Кировское областное государственное профессиональное

образовательное бюджетное учреждение

«Кировский авиационный техникум»

(КОГПОБУ «Кировский авиационный техникум»)

|  |  |
| --- | --- |
|  |  |

Курсовой проект

МДК 07.03 – ИТ-РЕШЕНИЯ ДЛЯ БИЗНЕСА

*Разработка информационной системы для больницы*

Пояснительная записка

КП.09.02.07.ИР41.023 ПЗ

|  |  |  |
| --- | --- | --- |
| Выполнил студент | \_\_\_\_\_\_\_\_\_\_\_\_ | Хорошев Д. Р. |
|  | подпись | Расшифровка подписи |
| Руководитель КП | \_\_\_\_\_\_\_\_\_\_\_\_ | Кононова О. А. |
|  | подпись | Расшифровка подписи |

Оценка защиты курсового проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты курсового проекта «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_\_ г.

СОДЕРЖАНИЕ

[Введение 3](#_Toc185707595)

[1 Анализ предметной области 4](#_Toc185707596)

[2 Проектирование 7](#_Toc185707597)

[3 Разработка графического интерфейса и модулей приложения 14](#_Toc185707598)

[4 Тестирование программного продукта 19](#_Toc185707599)

[Заключение 23](#_Toc185707600)

[Библиографический список источников информации 25](#_Toc185707601)

[Приложение А 26](#_Toc185707602)

[Приложение Б 38](#_Toc185707603)

# ВВЕДЕНИЕ

В современном мире информационные технологии играют ключевую роль в развитии различных сфер деятельности, включая здравоохранение. Внедрение информационных систем в медицинские учреждения позволяет повысить качество и эффективность оказания медицинской помощи, оптимизировать работу персонала и улучшить взаимодействие между различными подразделениями больницы.

Целью данного курсового проекта является разработка информационной системы для больницы, которая будет способствовать повышению эффективности работы медицинского учреждения и улучшению качества оказания медицинской помощи. Для достижения поставленной цели необходимо решить следующие задачи:

* провести анализ существующих ИС в больницах;
* определить требования к разрабатываемой системе;
* разработать структуру и функциональность системы;
* реализовать систему на выбранной платформе;
* протестировать систему и оценить её эффективность.

# АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Предметная область — это сфера деятельности, в рамках которой будет функционировать разрабатываемая информационная система. В данном случае это больница – медицинское учреждение, оказывающее медицинскую помощь пациентам.

Для разработки информационной системы необходимо провести анализ предметной области, чтобы определить основные процессы и требования к системе. Это позволит создать эффективное решение, которое будет соответствовать потребностям больницы.

Основные технологические процессы в больнице:

* распределение больных по палатам;
* назначение лечащих врачей;
* ежедневный учёт состояния больных;
* назначения препаратов, процедур.

Эти процессы требуют автоматизации для повышения эффективности работы больницы и улучшения качества оказания медицинской помощи.

Требования к информационной системе:

* Функциональность: система должна обеспечивать автоматизацию основных технологических процессов в больнице, таких как распределение пациентов по палатам, ведение учёта состояния больных, назначение препаратов и т. д.
* Надёжность: система должна быть надёжной и устойчивой к сбоям, чтобы обеспечить непрерывность работы больницы.
* Безопасность: система должна защищать данные пациентов и сотрудников от несанкционированного доступа.
* Простота использования: интерфейс системы должен быть интуитивно понятным и удобным для пользователей.
* Экономическая эффективность: внедрение системы должно быть экономически оправданным для больницы.

Обзор аналогов:

1. MEDESK

Информационная система включает в себя следующие модули: "Электронная медицинская карта" (ЭМК), "ЕГИСЗ", "Медосмотры", "Онлайн-запись", "API 2.0", "Задачи", "Онлайн-кассы 54-ФЗ", "Склад", "Работа со страховыми компаниями", "Лаборатория", "Стоматология", "Маркетинг", "СМС-рассылки", "СМС и Whatsapp уведомления", "Телефония", "Управленческая отчетность". Плата за использование MEDESK помесячная или погодовая.

1. Клиника онлайн

Информационная система включает в себя следующие модули: Онлайн-расписание доступное с компьютера, телефона и планшета, Онлайн-запись пациентов, - Кабинеты, Интеграция с МКБ 10, Бесплатное мобильное приложение для сотрудников, Учет финансов, товаров и услуг, Управление сетью клиник, Автоматический расчет зарплаты врачей, Аналитика продаж, IP-телефония, Автоматизация систем лояльности. Система поставляется только в облачном виде. Стоимость формируется от количества сотрудников, которые работают одновременно в один день.

1. Медиалог

Информационная система включает большое количество разнообразных модулей, подключаемых в зависимости от нужд клиента. Стоимость системы формируется из количества подключенных модулей.

Проблемами данных аналогов являются избыточная комплексность, отсутствие возможности приобретения коробочной версии, а также невозможность установки на локальный сервер, что является неприемлемым для заказчика. В связи с вышеизложенным, принято решение о разработке информационной системы для больницы по требованиям заказчика.

1. ПРОЕКТИРОВАНИЕ

Проектирование функциональной модели

Диаграмма IDEF0 используется для создания функциональной модели, которая является структурированным отображением функций производственной системы или среды, а также информации и объектов, связывающих эти функции.

Функциональная модель представленной информационной системы состоит из 6 диаграмм:

контекстная диаграмма (Рисунок 1)

диаграмма основных процессов (Рисунок 2)

декомпозиция процесса «Госпитализация пациентов» (Рисунок 3)

декомпозиция процесса «Лечение пациентов» (Рисунок 4)

декомпозиция процесса «Выписка пациентов» (Рисунок 5)

