов**

В качестве дополнительного метода был использован анализ пользовательского опыта, основанный на изучении отзывов пациентов.

Данный метод позволил:

* выявить типовые проблемы пациентов, связанные с записью на приём;
* определить ожидания пользователей в части удобства сервиса и скорости обслуживания;
* зафиксировать повторяющиеся жалобы, связанные с человеческим фактором и отсутствием автоматизации.

### **3. Сравнительный анализ с аналогичными решениями**

Для формирования представления о возможных функциональных возможностях системы был проведён сравнительный анализ аналогичных цифровых решений, используемых в медицинских клиниках.

В рамках данного метода анализировались:

* функциональность онлайн-записи;
* личные кабинеты пациентов;
* способы уведомлений и подтверждений приёма;
* интерфейсные решения и пользовательские сценарии.

Сравнительный анализ позволил выделить базовый набор функций, ожидаемых пользователями от современной медицинской CRM-системы, и использовать его при формировании требований к проектируемой системе.

### **4. Анализ документации (открытые нормативные и публичные документы)**

Метод применялся для уточнения информации в разделах 2.1 и 2.2. Анализировались открытые нормативные документы и регламенты, применимые к медицинской деятельности и ИТ-системам (федеральные законы, требования к обработке персональных данных, общие правила оказания медицинских услуг), а также публичные пользовательские соглашения и политики конфиденциальности аналогичных сервисов. Метод позволил выявить базовые регуляторные ограничения и типовые требования, которые необходимо учитывать при проектировании системы. Использованные документы и ссылки на них указаны в приложениях.

3. Основной перечень проблем и пути решения в ООО «МедЛайн Клиника»

В ходе анализа текущей деятельности ООО «МедЛайн Клиника», были выявлены следующие основные проблемы, негативно влияющие на операционную эффективность клиники и качество обслуживания пациентов.

**Проблема 1. Отсутствие единой системы управления записью пациентов и расписанием врачей**

**Что?**  
В клинике отсутствует единая централизованная информационная система для управления записью пациентов и расписанием врачей.

**Где?**  
Проблема проявляется на этапе взаимодействия пациентов с администраторами клиники и при планировании приёмов медицинским персоналом.

**Почему это проблема?**  
Запись пациентов осуществляется через несколько каналов (телефон, сайт, мессенджеры), при этом данные фиксируются в разрозненных системах и частично вручную. Это приводит к накладкам в расписании врачей, дублированию записей, увеличению времени ожидания пациентов и росту нагрузки на административный персонал.

**Решение проблемы 1**

Для автоматизации процессов записи пациентов и управления расписанием врачей предлагается внедрение **единой информационной системы онлайн-записи и управления приёмами**, обеспечивающей актуальность данных и прозрачность процессов для всех участников.

Разрабатываемая система позволит:

* централизовать управление расписанием врачей;
* автоматизировать запись пациентов с учётом доступных временных слотов;
* сократить количество ошибок, связанных с ручным вводом данных;
* обеспечить автоматические уведомления пациентов и персонала.

**Преследуемые цели (SMART):**

* **Сократить количество ошибок и накладок в расписании врачей не менее чем на 30%** в течение 6 месяцев после внедрения системы.
* **Снизить среднее время записи пациента на приём не менее чем на 25%** в течение 3 месяцев после запуска системы.

**Проблема 2. Фрагментарное хранение медицинских данных и ограниченный доступ к информации**

**Что?**  
Медицинская информация о пациентах хранится в разрозненных источниках и не объединена в единую медицинскую карту.

**Где?**  
Проблема возникает в процессе работы врачей и администраторов при оказании медицинских услуг и последующем сопровождении пациентов.

**Почему это проблема?**  
Отсутствие единого хранилища медицинских данных затрудняет оперативный доступ врачей к истории посещений, результатам анализов и назначениям. Это увеличивает время обслуживания пациентов, повышает риск ошибок и снижает качество медицинской помощи.

**Решение проблемы 2**

Для устранения данной проблемы предлагается внедрение **централизованного модуля электронной медицинской карты пациента**, интегрированного с системой записи и лабораторными сервисами.

Решение позволит:

* обеспечить единое хранилище медицинских данных;
* предоставить врачам быстрый доступ к полной истории пациента в рамках их полномочий;
* повысить точность и полноту медицинской информации;
* обеспечить соблюдение требований законодательства по защите медицинских данных.

**Преследуемые цели (SMART):**

* **Сократить время поиска и доступа врача к медицинской информации пациента не менее чем на 40%** в течение 4 месяцев после внедрения системы.
* **Снизить количество инцидентов, связанных с неполными или утерянными медицинскими данными, не менее чем на 50%** в течение 6 месяцев эксплуатации системы.

# 4. Модель жизненного цикла

## 4.1. Выбор методологии жизненного цикла ПО

Жизненный цикл программного обеспечения (ЖЦ ПО) — это совокупность этапов, которые проходит программный продукт с момента возникновения идеи и формирования требований до внедрения, эксплуатации и сопровождения системы. Жизненный цикл определяет последовательность работ, роли участников и способы управления изменениями на протяжении всего периода разработки и использования ПО.

Для проекта разработки информационной системы управления записью пациентов и медицинскими данными в ООО «МедЛайн Клиника» в качестве модели жизненного цикла выбрана **гибкая методология Agile (Scrum)**.

**Обоснование выбора методологии Scrum:**

Выбор методологии Scrum обусловлен спецификой проекта и следующими факторами:

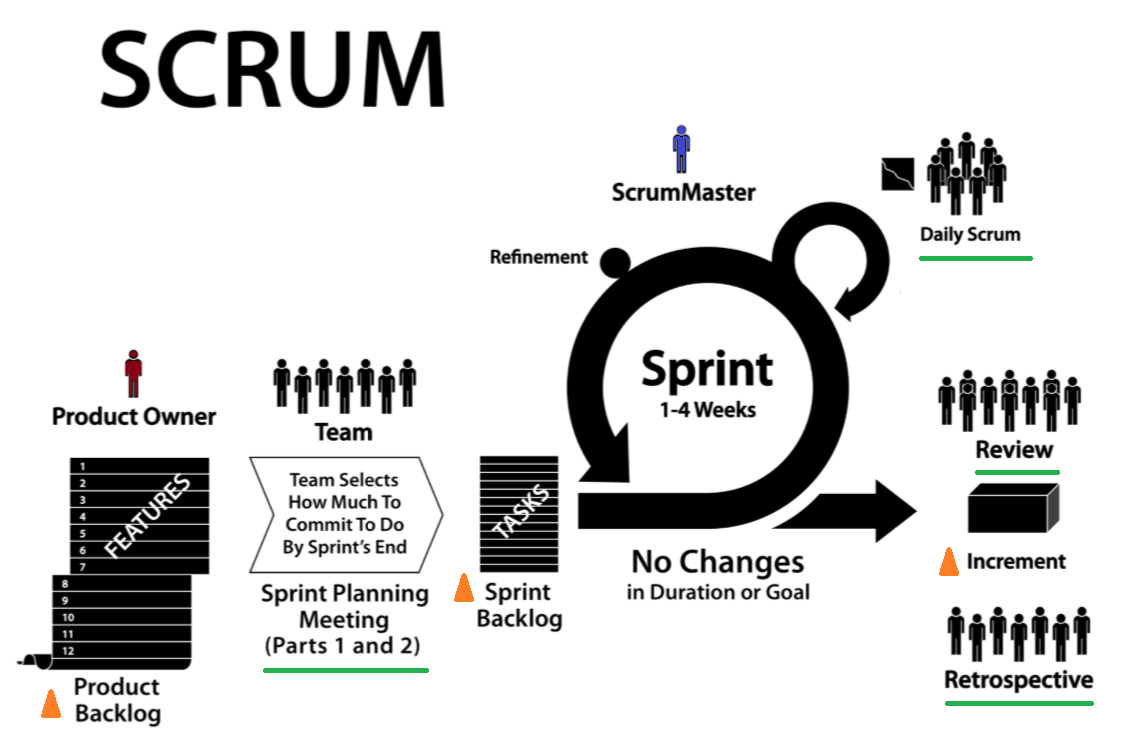
1. **Неполнота и изменяемость требований на старте проекта**  
   В процессе внедрения медицинской информационной системы требования со стороны врачей, администраторов и руководства могут уточняться и изменяться. Scrum позволяет гибко управлять требованиями и вносить изменения между спринтами без необходимости пересмотра всего плана проекта.
2. **Необходимость регулярного взаимодействия с заказчиком и пользователями**  
   Для клиники критически важно получать работающий функционал поэтапно и оперативно собирать обратную связь от медицинского персонала. Scrum предполагает регулярные демонстрации результатов (Sprint Review) и ретроспективы, что способствует более точному соответствию системы реальным бизнес-потребностям.
3. **Постепенное внедрение и снижение проектных рисков**  
   Итеративная разработка позволяет поэтапно вводить функциональность в эксплуатацию, выявлять проблемы на ранних стадиях и снижать риски, связанные с разработкой сложной информационной системы в сфере здравоохранения.

**Выбранная модель жизненного цикла:**

В рамках методологии Scrum разработка системы организуется в виде последовательных **спринтов продолжительностью 2–4 недели**, в ходе которых команда реализует приоритетные элементы бэклога продукта. По итогам каждого спринта формируется готовый инкремент системы, пригодный для демонстрации и тестирования.

**Основные роли в Scrum-процессе проекта:**

* Product Owner (представитель заказчика);
* Scrum Master;
* Кросс-функциональная команда разработки.



## 4.2. Этапы жизненного цикла в Agile (Scrum)

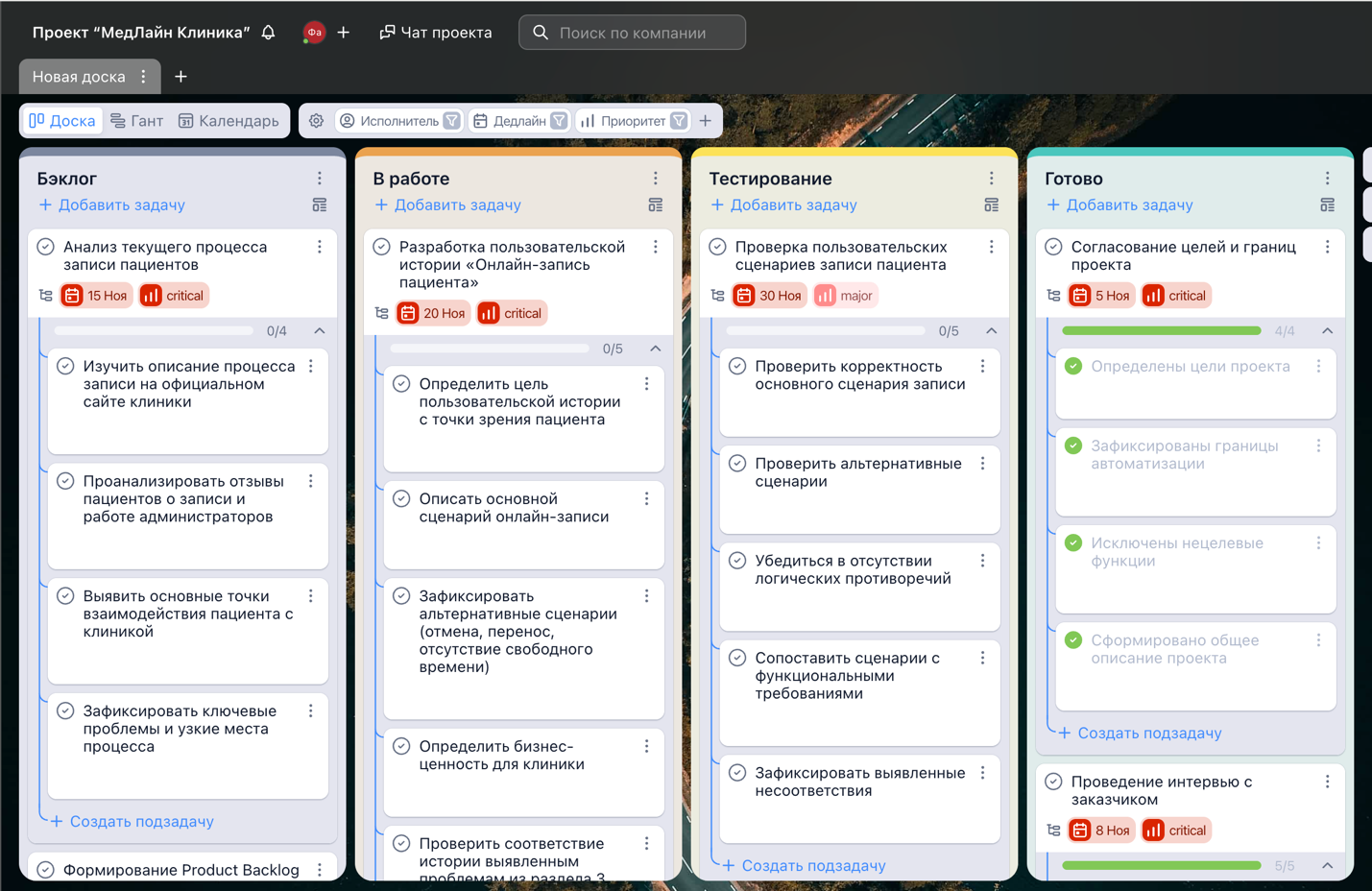
|  |  |  |
| --- | --- | --- |
| **Этап** | **Описание** | **Результаты** |
| **Инициация** | Определение целей проекта, формирование видения продукта, выявление ключевых стейкхолдеров и высокоуровневых требований. Назначение ролей Product Owner и Scrum Master. | Сформированное видение продукта, определённый состав проектной команды, согласованные цели и границы проекта. |
| **Планирование** | Формирование и приоритизация Product Backlog, проведение Sprint Planning, определение целей спринта и набора задач для реализации. | Product Backlog с приоритетами, Sprint Backlog, зафиксированные цели спринта. |
| **Разработка (итерация)** | Реализация функциональности в рамках спринта, ежедневные синхронизации (Daily Scrum), уточнение требований и разработка инкремента продукта. | Реализованный инкремент системы, готовый к демонстрации и тестированию. |
| **Тестирование** | Проверка реализованного функционала на соответствие требованиям, выявление и устранение дефектов, проведение функционального и интеграционного тестирования. | Протестированный инкремент продукта, список выявленных и устранённых дефектов. |
| **Релиз** | Подготовка и выпуск новой версии системы, демонстрация результатов заказчику (Sprint Review), принятие решения о вводе функционала в эксплуатацию. | Развернутая версия системы, принятая заказчиком функциональность. |
| **Сопровождение** | Поддержка системы в эксплуатации, обработка инцидентов, сбор обратной связи пользователей, формирование новых требований для Product Backlog. | Стабильная работа системы, обновлённый Product Backlog, предложения по развитию продукта. |