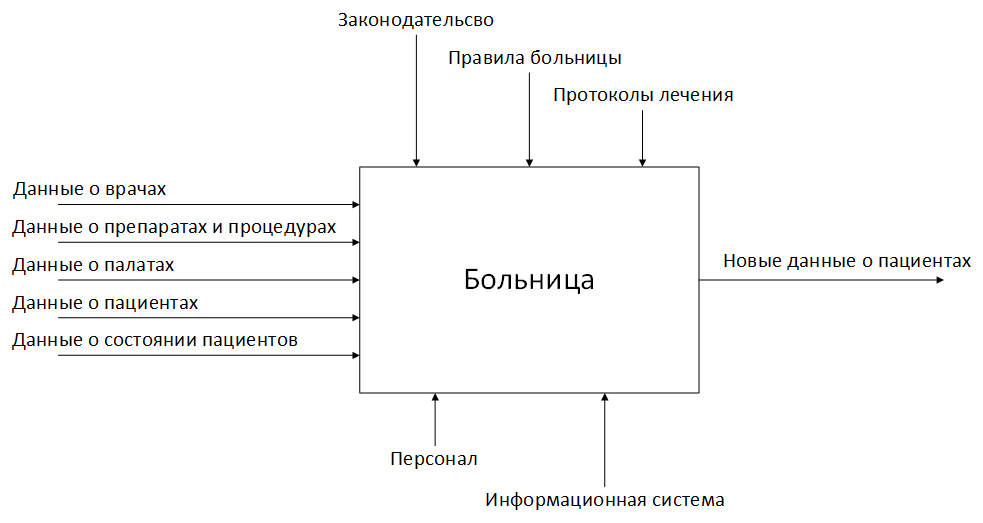


Рисунок 1 – Контекстная диаграмма

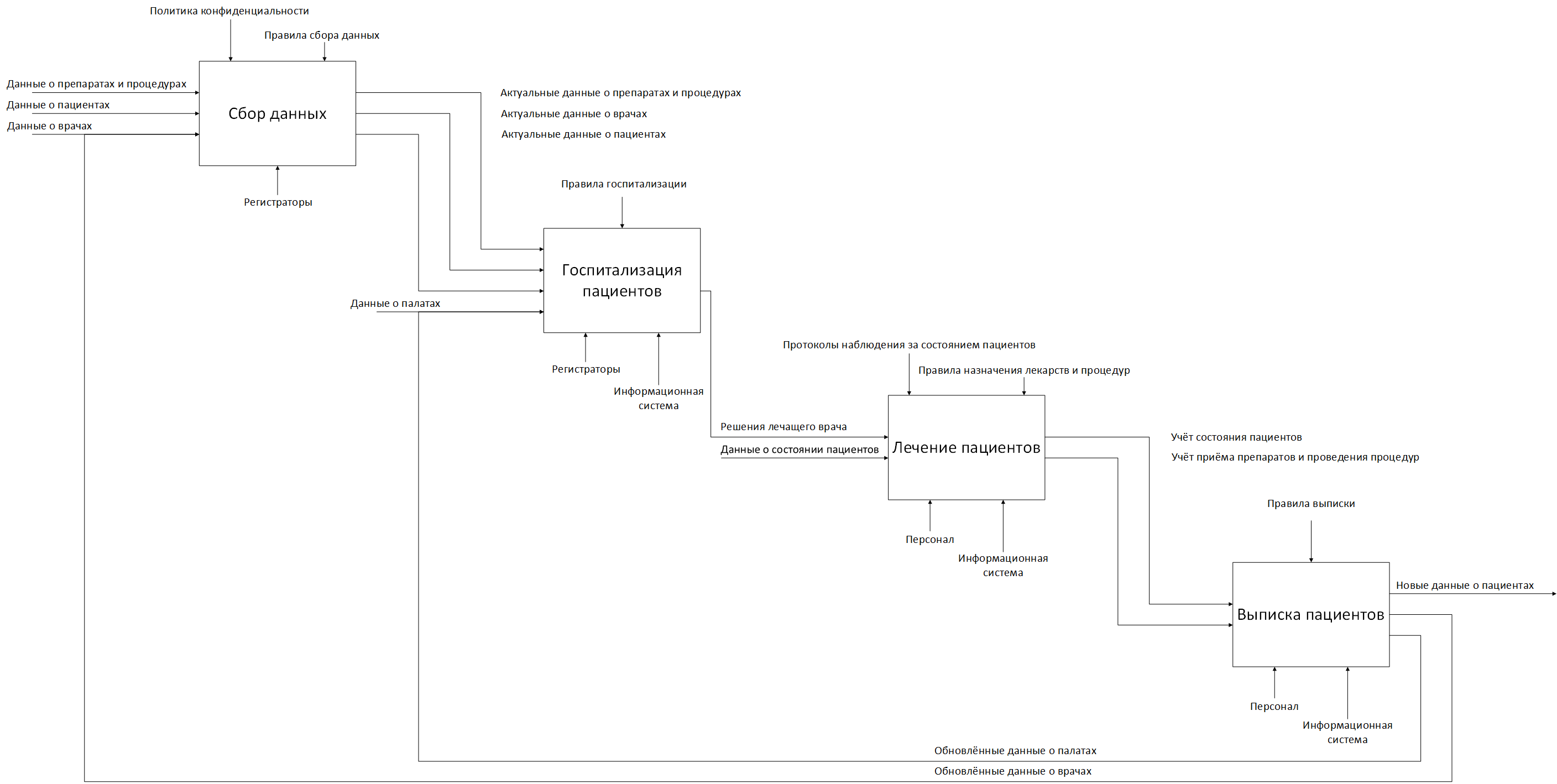


Рисунок 2 – Диаграмма основных процессов

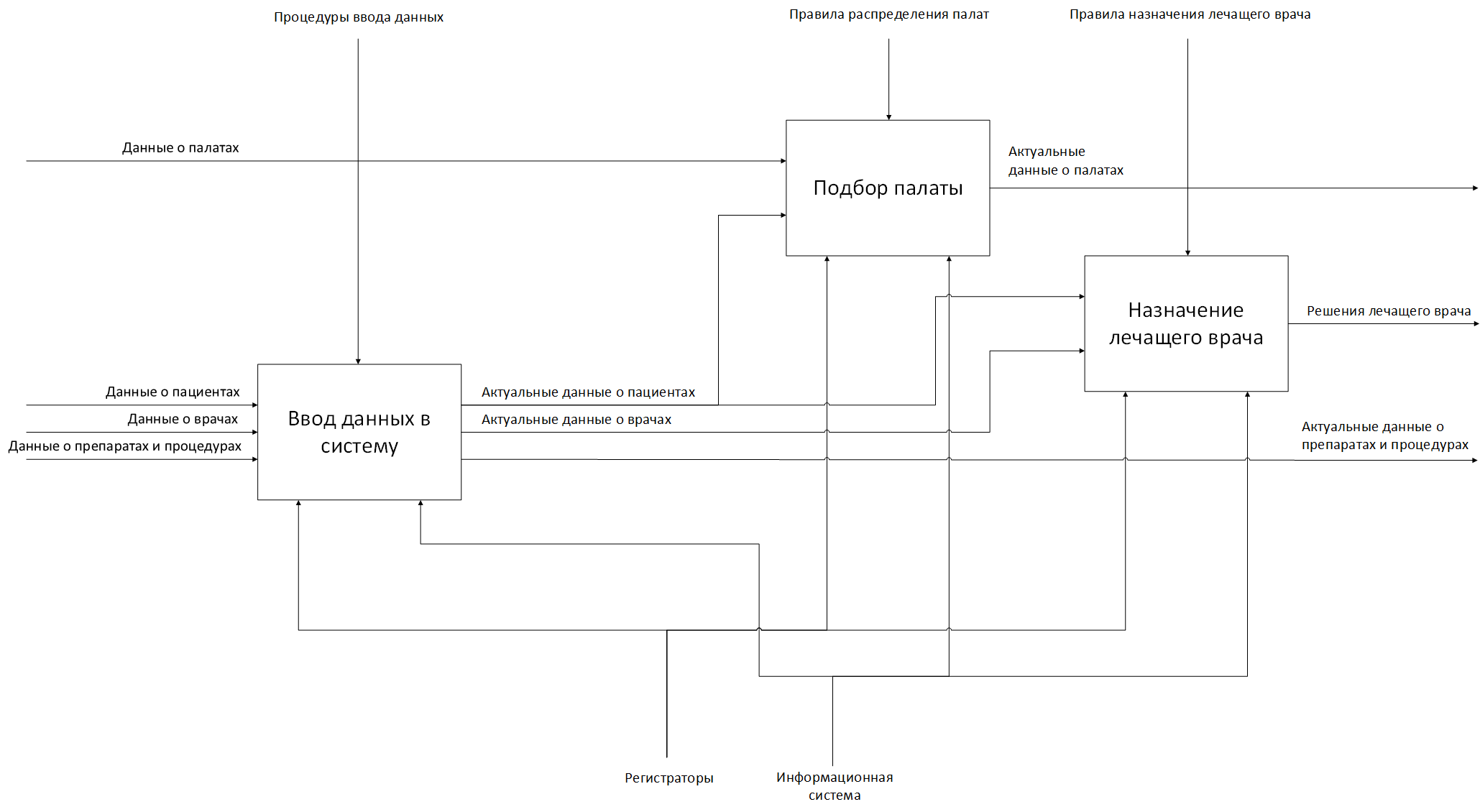


Рисунок 3 – Декомпозиция процесса «Госпитализация пациентов»

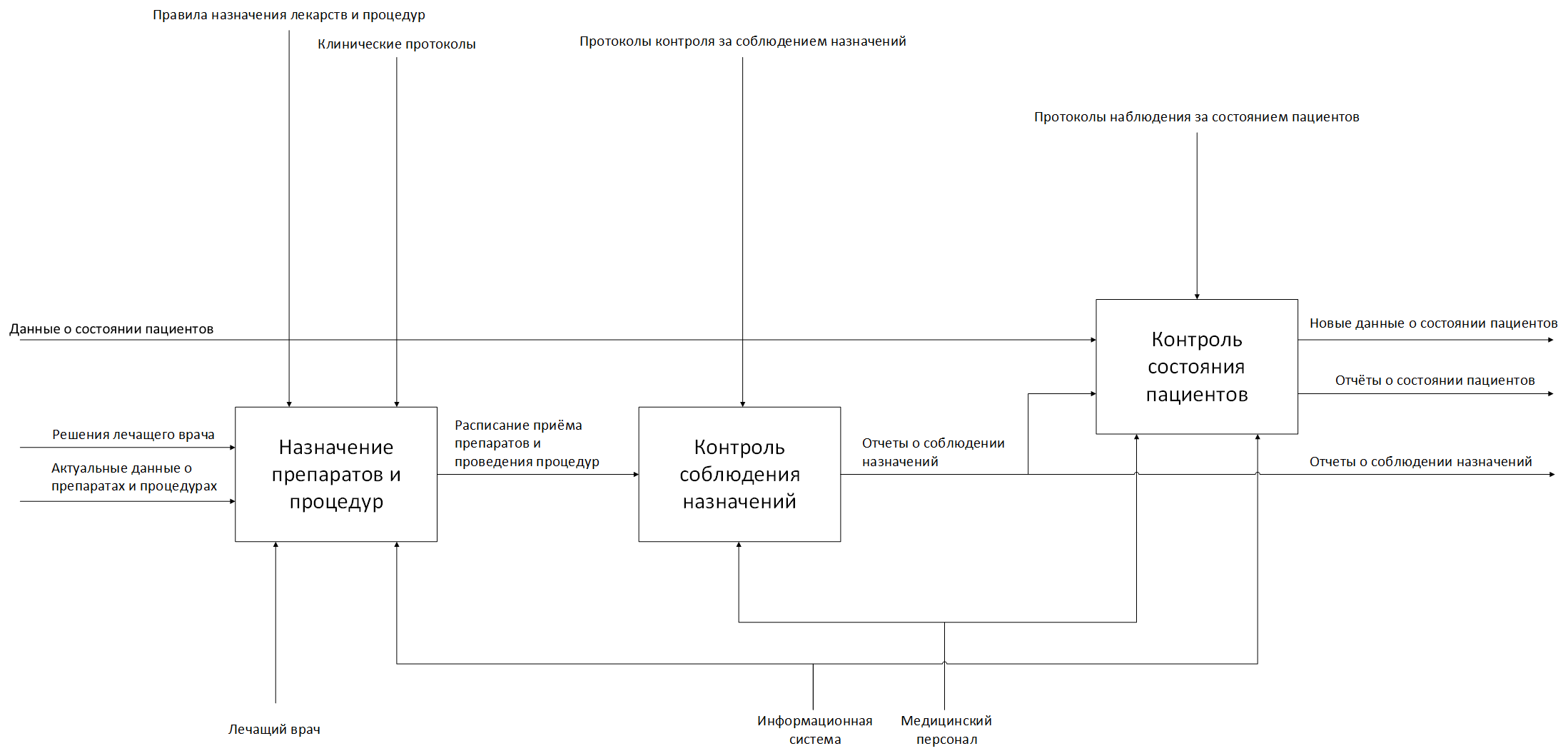


Рисунок 4 – Декомпозиция процесса «Лечение пациентов»

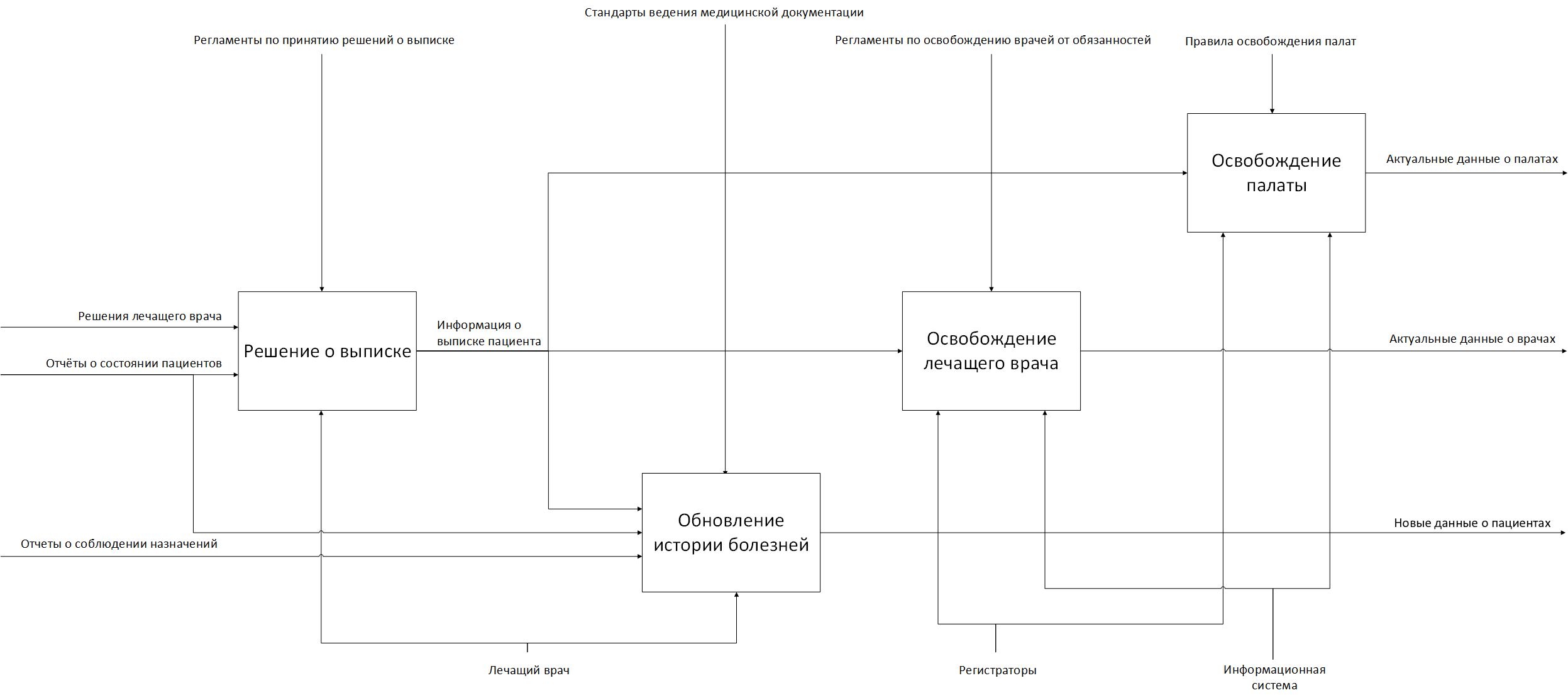


Рисунок 5 – Декомпозиция процесса «Выписка пациентов»

Проектирование базы данных

ER-диаграмма позволяет наглядно представить структуру данных и взаимосвязи между ними.