## 4.3. Сравнение методологии с другими методологиями

|  |  |  |
| --- | --- | --- |
| **Критерий** | **Agile (Scrum / Kanban)** | **Waterfall** |
| **Гибкость** | Высокая гибкость. Возможность изменения требований между итерациями без пересмотра всего плана проекта. | Низкая гибкость. Изменения требований требуют возврата к предыдущим этапам и пересмотра документации. |
| **Длительность этапов** | Короткие итерации (спринты) продолжительностью 2–4 недели с регулярным выпуском инкрементов. | Длительные последовательные этапы, каждый из которых завершается перед переходом к следующему. |
| **Участие заказчика** | Постоянное участие заказчика: формирование бэклога, регулярные демонстрации и обратная связь. | Ограниченное участие заказчика: в основном на этапах сбора требований и приёмки итогового результата. |
| **Риски** | Риски выявляются и снижаются на ранних этапах за счёт итеративной разработки и частой обратной связи. | Высокие риски обнаружения проблем на поздних этапах проекта, когда изменения дорогостоящи. |
| **Подходит для** | Проектов с изменяемыми требованиями, высокой неопределённостью и необходимостью быстрого получения результата. | Проектов с чётко зафиксированными требованиями и стабильной предметной областью. |

## **4.4. Инструменты управления**

Для эффективного управления задачами в рамках гибкой методологии Agile (Scrum) в проекте используется **Kanban-доска**, обеспечивающая наглядность статуса работ, прозрачность процесса и контроль загрузки команды, был выбран инструмент **Yougile**



Изображение выглядит как текст, программное обеспечение, веб-страница, Веб-сайт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

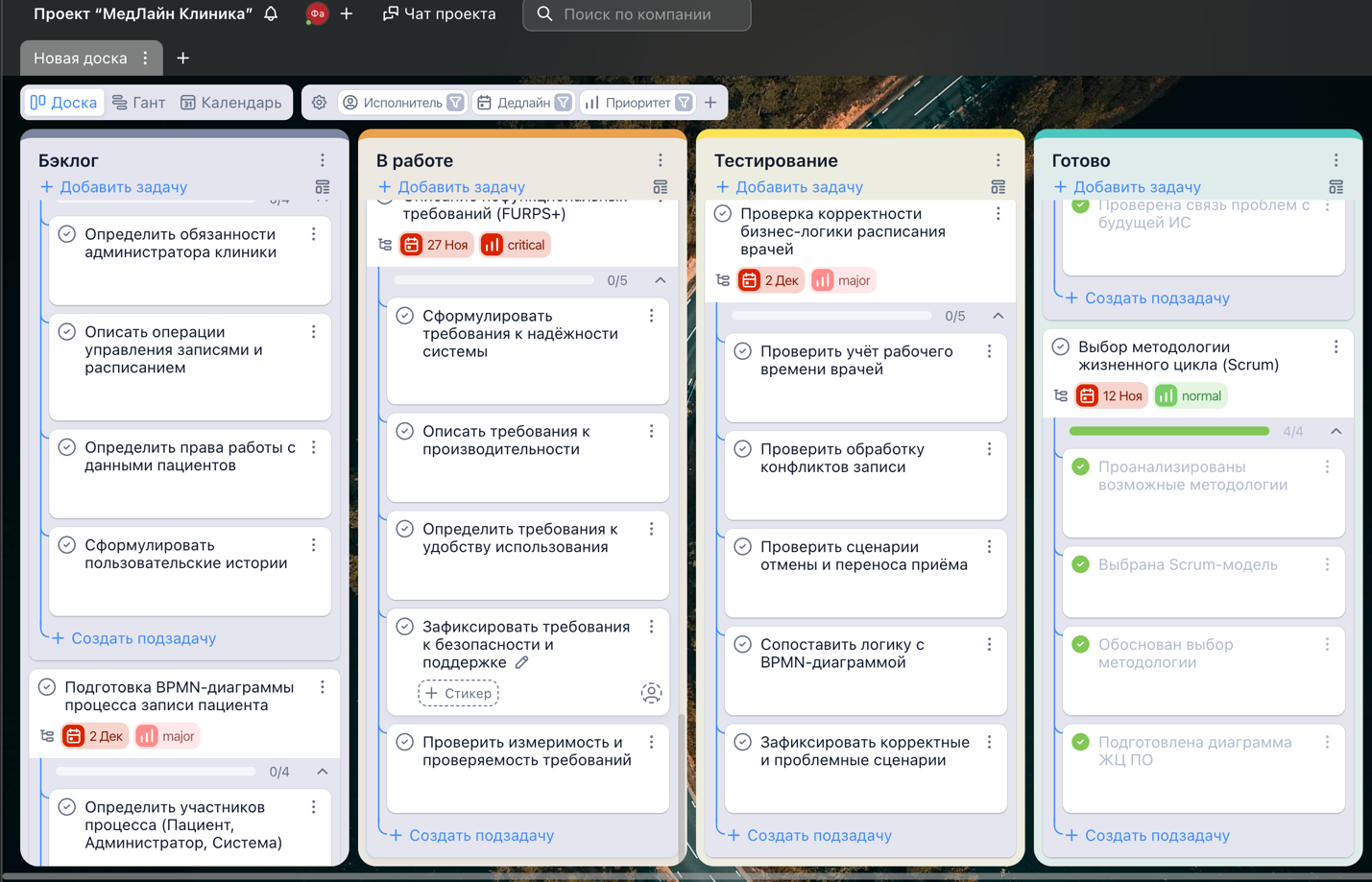
Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Изображение выглядит как текст, программное обеспечение, веб-страница, Веб-сайт