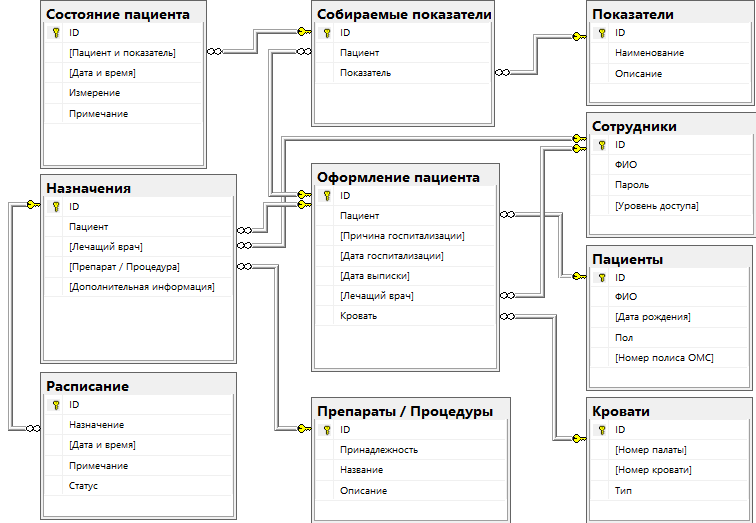


Рисунок 6 – ER-диаграмма

ER-диаграмма данного проекта включает в себя следующие сущности:

Пациент – сущность, хранящая данные обо всех пациентах больницы.

Включает в себя уникальный идентификатор пациента, фамилию, имя, отчество, дату рождения, пол и номер полиса обязательного медицинского страхования пациента.

Сотрудник – сущность, хранящая данные обо всех сотрудниках больницы.

Включает в себя уникальный идентификатор сотрудника, фамилию, имя, отчество, пароль и уровень доступа к системе (администратор, регистратор, врач, медицинский персонал или уровень без доступа).

Кровать – сущность, хранящая данные обо всех местах пребывания пациентов на время госпитализации.

Включает в себя уникальный идентификатор кровати, палату, в которой она находится, номер кровати в палате и тип кровати.

Препарат / Процедура – сущность, хранящая данные обо всех препаратах и процедурах доступных для назначения пациентам в больнице.

Включает в себя уникальный идентификатор объекта, к какой группе он относится (препарат или процедура), название объекта и его описание.

Показатель – сущность, хранящая данные обо всех показателях состояния пациентов доступных для сбора в больнице.

Включает в себя уникальный идентификатор показателя, его название и описание.

Оформление – сущность, хранящая данные об оформлении госпитализации.

Включает в себя уникальный идентификатор оформления, идентификатор пациента из таблицы «Пациенты», причину госпитализации, дату госпитализации, дату выписки, идентификатор лечащего врача из таблицы «Сотрудники» и идентификатор кровати, назначенной пациенту из таблицы «Кровати».

Собираемый показатель – сущность, хранящая данные о пациенте и показателе, который необходимо у него собирать.

Включает в себя уникальный идентификатор объекта, идентификатор оформления госпитализации из таблицы «Оформления», идентификатор показателя из таблицы «Показатели».

Состояние пациента – сущность, хранящая данные о пациенте, показателе, дате и времени сбора данных о состоянии, значении показателя и примечание с уточнением в случае необходимости.

Включает в себя уникальный идентификатор объекта, идентификатор собираемого показателя из таблицы «Собираемые показатели», дату, время, измерение и примечание.

Назначение – сущность, хранящая данные о назначении препарата или процедуры пациенту.

Включает в себя уникальный идентификатор назначения, идентификатор оформления госпитализации из таблицы «Оформления», идентификатор лечащего врача, создавшего назначение, из таблицы «Сотрудники», препарат или процедуру из таблицы «Препараты / Процедуры», и дополнительную информацию о назначении в случае необходимости.

Расписание – сущность, хранящая данные об одном конкретном приёме препарата или проведении процедуры соответственно.

Включает в себя уникальный идентификатор назначения, идентификатор назначения из таблицы «Назначения», дату и время приёма препарата или проведения процедуры, примечание, в случае необходимости, и статус выполнения (выполнено или не выполнено).

Разработка технического задания

Техническое задание – это документ, определяющий требования к разрабатываемой системе или продукту. Оно является основой разработки, управления и контроля качества проекта. В техническом задании, как правило, изложены не только основные характеристики будущего продукта, но и перечислены критерии, которым он обязан соответствовать. Это обеспечивает правовую базу, обеспечивающую защиту интересов и заказчика и разработчика.

Содержание технического задания:

* Вводная часть, включающая общие сведения, такие как наименование проекта, цели разработки и общая характеристика продукта.
* Функциональные требования, описывающие основные задачи выполняемые, которые должен будет выполнять разрабатываемый продукт и его основные функции.
* Технические требования, включающее параметры надёжности, производительности, совместимости с другими системами и требований к оборудованию, программному обеспечению.
* Дополнительно могут включаться разделы этапов реализации, сроков выполнения, критерии оценки результата, а также порядок приёмки.
* Приложения, содержащие вспомогательную информацию.

Техническое задание для данного курсового проекта разработано в соответствии с ГОСТ 19.201-79 (Приложение А).

1. РАЗРАБОТКА ГРАФИЧЕСКОГО ИНТЕРФЕЙСА И МОДУЛЕЙ ПРИЛОЖЕНИЯ

Обоснование выбора платформы

Для реализации проекта был выбран язык C#, .NET Framework 4.8, технология WPF и база данных SQLite с Entity Framework 6 в качестве ORM, а также Git для контроля версий.

C# – это объектно-ориентированный язык программирования, разработанный компанией Microsoft. C# интегрируется с библиотекой классов .NET, ускоряя разработку и улучшая качество кода. Язык подходит как для небольших проектов, так и для крупномасштабных корпоративных приложений, а большое сообщество разработчиков создаёт библиотеки, инструменты и ресурсы для помощи другим разработчикам. Он лёгок в изучении благодаря похожему на другие языки синтаксису, обеспечивает безопасность кода, предотвращая распространённые ошибки, и поддерживает объектно-ориентированное программирование, что упрощает разработку, тестирование и поддержку кода.

.NET Framework 4.8 — это версия общеязыковой исполняющей среды (Common Language Runtime, CLR) и инфраструктуры, которая позволяет запускать приложения, написанные для .NET Framework. Она включает в себя библиотеки классов, компоненты пользовательского интерфейса, инструменты разработки и другие функции, необходимые для создания и запуска приложений на платформе .NET.

WPF (Windows Presentation Foundation) – это технология, разработанная компанией Microsoft для создания интерактивных пользовательских интерфейсов. Она позволяет создавать приложения с графическим интерфейсом, которые могут работать на различных платформах и устройствах.

WPF упрощает разработку и поддержку приложений благодаря механизму привязки данных, который позволяет связывать данные из источника с элементами интерфейса. Это обеспечивает синхронизацию данных и интерфейса в реальном времени, что значительно облегчает работу разработчика.

Графические возможности WPF, включая работу с векторной графикой, растровыми изображениями, видео- и аудиофайлами, позволяют создавать сложные и красивые интерфейсы. Поддержка мультимедиа делает приложения более привлекательными и функциональными. Инструменты для анимации и визуальных эффектов в WPF придают приложениям дополнительную интерактивность и привлекательность, улучшая пользовательский опыт.

Система управления ресурсами в WPF облегчает загрузку и использование изображений, стилей и других ресурсов в приложениях, обеспечивая их эффективное и централизованное управление.

Использование языка разметки XAML в WPF упрощает создание и изменение интерфейса, делая процесс разработки более удобным и понятным.

SQLite – компактная встраиваемая база данных, которая позволяет хранить и управлять данными в приложениях. Она легко встраивается в приложения и не требует дополнительных настроек, работает на разных операционных системах, поддерживает шифрование данных и аутентификацию пользователей, использует эффективные алгоритмы для обработки запросов и управления данными, обеспечивает целостность информации с помощью транзакционных операций.

Entity Framework 6 — это ORM (Object-Relational Mapping) фреймворк, который позволяет разработчикам работать с реляционными базами данных, используя объекты и свойства. Он автоматически сопоставляет объекты с таблицами базы данных, упрощает процесс работы с данными и позволяет писать запросы к базе данных на языке программирования благодаря поддержке LINQ.

Git – это распределённая система контроля версий, которая позволяет разработчикам отслеживать изменения в исходном коде и совместно работать над проектами.

История изменений, сохраняемая Git, позволяет разработчикам возвращаться к предыдущим версиям кода при необходимости, что может быть полезно при отладке или восстановлении после ошибок.

Основная информация об интерфейсах

Интерфейсы информационной системы играют ключевую роль в обеспечении удобства и эффективности взаимодействия пользователей с системой. В контексте разработки информационной системы для больницы, интерфейсы должны быть интуитивно понятными и доступными для различных категорий пользователей.

Типы интерфейсов:

* Графический пользовательский интерфейс (GUI) – это наиболее распространенный тип интерфейса, который позволяет пользователям взаимодействовать с системой через визуальные элементы, такие как кнопки, меню и формы. Важно обеспечить высокую степень удобства использования, чтобы пользователи могли быстро и легко выполнять необходимые действия.
* Командный интерфейс (CLI) может быть полезен для опытных пользователей, которые предпочитают вводить команды напрямую.
* Веб-интерфейсы обеспечивают доступ к системе с различных устройств является важным аспектом для некоторых проектов.