Автоматически созданное описание



Изображение выглядит как текст, программное обеспечение, веб-страница, Веб-сайт

Автоматически созданное описание

## **4.5. Заключение**

Выбор гибкой методологии Agile (Scrum) для реализации проекта разработки информационной системы управления записью пациентов и медицинскими данными в ООО «МедЛайн Клиника» является обоснованным и соответствует специфике проекта.

Проект характеризуется высокой степенью неопределённости требований на начальных этапах, а также необходимостью их уточнения в процессе эксплуатации системы. В условиях медицинской организации требования со стороны врачей, администраторов и руководства могут изменяться по мере внедрения функционала, что делает использование жёстких каскадных моделей жизненного цикла менее эффективным.

Методология Scrum обеспечивает возможность итеративной разработки и поэтапного внедрения функциональности, что позволяет регулярно получать обратную связь от заказчика и конечных пользователей, оперативно вносить изменения и снижать риски несоответствия итогового продукта реальным бизнес-потребностям клиники. Наличие коротких спринтов и регулярных демонстраций результатов повышает прозрачность проекта и управляемость сроков и объёма работ.

Кроме того, использование Scrum способствует более эффективному управлению проектными рисками за счёт раннего выявления проблем, а также позволяет обеспечить контролируемое соблюдение регуляторных требований в сфере защиты персональных и медицинских данных.

Таким образом, выбор методологии Agile (Scrum) позволяет обеспечить гибкость, прозрачность и адаптивность процесса разработки, что является критически важным для успешной реализации проекта в сфере медицинских информационных систем.

# ****5. Функциональные требования****

Функциональные требования сформированы на основании выявленных проблем текущих процессов (раздел 3) и потребностей ключевых стейкхолдеров проекта (раздел 2.4). Каждое требование описывает конкретную функцию системы, является измеримым и проверяемым, а также имеет уникальный идентификатор.

## **5.1. FR1: Интеллектуальная онлайн-запись пациента**

**Описание требования (User Story):**  
Как пациент клиники, я хочу записываться на приём к врачу онлайн с учётом специализации, даты и времени, чтобы быстро выбрать удобный слот без обращения к администратору.

**Описание функциональности:**  
Система должна обеспечивать онлайн-запись пациента на приём к врачу с автоматической проверкой доступности расписания и предотвращением конфликтов записи.

При записи система должна учитывать:

* специализацию врача;
* дату и доступные временные интервалы;
* текущую загрузку врача;
* ограничения по длительности приёма.

**Критерии проверки (Acceptance Criteria):**

* пациент может создать запись на приём не более чем за 3 шага;
* система не позволяет записаться на занятый временной слот;
* при успешной записи пациент получает подтверждение (уведомление);
* запись отображается в расписании врача и администратора.

**Связь с проблемами:**  
Решает проблему отсутствия единой системы управления записью пациентов и накладок в расписании (Проблема 1).

## **5.2. FR2: Управление расписанием врачей**

**Описание требования (User Story):**  
Как администратор клиники, я хочу управлять расписанием врачей в единой системе, чтобы минимизировать ошибки и оптимально распределять нагрузку.

**Описание функциональности:**  
Система должна предоставлять администраторам возможность:

* создавать и редактировать расписание врачей;
* блокировать временные интервалы (отпуск, больничный);
* просматривать загрузку врачей в разрезе дней и недель.

Изменения в расписании должны автоматически применяться ко всем каналам записи.

**Критерии проверки:**

* администратор может изменить расписание врача в режиме реального времени;
* изменения немедленно отображаются в системе онлайн-записи;
* система предотвращает удаление слотов с уже существующими записями без подтверждения.

**Связь с проблемами:**  
Решает проблему дублирования данных и ошибок при ручном управлении расписанием (Проблема 1).

## **5.3. FR3: Электронная медицинская карта пациента**

**Описание требования (User Story):**  
Как врач клиники, я хочу иметь доступ к электронной медицинской карте пациента, чтобы быстро просматривать историю посещений, диагнозы и результаты исследований.

**Описание функциональности:**  
Система должна обеспечивать ведение электронной медицинской карты пациента, включающей:

* персональные данные пациента;
* историю посещений и приёмов;
* назначения и диагнозы;
* результаты лабораторных и инструментальных исследований.

Доступ к данным должен предоставляться в соответствии с ролью пользователя.

**Критерии проверки:**

* врач имеет доступ только к медицинским данным своих пациентов;
* данные пациента сохраняются и доступны при повторных визитах;
* все действия с медицинской картой журналируются.

**Связь с проблемами:**  
Решает проблему фрагментарного хранения медицинских данных и ограниченного доступа к информации (Проблема 2).

## **5.4. FR4: Уведомления и напоминания пациентам**

**Описание требования (User Story):**  
Как пациент клиники, я хочу получать напоминания о предстоящем приёме, чтобы не пропускать визиты к врачу.

**Описание функциональности:**  
Система должна автоматически отправлять уведомления пациентам:

* о подтверждении записи;
* за 24 часа до приёма;
* при изменении или отмене записи.

Уведомления могут направляться по SMS, электронной почте или через push-уведомления.

**Критерии проверки:**

* уведомление отправляется автоматически без участия администратора;
* пациент получает уведомление не позднее чем за 24 часа до приёма;
* факт отправки фиксируется в системе.

**Связь с проблемами:**  
Повышает качество обслуживания и снижает количество неявок пациентов (Проблема 1).

# ****6. Нефункциональные требования****

Нефункциональные требования сформированы с учётом целей проекта (раздел 3), особенностей предметной области здравоохранения и регуляторных ограничений (раздел 2.3). Требования сгруппированы по модели **FURPS+** и являются измеримыми и проверяемыми.

## **6.1. Usability (Удобство использования)**

**US1. Простота записи пациента**  
Система должна обеспечивать возможность записи пациента на приём не более чем за **3 пользовательских шага** от главной страницы без необходимости обращения к администратору.

**US2. Единый пользовательский интерфейс**  
Пользовательский интерфейс системы должен быть интуитивно понятным и единообразным для всех ролей (пациент, врач, администратор), при этом обучение базовым операциям не должно занимать более **30 минут**.

**US3. Адаптивность интерфейса**  
Система должна корректно работать на настольных компьютерах, планшетах и мобильных устройствах, обеспечивая корректное отображение и функциональность не менее чем в **95% распространённых браузеров**.

## **6.2. Reliability (Надёжность)**

**RL1. Непрерывность работы системы**  
Система должна обеспечивать круглосуточную доступность ключевых функций (онлайн-запись, просмотр расписания, доступ к медицинской карте) без плановых простоев в рабочее время клиники.

**RL2. Восстановление после сбоев**  
В случае критических сбоев, влияющих на доступность основных функций системы, полное восстановление работоспособности должно осуществляться не более чем за **4 часа** с момента обнаружения инцидента.

**RL3. Уровень операционной готовности**  
Система должна обеспечивать коэффициент готовности не ниже **99%** в течение календарного года, что соответствует допустимому времени недоступности не более **87,6 часов**.

## **6.3. Performance (Производительность)**

**PF1. Время отклика пользовательского интерфейса**  
Время отклика пользовательского интерфейса при выполнении операций онлайн-записи и просмотра расписания не должно превышать **2 секунд** при стандартной нагрузке.

**PF2. Обработка одновременных пользователей**  
Система должна обеспечивать стабильную работу при одновременном обслуживании не менее **300 активных пользователей** без деградации основных функций.

**PF3. Скорость доступа к медицинским данным**  
Открытие электронной медицинской карты пациента для врача должно занимать не более **3 секунд** при наличии стандартного объёма данных.

## **6.4. Supportability (Поддерживаемость)**

**SP1. Мониторинг и журналирование**  
Система должна вести журнал событий и ошибок, позволяющий идентифицировать причину инцидента не более чем за **15 минут** с момента его возникновения.

**SP2. Масштабируемость и расширяемость**  
Архитектура системы должна позволять добавление новых функциональных модулей (например, интеграция с новыми лабораторными сервисами) без переработки существующих компонентов.

**SP3. Обновление системы**  
Обновление программного обеспечения системы должно выполняться без остановки работы клиники либо с простоем не более **30 минут** в нерабочее время.

## **6.5. Security (Безопасность)**

**SC1. Защита персональных и медицинских данных**  
Обработка и хранение персональных и медицинских данных должны соответствовать требованиям **ФЗ №152 «О персональных данных»** и **ФЗ №323 «Об основах охраны здоровья граждан»**, включая шифрование данных и контроль доступа.

**SC2. Аутентификация и авторизация пользователей**  
Доступ к системе должен осуществляться с использованием механизмов аутентификации и ролевой модели доступа. Для пользователей с расширенными правами должна применяться двухфакторная аутентификация.

**SC3. Аудит и контроль доступа**  
Все действия пользователей, связанные с доступом к медицинским данным, должны журналироваться и храниться не менее **12 месяцев** в соответствии с требованиями **ФЗ №149 «Об информации, информационных технологиях и о защите информации»**.

# 7. Проектирование REST запроса

## **7.1. Требование / пользовательская история**

Для разработки REST-запроса выбирается требование из раздела 5:

**FR1: Интеллектуальная онлайн-запись пациента**

**Пользовательская история (из FR1):**  
Как пациент клиники, я хочу записываться на приём к врачу онлайн с учётом специализации, даты и времени, чтобы быстро выбрать удобный слот без обращения к администратору.

**Обоснование выбора FR1 для REST-запроса:**  
REST-запрос в рамках FR1 является ключевым для системы, так как обеспечивает основную бизнес-функцию — **создание записи на приём**. Запрос напрямую связан с решением **Проблемы 1** (отсутствие единой системы записи и накладки в расписании) и удовлетворяет интересы основных стейкхолдеров: пациентов, администраторов и врачей.