Руководство пользователя

Руководство пользователя – это документ, который содержит информацию о функциональности информационной системы. Оно предназначено помочь людям научиться использовать программу или приложение для решения своих задач. Руководство должно быть понятным, доступным и информативным, чтобы пользователи могли легко найти нужную информацию и быстро освоить продукт. Оно играет важную роль в процессе освоения программы или приложения, так как оно предоставляет пользователям необходимую информацию и инструкции для эффективного использования продукта. Это особенно полезно для тех, кто не имеет специальных знаний и навыков в области информационных технологий.

Руководство пользователя для данного курсового проекта разработано в соответствии со стандартом РД 50-34.698-90 (Приложение Б).

1. ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА

Тестирование программного продукта – это процесс проверки программного обеспечения на соответствие требованиям и стандартам качества. Оно включает в себя выявление ошибок, дефектов и проблем в работе программы, а также оценку её функциональности, производительности и надёжности.

Существует несколько типов тестирования программного продукта:

* Функциональное тестирование – проверка соответствия программы функциональным требованиям.
* Нагрузочное тестирование – оценка производительности программы при различных нагрузках.
* Стресс-тестирование – проверка работы программы при экстремальных нагрузках.
* Тестирование безопасности – поиск уязвимостей в программе, которые могут быть использованы злоумышленниками.
* Юзабилити-тестирование – оценка удобства использования программы.
* Регрессионное тестирование – проверка того, что новые изменения в программе не привели к появлению новых ошибок.

Тестирование играет важную роль в процессе разработки программного обеспечения. Оно помогает выявить и исправить ошибки и дефекты в программе до её выпуска, что обеспечивает её качество и надёжность. Тестирование также позволяет оценить производительность и удобство использования программы, что важно для удовлетворения потребностей пользователей.

Для курсового проекта необходимо провести функциональное тестирование. Было проведено 3 теста.

Таблица 1 – Аннотация к тестам

|  |  |
| --- | --- |
| Название проекта | Информационная система для больницы |
| Рабочая версия | 1.0 |
| Имя тестирующего | Хорошев Дмитрий Романович |
| Дата теста | 22.12.2024 |

Таблица 2 – Тест №1

|  |  |
| --- | --- |
| Тестовый пример # | 1 |
| Приоритет тестирования | Низкий |
| Название теста | Ввод неверных значений |
| Краткое изложение теста | Необходимо произвести попытку ввода некорректного значение в любое текстовое поле, где значение может быть некорректным. Тест должен показать защищена ли система от ввода некорректных данных. |
| Этапы теста | Ввод в поле «Количество в день» в форме создания назначения строкового значения |
| Тестовые данные | “Строка” |
| Ожидаемый результат | Запрет ввода данного значения. |
| Фактический результат | Ошибка «Некорректные данные в поле «Количество в день»»  Закрытие формы |
| Статус | Зачёт |
| Предварительное условие | - |
| Постусловие | - |
| Примечания/комментарии | - |

Таблица 3 – Тест №2

|  |  |
| --- | --- |
| Тестовый пример # | 2 |
| Приоритет тестирования | Низкий |
| Название теста | Вывод сообщения, защита от вредительства |
| Краткое изложение теста | Необходимо попытаться удалить единственного администратора, что может привести к полному отказу программы, так как нельзя будет назначать должности новым сотрудникам |
| Этапы теста | Переходим на страницу Сотрудники у регистратора  Выбираем единственного администратора  Нажимаем кнопку «Удалить» |
| Тестовые данные | - |
| Ожидаемый результат | Отмена удаления  Вывод сообщения, информирующего о причине отмены |
| Фактический результат | Отмена удаления, сообщение «Невозможно удалить сотрудника с этим уровнем доступа» |
| Статус | Зачёт |
| Предварительное условие | Один администратор |
| Постусловие | - |
| Примечания/комментарии | - |

Таблица 4 – Тест №3

|  |  |
| --- | --- |
| Тестовый пример # | 3 |
| Приоритет тестирования | Высокий |
| Название теста | Расчёт времени приёма препаратов или проведения процедур при создании назначения |
| Краткое изложение теста | Необходимо создать назначение и проверить корректно ли создаются записи расписания. |
| Этапы теста | Добавление нового назначения  Ввод данных  Проверка созданных записей |
| Тестовые данные | Выбор препарата «Нурофен»  Дата начала 17.12.2024  Дата конца 24.12.2024  Количество в день 3 |
| Ожидаемый результат | Создастся 24 записи, по 3 на каждый день, время записей в промежутке с 8:00 до 20:00 |
| Фактический результат | Соответствует ожидаемому. |
| Статус | Зачет |
| Предварительное условие | - |
| Постусловие | - |
| Примечания/комментарии | - |

ЗАКЛЮЧЕНИЕ

В результате выполнения курсового проекта была разработана информационная система для больницы, направленная на оптимизацию процессов оказания медицинской помощи и улучшение взаимодействия между сотрудниками. В ходе работы были успешно решены следующие ключевые задачи:

1. Проведен анализ существующих информационных систем в медицинских учреждениях, что позволило выявить их сильные и слабые стороны.
2. Определены функциональные и технические требования к новой системе, учитывающие потребности пользователей и специфику работы больницы.
3. Разработаны структурные модели системы, включая функциональную модель и модель данных, что обеспечило четкое понимание ее архитектуры.
4. Создан интуитивно понятный интерфейс пользователя, способствующий легкости освоения системы различными категориями пользователей.
5. Реализована база данных, которая обеспечивает надежное хранение и обработку информации о пациентах и медицинских процедурах.
6. Разработаны модули системы, охватывающие все ключевые аспекты работы больницы, от регистрации пациентов до управления медицинскими записями.
7. Проведено тестирование системы, что подтвердило ее стабильность и эффективность в реальных условиях.

Информационная система для больницы обладает рядом преимуществ, среди которых: улучшение качества обслуживания пациентов, повышение скорости обработки информации и упрощение административных процессов. Она также предоставляет возможность интеграции новых функций в будущем, таких как телемедицина или расширенные аналитические инструменты.

В дальнейшем система может быть доработана с учетом обратной связи от пользователей, а также адаптирована для интеграции с другими медицинскими приложениями и платформами. Это позволит обеспечить еще более высокий уровень обслуживания и соответствовать современным требованиям здравоохранения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК ИСТОЧНИКОВ ИНФОРМАЦИИ

1. Хабр: C# [Электронный ресурс] – Режим доступа: https://habr.com/ru/hubs/csharp/articles (20.11.2024)
2. Metanit: Руководство по WPF [Электронный ресурс] – Режим доступа: https://metanit.com/sharp/wpf (07.11.2024)
3. Microsoft Learn: Документация по Windows Presentation Foundation [Электронный ресурс] – Режим доступа: https://learn.microsoft.com/ru-ru/dotnet/desktop/wpf (07.11.2024).
4. Microsoft Learn: Entity Framework 6 [Электронный ресурс] – Режим доступа: https://learn.microsoft.com/ru-ru/ef/ef6/ (05.12.2024).
5. OxyPlot: Welcome to OxyPlot’s documentation! [Электронный ресурс] – Режим доступа: https://oxyplot.readthedocs.io/en/latest (12.12.2024).
6. Stack Overflow: A question-and-answer website for computer programmers [Электронный ресурс] – Режим доступа: https://stackoverflow.com (18.11.2024).

Приложение А

(обязательное)

Техническое задание

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ БОЛЬНИЦЫ

Техническое задание

КП. 09.02.07. ИР41. 023.ТЗ

Листов 12

1 Требования к системе в целом

1.1 Требования к структуре и функционированию системы

В состав информационной системы для больницы должны входить следующие подсистемы:

1. Подсистема администратора
2. Подсистема регистратора
3. Подсистема врача
4. Подсистема медицинского персонала

Информационная система для больницы (в дальнейшем будем называть ее Система) должна быть организована централизованно, на одном компьютере. Режим функционирования – автоматизированный, диалоговый (элементы диалога – экранные формы).

Информация должна вводиться и поддерживаться в актуальном состоянии специалистом, сотрудниками больницы.

1.2 Требования к персоналу

Для функционирования и поддержания работоспособности Системы необходим специалист - техник по информационным системам (или информационным технологиям) – 1 ед. Техник по информационным системам должен иметь опыт администрирования СУБД SQLite, знание и понимание концепций реляционных баз данных.

1.3 Показатели назначения

Система должна обеспечивать возможность исторического хранения данных с глубиной не менее 10 лет.

Система должна обеспечивать возможность одновременной работы только одного пользователя.

Характеристики времени отклика Системы:

* для операций навигации по экранным формам системы – не более 1 сек;
* для операций формирования справок – не более 1 сек.

Время формирования аналитических отчетов определяется их сложностью и может занимать продолжительное время.

1.4 Требования к надёжности

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении следующих внештатных ситуаций:

* при сбоях в системе электроснабжения;
* при ошибках в работе аппаратных средств;
* при ошибках, связанных с системным программным обеспечением.

Потери данных в указанных случаях должны быть минимальными. Должна быть предусмотрена возможность восстановления данных из резервной копии.

1.5 Требования к безопасности

Система должна включать программные средства для ограничения прав доступа к ней.

1.6 Требования к эргономике

Система должна обеспечивать удобный для пользователей Системы интерфейс, отвечающий следующим требованиям:

* единый стиль оформления для пользовательских интерфейсов;
* должна быть удобная, интуитивно понятная навигация в интерфейсе пользователя;
* взаимодействие пользователя с Системой должно осуществляться на русском языке; исключения могут составлять только системные сообщения.

Пользовательские интерфейсы Системы должны быть спроектированы и разработаны с применением единых принципов графического представления информации и организации доступа к функциональным возможностям и сервисам. Должен быть разработан графический дизайн пользовательских интерфейсов, цветовые, шрифтовые и композиционные решения для отображения текстов, изображений, таблиц, гиперссылок, управляющих и навигационных элементов (меню, кнопок, форм и т.п.).

1.7 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы.

Система должна быть рассчитана на эксплуатацию в составе программно-технического комплекса Заказчика.

Техническая и физическая защита аппаратных компонентов системы, носителей данных, бесперебойное энергоснабжение, резервирование ресурсов, текущее обслуживание реализуется техническими и организационными средствами Заказчика.

При вводе системы в опытную эксплуатацию должен быть разработан план выполнения резервного копирования программного обеспечения и обрабатываемой информации. Во время эксплуатации системы, персонал, ответственный за эксплуатацию системы должен выполнять разработанный план.

Размещение помещений и их оборудование должны исключать возможность бесконтрольного проникновения в них посторонних лиц.

Размещение оборудования, технических средств должно соответствовать требованиям техники безопасности, санитарным нормам и требованиям пожарной безопасности.

Все пользователи системы должны соблюдать правила эксплуатации электронной вычислительной техники.

2 Требования к функциям Системы

В Системе должно быть предусмотрено выполнение следующих функций:

1. Смена цветовых тем из представленных в Системе.
2. Смена пользователя без необходимости выхода из Системы.
3. Учёт сотрудников:

* Ввод регистратором данных о сотрудниках.
* Назначение администратором паролей и должностей сотрудникам.
* При обнаружении ошибочных или измененных данных, изменение информации о сотрудниках.
* Удаление сотрудника в случае ошибочного добавления.

1. Учёт пациентов:

* Ввод регистратором данных о сотрудниках.
* При обнаружении ошибочных или измененных данных, изменение информации о пациентах.
* Удаление пациента в случае ошибочного добавления.

1. Ведение справочной информации о доступных препаратах и процедурах:

* При появлении нового препарата или процедуры в больнице добавление регистратором информации в справочник.
* При обнаружении ошибочных или измененных данных, изменение информации в справочнике.
* В случае завершения применения препарата или процедуры удаление информации из справочника.

1. Ведение справочной информации о доступных для сбора в больнице показателей состояния пациентов:

* При появлении показателя добавление регистратором информации в справочник.
* При обнаружении ошибочных или измененных данных, изменение информации в справочнике.
* В случае отсутствия возможности сбора показателя удаление информации из справочника.

1. Учёт кроватей для пациентов:

* Ввод, изменение, удаление регистратором данных о кроватях.
* Отслеживание свободных кроватей.

1. Учёт оформлений госпитализации:

* Ввод, изменение, удаление регистратором данных об оформлениях.
* Отслеживание количества текущих госпитализаций.

1. Просмотр врачом данных о доступных препаратах и процедурах.
2. Отслеживание врачом его текущих пациентов.
3. Учёт назначений препаратов и процедур:

* Ввод, удаление регистратором данных о назначениях.
* Автоматическая генерация расписания назначения с возможностью корректировки.

1. Учёт собираемых показателей у пациента:

* Выбор показателей, которые должны собираться у пациентов.
* Отмена сбора показателя в случае ошибочного добавления.

1. Учёт данных о состоянии пациентов:

* Ввод, изменение, удаление медицинским персоналом данных о значениях собираемых у пациента показателях.
* При обнаружении ошибочных или измененных данных, изменение записи состояния.
* Удаление записи состояния в случае ошибочного добавления.

1. Контроль выполнения расписания назначения медицинским персоналом:

* Отображение только уже произошедших приёмов препаратов или проведения процедур.
* Возможность изменения статуса расписания на выполнено или не выполнено.
* Возможность добавления, изменения дополнительной информации касательно приёма препарата или проведения процедуры.

1. Подготовка отчета о состояния пациента в виде графика.
2. Фильтрация записей на страницах Системы для упрощения поиска.

3 Требования к видам обеспечения

3.1 Требования к информационному обеспечению

Хранение и актуализация данных о сотрудниках, пациентах, препаратах, процедурах, показателях, оформлениях, состоянии пациентов, назначениях, расписаниях.

Система должна позволять вводить и корректировать следующие данные о сотрудниках в справочник сотрудниках:

* Полное имя сотрудника;
* Пароль;
* Должность;

Система должна позволять вводить и корректировать следующие данные о пациентах:

* Полное имя;
* Дату рождения;
* Пол;
* Номер полиса обязательного медицинского страхования.

Система должна позволять вводить и корректировать следующие данные о кроватях:

* Номер (название) палаты;
* Номер кровати в палате;
* Тип кровати;

Система должна позволять вводить и корректировать следующие данные о показателях:

* Наименование;
* Описание;

Система должна позволять вводить и корректировать следующие данные о препаратах и процедурах:

* Название;
* Описание;

Система должна позволять вводить и корректировать следующие данные об оформлении госпитализации:

* Пациента;
* Причину госпитализации;
* Дату госпитализации;
* Дату выписки;
* Лечащего врача;
* Кровать;

Система должна позволять вводить и корректировать следующие данные о состоянии пациента:

* Пациента;
* Показатель;
* Значение показателя
* Примечание;

Система должна позволять вводить и корректировать следующие данные о назначениях:

* Пациент;
* Препарат или процедура;
* Дополнительная информация;

Система должна позволять вводить и корректировать следующие данные о расписании:

* Назначение;
* Дата и время;
* Примечание;
* Статус;

3.2 Требования к программному обеспечению

Для управления Системой должна использоваться операционная система Microsoft Windows 10 и выше.

Прикладное программное обеспечение в составе Системы должно соответствовать следующим основным требованиям:

* функционировать в среде операционной системы и взаимодействовать с СУБД в соответствии с требованиями настоящего ТЗ;
* поддерживать русский и английский языки, символы кириллицы и латиницы;
* иметь удобный пользовательский интерфейс;
* обеспечивать реализацию всех функций Системы в соответствии с требованиями настоящего ТЗ;
* иметь комплект пользовательской документации на русском языке.

Качество разработки программных средств должно обеспечиваться соответствующими процедурами управления проектом по реализации Системы.

3.3 Требования к техническому обеспечению

Состав технические средства, функционирующие в составе Системы:

1. Персональный компьютер

* Процессор – на базе архитектуры x64, с частотой не менее 2000MHz;
* Объем оперативной памяти – не менее 1 Гб;
* Дисковая память – не менее 40 Гб;

1. Цветной монитор, размер не менее 15”, разрешение не менее 1024х768 точек;
2. Клавиатура, мышь.

3.4 Требования к организационному обеспечению

В целях обеспечения штатного функционирования Системы необходимо наличие подразделений, производящих техническое и программное обслуживание компонент Системы. Требуется наличие должностных инструкций, регламентирующих порядок использования Системы и разграничивающих права ее использования. Должно быть разработано руководство пользователя системы.

# Приложение Б

(обязательное)

Руководство пользователя

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ БОЛЬНИЦЫ

Руководство пользователя

КП. 09.02.07. ИР41. 023.34

Листов 31

АННОТАЦИЯ

В данном документе приведено руководство оператора информационной системы для больницы.

В разделе «Назначение программы» указаны сведения о назначении программы и информация, достаточная для понимания функций программы и ее эксплуатации.

В разделе «Условия выполнения программы» указаны условия, необходимые для выполнения программы.

В разделе «Выполнение программы» указаны последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы, приведены описание функций, формата и возможных вариантов команд, с помощью которых оператор осуществляет загрузку и управляет выполнением программы, а также ответы программы на эти команды.

1. НАЗНАЧЕНИЕ ПРОГРАММЫ

Информационная система для больницы предназначена для обеспечения хранения, накопления и предоставления всей необходимой информации о сотрудниках, пациентах, показателях, препаратах, процедурах, кроватях, оформлениях, состоянии пациентов, назначениях и расписании назначений.

Данная информационная система эксплуатируется в больнице.

Основными функциями информационной системы являются:

* оперативный доступ к имеющимся данным;
* ввод, хранение, изменение и удаление информации о сотрудниках, пациентах, препаратах, процедурах, кроватях, оформлениях, состоянии пациентов, назначениях и расписании назначений;
* фильтрация информации;
* сортировка информации по убыванию и возрастанию;
* поиск оформлений;

Конечными пользователями информационной системы являются администраторы, регистраторы, врачи и медицинский персонал.

1. Условия выполнения программы

Минимальные требования к рабочей станции (ПК):

* операционная система Windows 10 и выше;
* процессор с тактовой частотой 2000 МГц и выше;
* оперативная память 1 Гб и выше;
* жесткий диск 40 Гб и выше;
* видеокарта, поддерживающая монитор с минимальным разрешением точек;
* устройства ввода информации: клавиатура, мышь.

1. ВЫПОЛНЕНИЕ ПРОГРАММЫ

3.1 Общие сведения

1. Авторизация

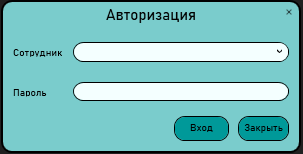


Рисунок Б1 – Окно авторизации

* Введите полное имя пользователя или выберите из списка.
* Введите пароль, полученный от администратора.
* Нажмите кнопку «Вход».

1. Открытие справки

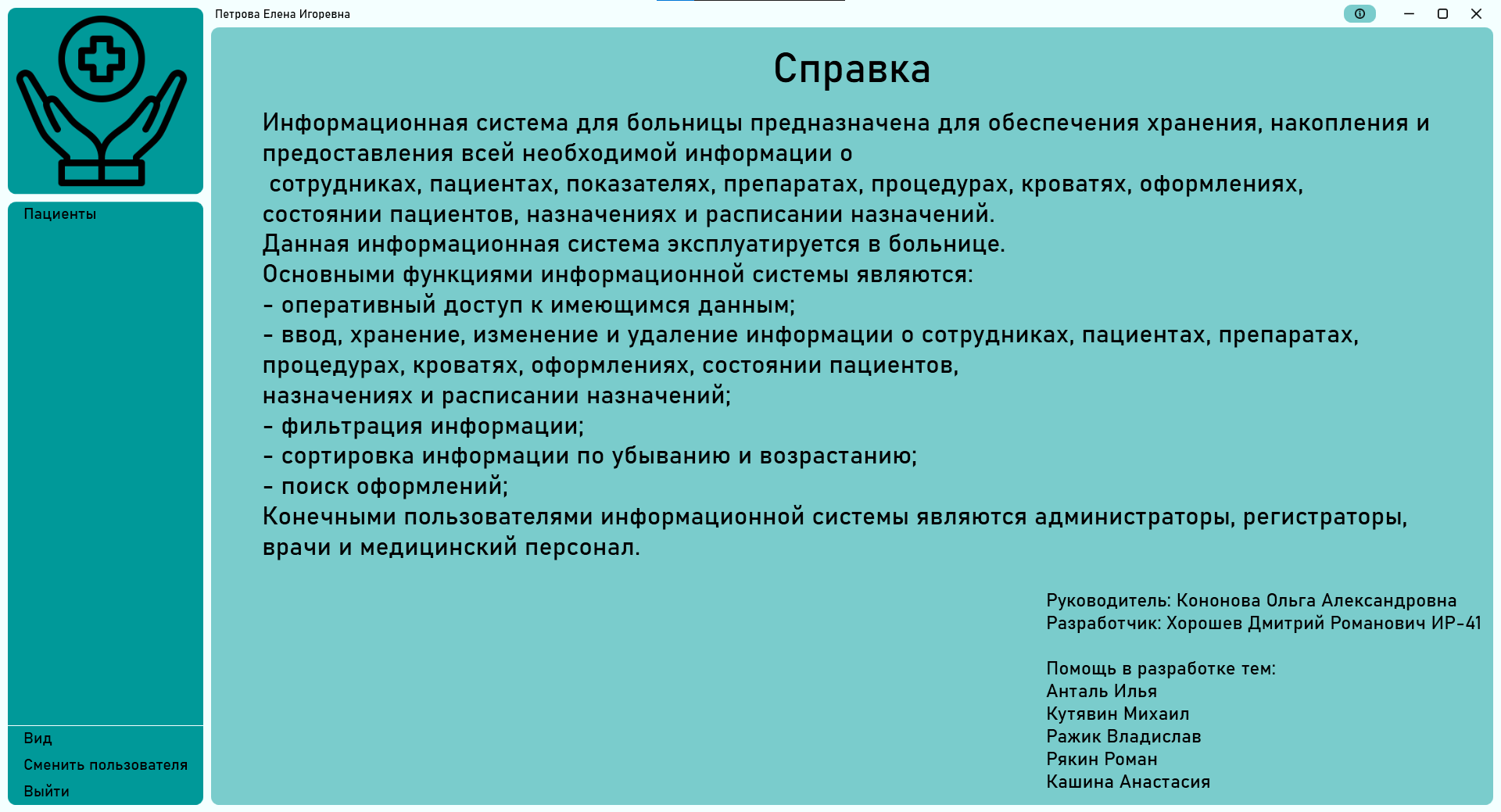


Рисунок Б2 – Страница «Справка»

* Нажмите на пиктограмму информации в правой верхней части окна.