## 7.3. API

### 7.3.1. Описание запроса

| **Наименование** | **Описание** |
| --- | --- |
| **Цель** | Создание записи пациента на приём к выбранному врачу на указанную дату и время (онлайн-запись). |
| **Источник** | UI (форма онлайн-записи пациента). |
| **Потребитель** | Сервис «MedLine Appointment Service» (подсистема управления расписанием и приёмами). |
| **Механизм обмена** | 2-Way (request/response), синхронный. |
| **Способ взаимодействия** | REST API |
| **Метод** | POST |
| **URL** | /api/v1/appointments |
| **Формат запроса** | JSON |

## **7.3.2. Входящее сообщение (Request)**

Контейнер — это вложенность атрибута (если атрибут в корне, контейнер не заполняем).  
Обязательность: **О** — обязательный, **Н** — необязательный, **УО** — условно обязательный.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Контейнер** | **Атрибут** | **Тип данных** | **Размерность** | **Обязательность** | **Описание** | **Особенности заполнения** |
|  | patientName | String | 150 | О | ФИО пациента | Не менее 2 слов, только буквы/пробелы; лишние пробелы обрезать |
|  | patientPhone | String | 20 | О | Телефон пациента | Формат E.164 или маска +7 (XXX) XXX-XX-XX; проверка длины |
|  | doctorId | String | 36 | О | Идентификатор врача | UUID; выбирается из справочника врачей |
|  | specializationId | String | 36 | УО | Идентификатор специализации | Обязателен, если doctorId не передан (в нашем UI передаём всегда, но правило фиксируем) |
|  | appointmentDate | String | 10 | О | Дата приёма | Формат YYYY-MM-DD; дата не может быть в прошлом |
|  | timeSlot | String | 5 | О | Время приёма | Формат HH:MM; должно соответствовать доступным слотам врача |
|  | channel | String | 20 | Н | Канал записи | Возможные значения: web, mobile, callcenter |
| meta | clientRequestId | String | 36 | Н | Идентификатор запроса клиента | UUID для идемпотентности/трассировки (рекомендуется) |

## **7.3.3. Пример запроса (JSON)**

{

"patientName": "Иванова Мария Сергеевна",

"patientPhone": "+79991234567",

"doctorId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",

"specializationId": "b7b3c7a1-1b0a-4f9b-9f6b-8f0d7d2a1123",

"appointmentDate": "2025-04-10",

"timeSlot": "14:30",

"channel": "web",

"meta": {

"clientRequestId": "9c2a3d1e-6e7f-4e2b-9c7a-1a2b3c4d5e6f"

}

}

**Пример успешного ответа (201 Created):**

{

"appointmentId": "c1b2a3d4-5e6f-7a8b-9c0d-1e2f3a4b5c6d",

"status": "CONFIRMED",

"doctorName": "Петров Илья Андреевич",

"specializationName": "Кардиолог",

"appointmentDate": "2025-04-10",

"timeSlot": "14:30",

"createdAt": "2025-04-01T12:45:10Z"

}

**Ошибки (коды)**

* **400 Bad Request** — некорректные/неполные данные (например, неверный формат даты/телефона)
* **401 Unauthorized** — пользователь не авторизован (если запись доступна только из ЛК)
* **409 Conflict** — выбранный слот уже занят / конфликт расписания
* **500 Internal Server Error** — внутренняя ошибка сервиса

## 7.3.4. Выходные данные

| **Результат выполнения запроса** | **Представление** |
| --- | --- |
| **Успешно** | **Code 201** + ответное сообщение (создана запись на приём) |
| **Перечисление ошибок с кодами и дополнительными атрибутами** | Code **400 / 401 / 403 / 409 / 500** + тело ошибки |

Пример:

| **Результат выполнения запроса** | **Представление** |
| --- | --- |
| Недостаточно прав | Code **403** |
| Пользователь не авторизован | Code **401** |
| Заполнены не все обязательные поля/ некорректный формат | Code **400** + "error": "..." |
| Конфликт записи (слот уже занят) | Code **409** + "error": "..." |

## 7.3.5. Ответное сообщение

Успешный ответ (HTTP code 201)

|  |  |  |  |
| --- | --- | --- | --- |
| **Атрибут** | **Тип данных** | **Размерность** | **Описание** |
| appointmentId | string | 36 | Идентификатор созданной записи (UUID) |
| status | string | 20 | Статус записи (CONFIRMED/PENDING/CANCELLED) |
| doctorName | string | 150 | ФИО врача |
| specializationName | string | 100 | Название специализации |
| appointmentDate | string | 10 | Дата приёма в формате YYYY-MM-DD |
| timeSlot | string | 5 | Время приёма в формате HH:MM |
| createdAt | string | 25 | Дата/время создания записи в формате ISO-8601 |

Неуспешный ответ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **HTTP code** | **Атрибут** | **Тип данных** | **Размерность** | **Описание** | **Правило заполнения** |
| 400 | error | string | 1000 | Текст сообщения об ошибке | Заполнить строкой: Некорректные данные: {1}, где {1} — перечень ошибок валидации (например, неверный формат телефона/даты) |
| 401 | error | string | 1000 | Текст сообщения об ошибке | Заполнить строкой: Пользователь не авторизован |
| 403 | error | string | 1000 | Текст сообщения об ошибке | Заполнить строкой: Недостаточно прав для выполнения операции |
| 409 | error | string | 1000 | Текст сообщения об ошибке | Заполнить строкой: Выбранный слот недоступен |
| 500 | error | string | 1000 | Текст сообщения об ошибке | Заполнить строкой: Внутренняя ошибка сервера. Повторите попытку позже |

## 7.3.6. Пример ответного сообщения

Пример успешного ответа (201):

{

"appointmentId": "c1b2a3d4-5e6f-7a8b-9c0d-1e2f3a4b5c6d",

"status": "CONFIRMED",

"doctorName": "Петров Илья Андреевич",

"specializationName": "Кардиолог",

"appointmentDate": "2025-04-10",

"timeSlot": "14:30",

"createdAt": "2025-04-01T12:45:10Z"

}

Пример ошибки (409):

{

"error": "Выбранный слот недоступен"

}

## **7.3.7. Описание логики выполнения метода**

1. **Выполнить проверки:**
   1. **Проверка прав доступа**
      * Если пользователь не авторизован → вернуть **HTTP 401**.
      * Если роль пользователя не позволяет создавать запись (например, запрещено политикой доступа) → вернуть **HTTP 403**.
   2. **Проверка входящего сообщения**
      * Если не заполнены обязательные поля (patientName, patientPhone, doctorId, appointmentDate, timeSlot) или формат некорректен → вернуть **HTTP 400** и сообщение об ошибке.
   3. **Другие бизнес-проверки**
      * Проверить, что appointmentDate не в прошлом.
      * Проверить, что doctorId существует и активен.
      * Проверить, что слот timeSlot входит в доступное расписание врача на appointmentDate.
2. **Выполнение сути метода:**
   1. Проверить отсутствие конфликта записи:
      * Если на выбранный слот уже существует запись → вернуть **HTTP 409** (Выбранный слот недоступен).
   2. Создать запись на приём:
      * Сформировать appointmentId, присвоить статус CONFIRMED, сохранить запись в БД.
   3. Выполнить дополнительные действия:
      * Отправить пациенту уведомление о записи (SMS/push/email — в зависимости от настроек).
      * Записать событие в журнал аудита (кто создал запись, когда, для какого врача).
3. **Формирование ответного сообщения:**
   1. Сформировать ответ **HTTP 201** согласно контракту (поля: appointmentId, status, doctorName, specializationName, appointmentDate, timeSlot, createdAt).