1. Переход между страницами

* Нажмите соответствующий пункт меню в левой части окна.

1. Управление размерами и состоянием окна

* Для закрытия окна нажмите на пиктограмму крестика в правом верхнем углу или нажмите пункт меню «Выйти» в левом нижнем углу окна и подтвердите выход, нажав на кнопку «Да».
* Для изменения размера окна наведите курсор на край окна, зажмите левую кнопку мыши и потяните в нужную сторону.
* Для максимизации или возврата к не максимизированному размеру окна нажмите на пиктограмму квадрата в правом верхнем углу.
* Для сворачивания окна нажмите на пиктограмму горизонтальной полоски в правом верхнем углу.
* Для перемещения окна зажмите левую кнопку мыши у верхнего края окна и потяните в нужную сторону.

1. Смена пользователя

* Нажмите на пункт «Сменить пользователя» в левом нижнем углу.
* Авторизуйтесь в окне авторизации.

1. Изменение внешнего вида

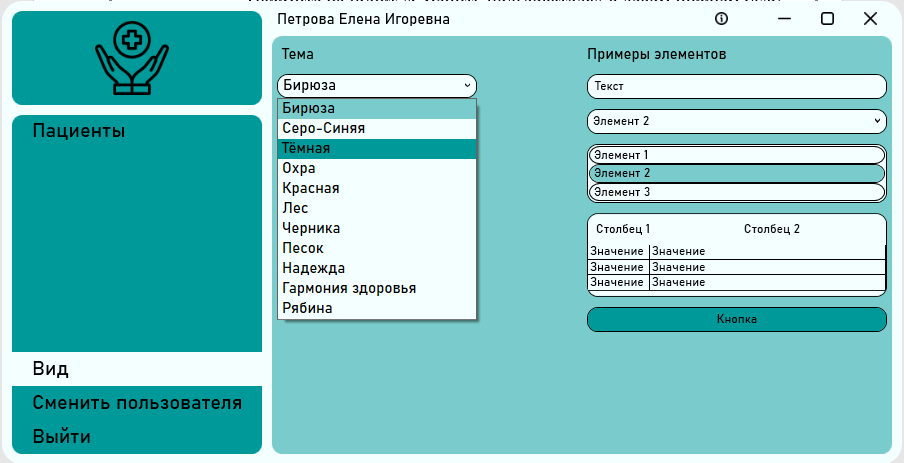


Рисунок Б3 – Страница «Вид»

* Нажмите на пункт меню «Вид».
* Раскройте список «Тема».
* Выберите тему из списка предложенных.

3.2 Администратор

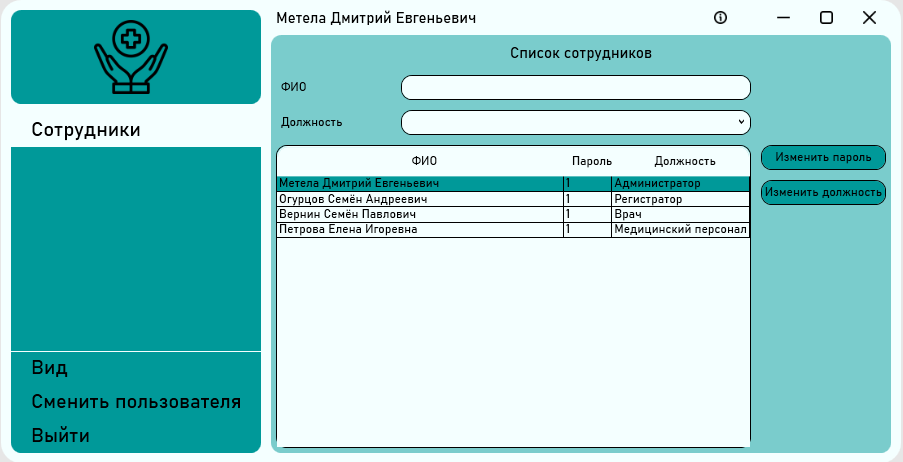


Рисунок Б4 – Страница «Сотрудники» у администратора

1. Поиск и фильтрация сотрудников

* Для поиска сотрудника по полному имени нажмите на текстовое поле «ФИО» и начните вводить фамилию, имя или отчество сотрудника (Рисунок Б4).
* Для фильтрации под должности нажмите на поле списка «Должность» и выберите должность из выпадающего списка (Рисунок Б4).

1. Изменение пароля или должности сотрудника

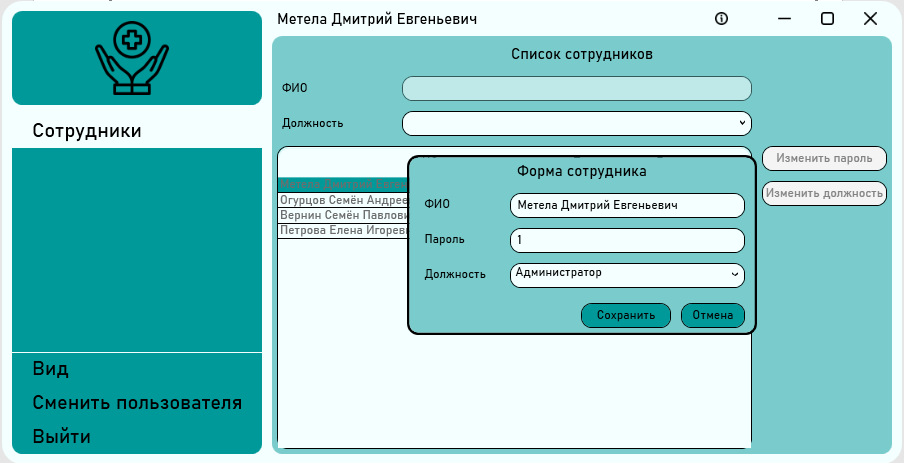


Рисунок Б5 – Форма сотрудника у администратора

* Нажмите на строку таблицы с сотрудником, чьи данные вы хотите изменить.
* Нажмите на кнопку «Изменить пароль» для изменения пароля или на кнопку «Изменить должность» для изменения должности.
* В открывшейся форме введите новый пароль или выберите из списка нужную должность.
* Нажмите кнопку «Сохранить» для сохранения изменений.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

3.3 Регистратор

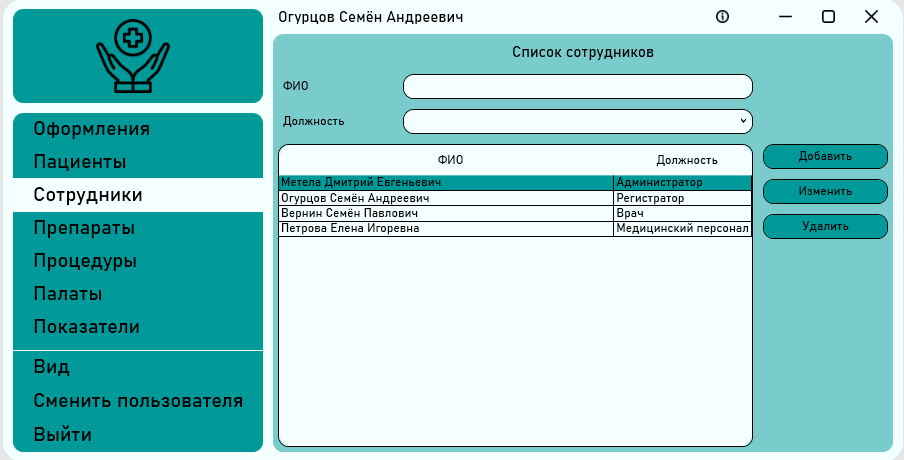


Рисунок Б6 – Страница «Сотрудники» у регистратора

1. Поиск и фильтрация сотрудников – аналогично поиску и фильтрации сотрудников (Пункт 7)

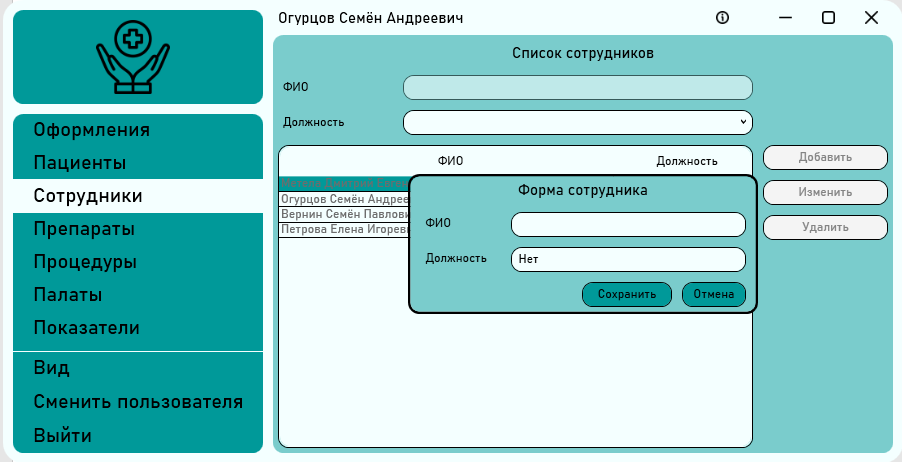


Рисунок Б7 – Форма сотрудника у регистратора

1. Добавление сотрудника

* Нажмите на кнопку «Добавить».
* В открывшейся форме нажмите на текстовое поле «ФИО».
* Введите полное имя сотрудника.
* Нажмите на кнопку «Сохранить» для сохранения нового сотрудника.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

1. Изменение сотрудника

* Нажмите на строку таблицы с сотрудником, чьи данные вы хотите изменить.
* Нажмите на кнопку «Изменить».
* В открывшейся форме нажмите на текстовое поле «ФИО».
* Исправьте полное имя сотрудника.
* Нажмите на кнопку «Сохранить» для сохранения изменений.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

1. Удаление сотрудника

* Нажмите на строку таблицы с сотрудником, чьи данные вы хотите изменить.
* Нажмите на кнопку «Удалить».
* Подтвердите удаление нажав на кнопку «Да».



Рисунок Б8 – Страница «Пациенты» у регистратора

1. Поиск пациентов – аналогично поиску сотрудников (Пункт 7)
2. Добавление пациента

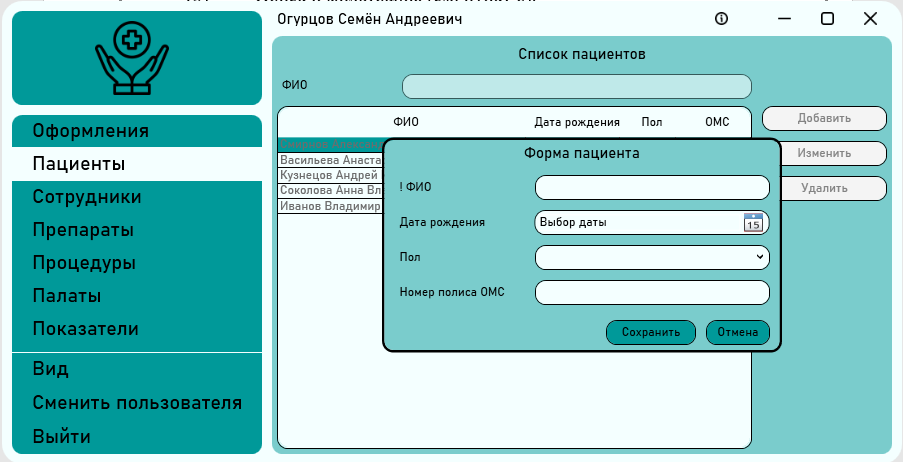


Рисунок Б9 – Форма пациента

* Нажмите на кнопку «Добавить».
* Нажмите на текстовое поле «ФИО».
* Введите полное имя пациента или уникальный номер, если оно неизвестно.
* Переходя между текстовыми полями, с помощью нажатий левой кнопки мыши или клавиши Tab, заполните известную информацию о пациенте.
* Нажмите на кнопку «Сохранить» для сохранения изменений.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

1. Изменение пациента

* Нажмите на строку таблицы с пациентом, чьи данные вы хотите изменить.
* Нажмите на кнопку «Изменить».
* В открывшейся форме, переходя между текстовыми полями, с помощью нажатий левой кнопки мыши или клавиши Tab, измените информацию о пациенте.
* Нажмите на кнопку «Сохранить» для сохранения изменений.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

1. Удаление пациента – аналогично удалению сотрудника (Пункт 12)

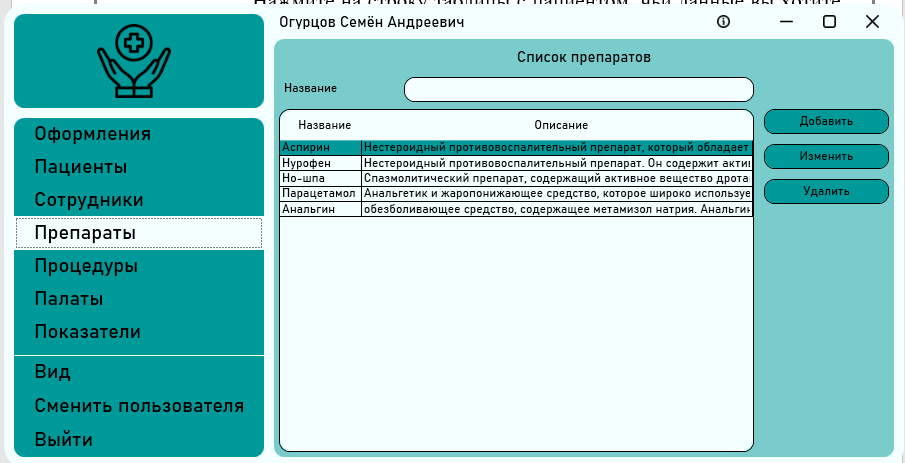


Рисунок Б10 – Страница «Препараты»

1. Поиск препаратов – аналогично поиску сотрудников (Пункт 7)
2. Добавление препарата

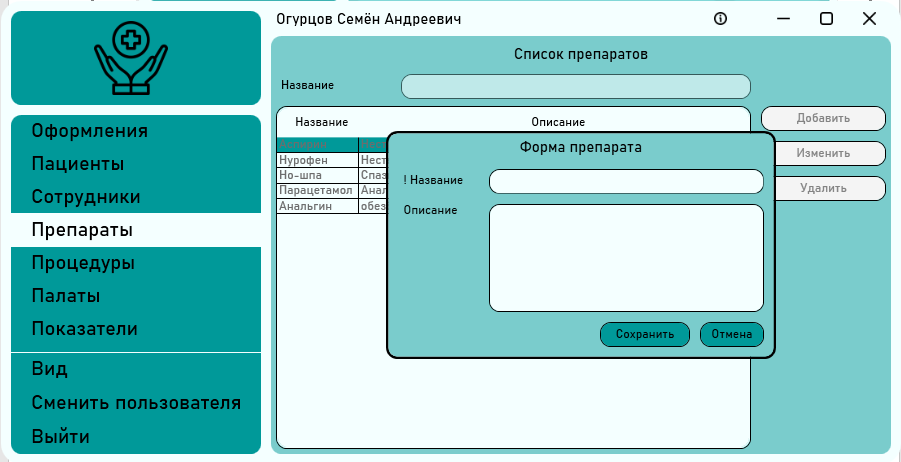


Рисунок Б11 – Форма препарата

* Нажмите на текстовое поле «Название».
* Введите название препарата.
* Если необходимо добавить описание препарата, нажмите на тестовое поле «Описание» и введите описание препарата используя клавишу Enter для ручного переноса строки.
* Нажмите на кнопку «Сохранить» для сохранения изменений.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

1. Изменение препарата

* Нажмите на строку таблицы с препаратом, данные которого вы хотите изменить.
* Далее аналогично добавлению препарата (Пункт 18).

1. Удаление препарата – аналогично удалению сотрудника (Пункт 12)
2. Поиск процедуры – аналогично поиску сотрудников (Пункт 7)
3. Добавление процедуры – аналогично добавлению препарата (Пункт 18)
4. Изменение процедуры – аналогично изменению препарата (Пункт 19)
5. Удаление процедуры – аналогично удалению сотрудника (Пункт 12)

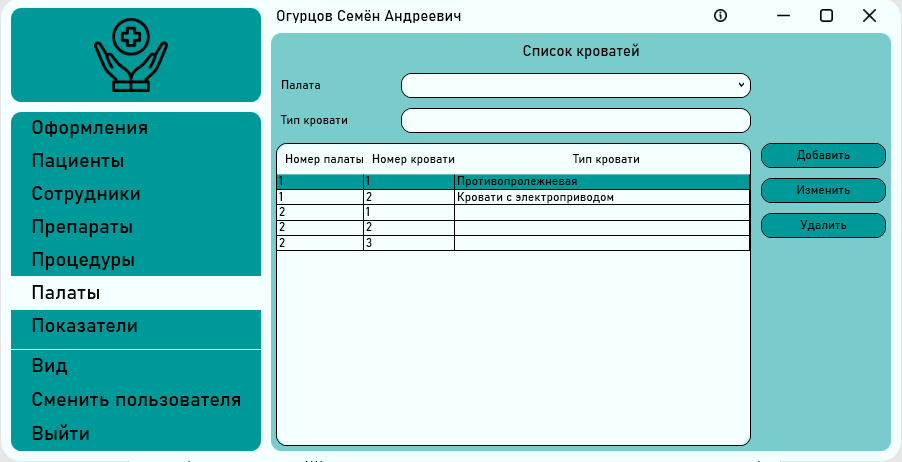


Рисунок Б12 – Страница «Палаты»

1. Поиск и фильтрация палат – аналогично поиску и фильтрации сотрудников (Пункт 7)
2. Добавление кровати

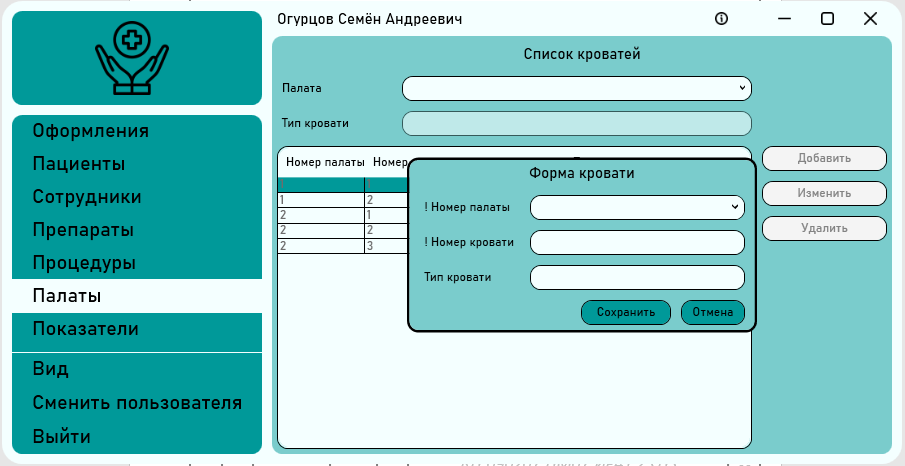


Рисунок Б13 – Форма кровати

* Нажмите на кнопку «Добавить».
* Нажмите на поле «Номер палаты».
* Введите или выберите из списка палату.
* Нажмите на тестовое поле «Номер кровати».
* Введите номер кровати.
* Если необходимо перейдите на текстовое поле «Тип кровати» и введите тип кровати.
* Нажмите кнопку «Сохранить» для сохранения новой кровати.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

1. Изменение кровати

* Нажмите на строку таблицы с кроватью, данные которой вы хотите изменить.
* Далее аналогично добавлению кровати (Пункт 26)

1. Удаление кровати – аналогично удалению сотрудника (Пункт 12)
2. Поиск показателя – аналогично поиску сотрудников (Пункт 7)
3. Добавление показателя – аналогично добавлению препарата (Пункт 18)
4. Изменение показателя – аналогично изменению препарата (Пункт 19)
5. Удаление показателя – аналогично удалению сотрудника (Пункт 12)