## 7.3.8. Спецификация в формате OpenAPI (YAML)

openapi: 3.0.3

info:

title: MedLine Clinic API

version: "1.0.0"

description: API для онлайн-записи пациентов на прием (МедЛайн Клиника)

servers:

- url: https://api.medline.local

paths:

/api/v1/appointments:

post:

summary: Создать запись на прием

description: Создает запись пациента на прием к врачу на выбранную дату и временной слот.

operationId: createAppointment

tags:

- Appointments

security:

- bearerAuth: []

requestBody:

required: true

content:

application/json:

schema:

$ref: "#/components/schemas/CreateAppointmentRequest"

examples:

createAppointmentExample:

summary: Пример запроса

value:

patientName: "Иванова Мария Сергеевна"

patientPhone: "+79991234567"

doctorId: "3fa85f64-5717-4562-b3fc-2c963f66afa6"

specializationId: "b7b3c7a1-1b0a-4f9b-9f6b-8f0d7d2a1123"

appointmentDate: "2025-04-10"

timeSlot: "14:30"

channel: "web"

meta:

clientRequestId: "9c2a3d1e-6e7f-4e2b-9c7a-1a2b3c4d5e6f"

responses:

"201":

description: Запись успешно создана

content:

application/json:

schema:

$ref: "#/components/schemas/CreateAppointmentResponse"

examples:

createdExample:

summary: Пример успешного ответа

value:

appointmentId: "c1b2a3d4-5e6f-7a8b-9c0d-1e2f3a4b5c6d"

status: "CONFIRMED"

doctorName: "Петров Илья Андреевич"

specializationName: "Кардиолог"

appointmentDate: "2025-04-10"

timeSlot: "14:30"

createdAt: "2025-04-01T12:45:10Z"

"400":

description: Некорректные входные данные

content:

application/json:

schema:

$ref: "#/components/schemas/ErrorResponse"

examples:

badRequest:

summary: Пример ошибки 400

value:

error: "Некорректные данные: неверный формат телефона; appointmentDate в прошлом"

"401":

description: Пользователь не авторизован

content:

application/json:

schema:

$ref: "#/components/schemas/ErrorResponse"

examples:

unauthorized:

summary: Пример ошибки 401

value:

error: "Пользователь не авторизован"

"403":

description: Недостаточно прав

content:

application/json:

schema:

$ref: "#/components/schemas/ErrorResponse"

examples:

forbidden:

summary: Пример ошибки 403

value:

error: "Недостаточно прав для выполнения операции"

"409":

description: Конфликт записи (слот недоступен)

content:

application/json:

schema:

$ref: "#/components/schemas/ErrorResponse"

examples:

conflict:

summary: Пример ошибки 409

value:

error: "Выбранный слот недоступен"

"500":

description: Внутренняя ошибка сервера

content:

application/json:

schema:

$ref: "#/components/schemas/ErrorResponse"

examples:

internalError:

summary: Пример ошибки 500

value:

error: "Внутренняя ошибка сервера. Повторите попытку позже"

components:

securitySchemes:

bearerAuth:

type: http

scheme: bearer

bearerFormat: JWT

schemas:

CreateAppointmentRequest:

type: object

required:

- patientName

- patientPhone

- doctorId

- appointmentDate

- timeSlot

properties:

patientName:

type: string

maxLength: 150

description: ФИО пациента (не менее 2 слов)

example: "Иванова Мария Сергеевна"

patientPhone:

type: string

maxLength: 20

description: Телефон пациента (E.164 или формат РФ)

example: "+79991234567"

doctorId:

type: string

format: uuid

description: Идентификатор врача

example: "3fa85f64-5717-4562-b3fc-2c963f66afa6"

specializationId:

type: string

format: uuid

nullable: true

description: Идентификатор специализации (условно обязательный, если doctorId не передан)

example: "b7b3c7a1-1b0a-4f9b-9f6b-8f0d7d2a1123"

appointmentDate:

type: string

pattern: "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"

description: Дата приема в формате YYYY-MM-DD

example: "2025-04-10"

timeSlot:

type: string

pattern: "^[0-9]{2}:[0-9]{2}$"

description: Временной слот в формате HH:MM

example: "14:30"

channel:

type: string

maxLength: 20

description: Канал записи

enum: [web, mobile, callcenter]

example: "web"

meta:

$ref: "#/components/schemas/RequestMeta"

RequestMeta:

type: object

properties:

clientRequestId:

type: string

format: uuid

description: Идентификатор запроса клиента для трассировки/идемпотентности

example: "9c2a3d1e-6e7f-4e2b-9c7a-1a2b3c4d5e6f"

CreateAppointmentResponse:

type: object

required:

- appointmentId

- status

- doctorName

- specializationName

- appointmentDate

- timeSlot

- createdAt

properties:

appointmentId:

type: string

format: uuid

description: Идентификатор созданной записи

example: "c1b2a3d4-5e6f-7a8b-9c0d-1e2f3a4b5c6d"

status:

type: string

maxLength: 20

description: Статус записи

enum: [CONFIRMED, PENDING, CANCELLED]

example: "CONFIRMED"

doctorName:

type: string

maxLength: 150

description: ФИО врача

example: "Петров Илья Андреевич"

specializationName:

type: string

maxLength: 100

description: Название специализации

example: "Кардиолог"

appointmentDate:

type: string

pattern: "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"

description: Дата приема в формате YYYY-MM-DD

example: "2025-04-10"

timeSlot:

type: string

pattern: "^[0-9]{2}:[0-9]{2}$"

description: Временной слот в формате HH:MM

example: "14:30"

createdAt:

type: string

format: date-time

description: Дата и время создания записи (ISO-8601)

example: "2025-04-01T12:45:10Z"

ErrorResponse:

type: object

required:

- error

properties:

error:

type: string

maxLength: 1000

description: Сообщение об ошибке

example: "Некорректные данные: заполнены не все обязательные поля"

# 8. Концептуальная модель данных

## 8.1. СУБД

**PostgreSQL**

## 8.2. Сущности

| **Название сущности** | **Описание сущности** |
| --- | --- |
| patient | Данные пациента (клиента клиники) |
| doctor | Данные врача и его специализация |
| service | Медицинские услуги клиники |
| appointment | Запись пациента на приём (ключевая сущность процесса) |
| payment | Оплата услуг по записи (приёму) |
| clinic\_branch | Филиалы/отделения клиники (если несколько адресов) |

## 8.3. Атрибуты сущностей

**Таблица patient**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип данных** | **Обязательность** | **Ключ** | **Описание атрибута** |
| id | uuid | Да | PK | Идентификатор пациента |
| last\_name | varchar(100) | Да |  | Фамилия |
| first\_name | varchar(100) | Да |  | Имя |
| birth\_date | date | Нет |  | Дата рождения |
| phone | varchar(20) | Да |  | Телефон для связи |
| email | varchar(150) | Нет |  | Email (если есть) |
| created\_at | timestamptz | Да |  | Дата создания карточки пациента |

**Таблица doctor**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип данных** | **Обязательность** | **Ключ** | **Описание атрибута** |
| id | uuid | Да | PK | Идентификатор врача |
| last\_name | varchar(100) | Да |  | Фамилия врача |
| first\_name | varchar(100) | Да |  | Имя врача |
| specialization | varchar(120) | Да |  | Специализация (например, терапевт) |
| phone | varchar(20) | Нет |  | Контакт врача (служебный) |
| branch\_id | uuid | Да | FK →  clinic\_  branch.id | Филиал работы врача |
| is\_active | boolean | Да |  | Признак, что врач принимает пациентов |

**Таблица service**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип данных** | **Обязательность** | **Ключ** | **Описание атрибута** |
| id | uuid | Да | PK | Идентификатор услуги |
| name | varchar(200) | Да |  | Название услуги |
| category | varchar(100) | Да |  | Категория (диагностика/консультация/анализы) |
| price\_amount | numeric(12,2) | Да |  | Стоимость |
| duration\_min | integer | Да |  | Длительность (минуты) |
| is\_active | boolean | Да |  | Доступна ли услуга к записи |

**Таблица clinic\_branch**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип данных** | **Обязательность** | **Ключ** | **Описание атрибута** |
| id | uuid | Да | PK | Идентификатор филиала |
| name | varchar(200) | Да |  | Название филиала |
| address | varchar(300) | Да |  | Адрес |
| phone | varchar(20) | Да |  | Телефон регистратуры |
| work\_hours | varchar(200) | Нет |  | Режим работы (текст) |

**Таблица appointment (ключевая)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название атрибута** | **Тип данных** | **Обязательность** | **Ключ** | **Описание атрибута** |
| id | uuid | Да | PK | Идентификатор записи |
| patient\_id | uuid | Да | FK → patient.id | Пациент |
| doctor\_id | uuid | Да | FK → doctor.id | Врач |
| service\_id | uuid | Да | FK → service.id | Услуга |
| branch\_id | uuid | Да | FK → clinic\_branch.id | Филиал приёма |
| start\_at | timestamptz | Да |  | Дата и время начала |
| status | varchar(20) | Да |  | Статус: planned / confirmed / completed / cancelled / no\_show |
| created\_at | timestamptz | Да |  | Когда создана запись |
| cancel\_reason | varchar(300) | УО |  | Причина отмены |

## 8.4. ER-диаграмма

Изображение выглядит как снимок экрана

Автоматически созданное описание

## 8.5. Связи (для ER-диаграммы)

* appointment.patient\_id → patient.id (**многие записи** к **одному пациенту**)
* appointment.doctor\_id → doctor.id (**многие записи** к **одному врачу**)
* appointment.service\_id → service.id (**многие записи** к **одной услуге**)
* appointment.branch\_id → clinic\_branch.id (**многие записи** к **одному филиалу**)
* doctor.branch\_id → clinic\_branch.id (**многие врачи** в **одном филиале**)
* payment.appointment\_id → appointment.id (**одна/несколько оплат** на **одну запись**, если нужно; можно ограничить до 1:1)

## 8.6. скрипты для таблиц

* **Изображение выглядит как текст, снимок экрана, Шрифт

  Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

  Автоматически созданное описание**

# 9. Диаграммы процессов

## 9.1. BPMN-диаграмма процесса

**Изображение выглядит как текст, диаграмма, План, Технический чертеж

Автоматически созданное описание**

## 9.2 UML-диаграммы

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, чек, число

Автоматически созданное описаниеИзображение выглядит как диаграмма, зарисовка, линия, оригами

Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, чек, Шрифт

Автоматически созданное описание

# 10. Источники

 **Официальный сайт ООО «МедЛайн Клиника»**  
<https://medlineclinic.ru>

 **Раздел «Услуги» ООО «МедЛайн Клиника»**  
<https://medlineclinic.ru/services>

 **Раздел «О клинике» ООО «МедЛайн Клиника»**  
<https://medlineclinic.ru/about>

 **Онлайн-запись на приём — ООО «МедЛайн Клиника»**  
<https://medlineclinic.ru/appointment>

 **Единый государственный реестр юридических лиц (ЕГРЮЛ)**  
<https://egrul.nalog.ru>

 **ПроДокторов — сервис онлайн-записи и отзывов о врачах**  
<https://prodoctorov.ru>

 **СберЗдоровье (DocDoc)**  
<https://sberhealth.ru>

 **Яндекс.Карты**  
<https://yandex.ru/maps>

 **Google Maps**  
<https://www.google.com/maps>

 **2GIS — справочник организаций**  
<https://2gis.ru>

 **Федеральный закон № 152-ФЗ «О персональных данных»**  
<http://www.consultant.ru/document/cons_doc_LAW_61801>

 **Федеральный закон № 323-ФЗ «Об основах охраны здоровья граждан в Российской Федерации»**  
http://www.consultant.ru/document/cons\_doc\_LAW\_121895