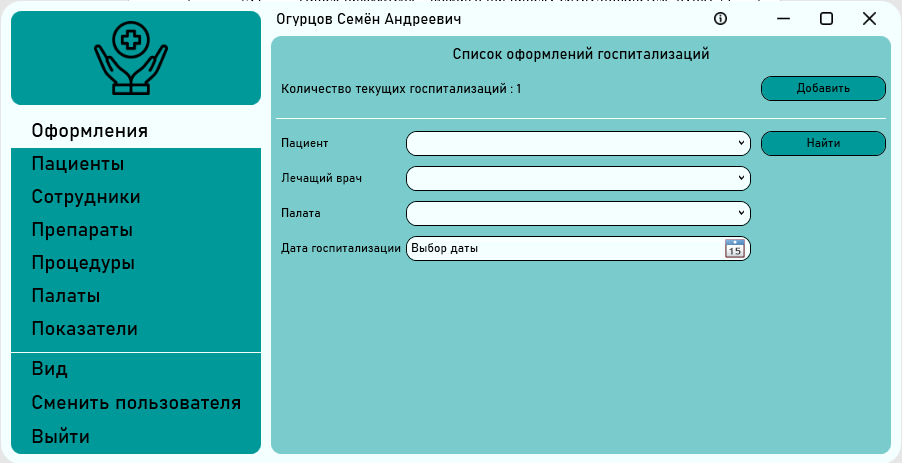


Рисунок Б14 – Страница «Оформления»

1. Добавление оформления

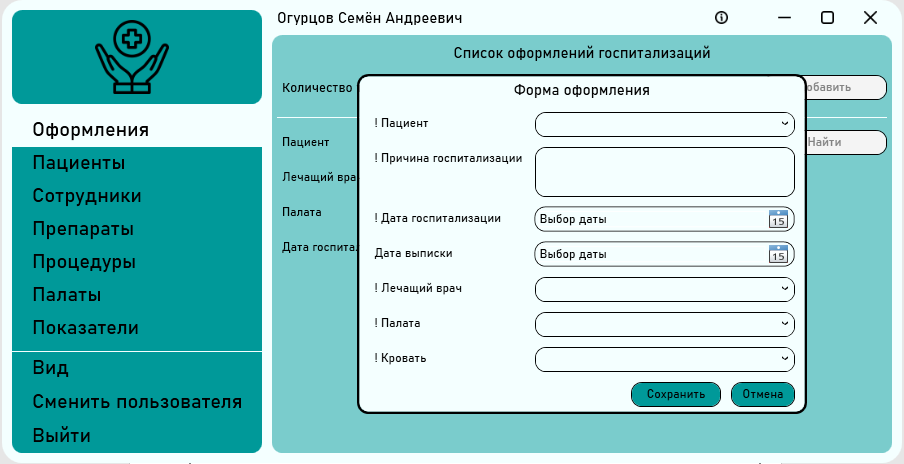


Рисунок Б15 – Форма оформления

* Нажать на кнопку «Добавить».
* Переходя между полями формы оформления (Рисунок Б15), заполните поля вводом данных или выбором из списка.
* Нажмите кнопку «Сохранить» для сохранения нового оформления.
* Нажмите на кнопку «Отмена» для закрытия формы без сохранения.

1. Поиск оформления

* Переходя между полями на странице «Оформления» (Рисунок Б14) заполнить их информацией об искомом оформлении.
* Нажать на кнопку «Найти».

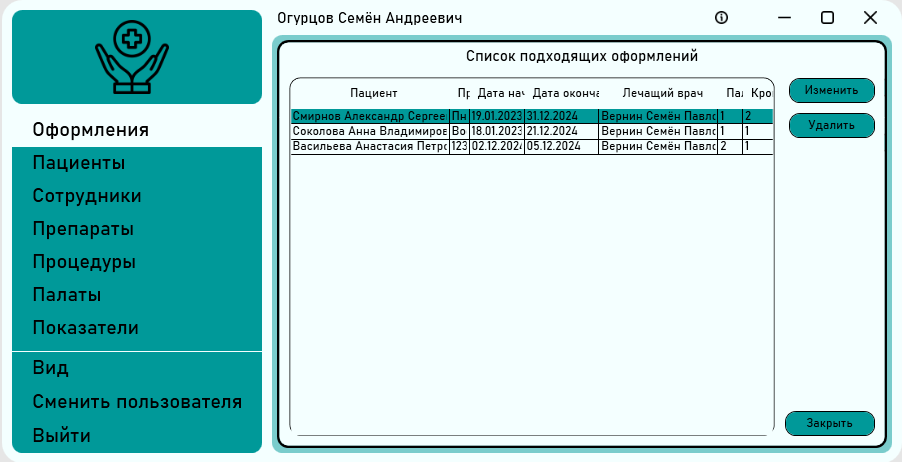


Рисунок Б16 – Форма списка подходящих оформлений

1. Изменение оформления

* Произведите поиск оформления, которое вы хотите изменить (Рисунок Б16).
* Нажмите на строку таблицы с оформлением, данные которого вы хотите изменить.
* Далее аналогично добавлению кровати (Пункт 26)

1. Удаление оформления – аналогично удалению сотрудника (Пункт 12)

3.4 Врач

1. Просмотр данных о препаратах и процедурах

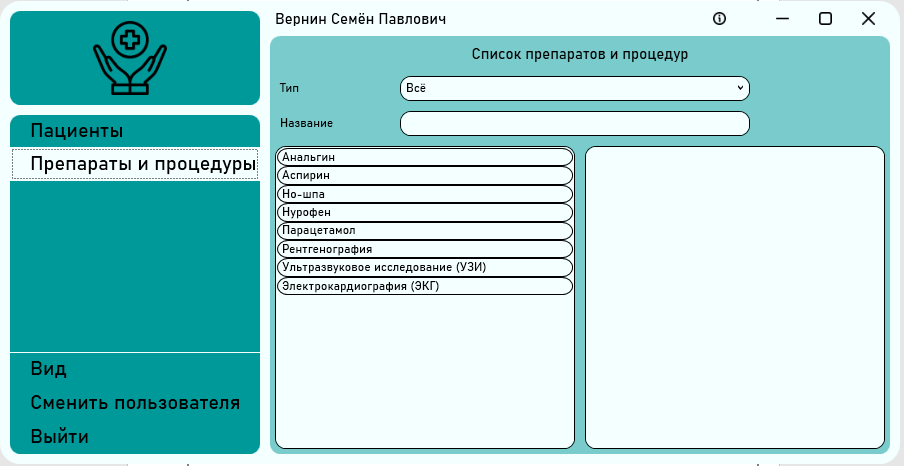


Рисунок Б17 – Страница «Препараты и процедуры»

* Нажмите на строку списка с препаратами и процедурами, описание которых вы хотите просмотреть.
* Описание появиться в поле справа.

1. Поиск и фильтрация препаратов и процедур – аналогично поиску и фильтрации сотрудников (Пункт 7)

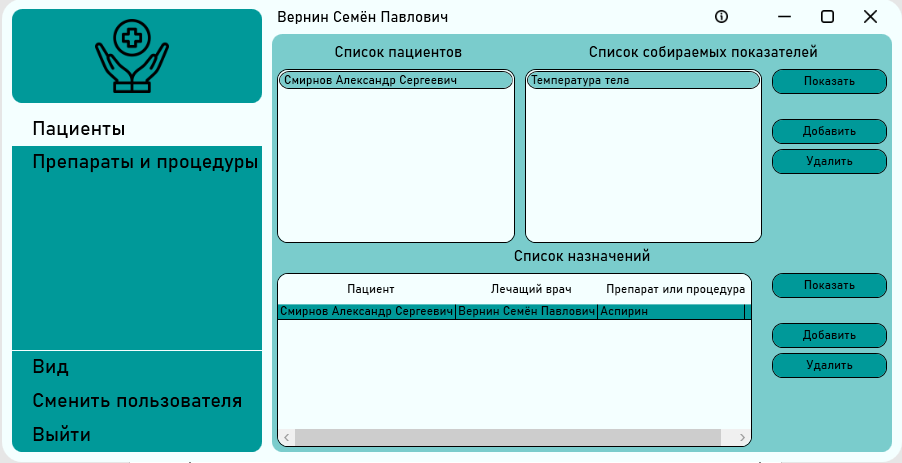


Рисунок Б18 – Страница «Пациенты» у врача

1. Просмотр отчёта-графика о состоянии пациента

* Нажмите на строку списка пациентов, состояние которого вы хотите просмотреть.
* Нажмите на строку списка собираемых показателей, значения которого вы хотите просмотреть.
* Нажмите на верхнюю кнопку «Показать»

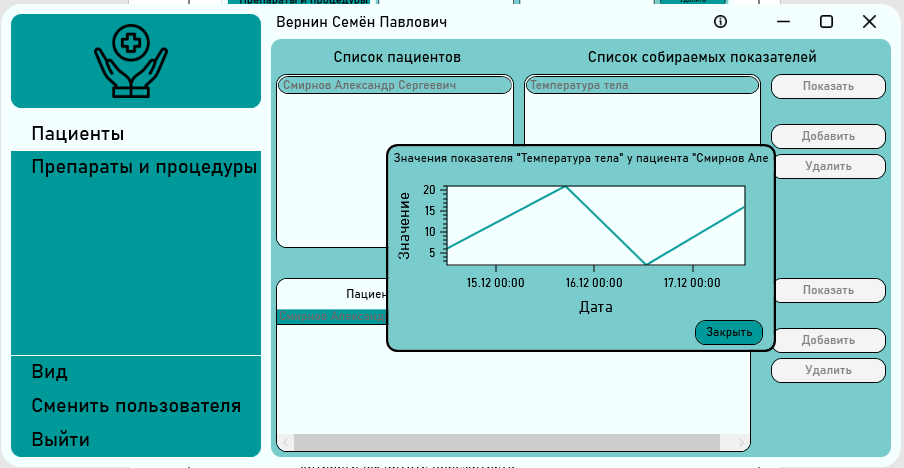


Рисунок Б19 – Форма отчёта-графика о состоянии пациента

* Прокрутите колёсико мыши, наведя его на одну из осей, чтобы изменить её масштаб.
* Прокрутите колёсико мыши, наведя его на область графика, чтобы изменить его масштаб.
* Зажмите правую кнопку мыши в области графика и потяните мышь, чтобы подвинуть график.
* Зажмите левую кнопку мыши в области графика, чтобы увидеть точное значение графика в точке курсора.
* Нажмите на кнопку «Закрыть», чтобы закрыть отчёт.

1. Добавление собираемого показателя.

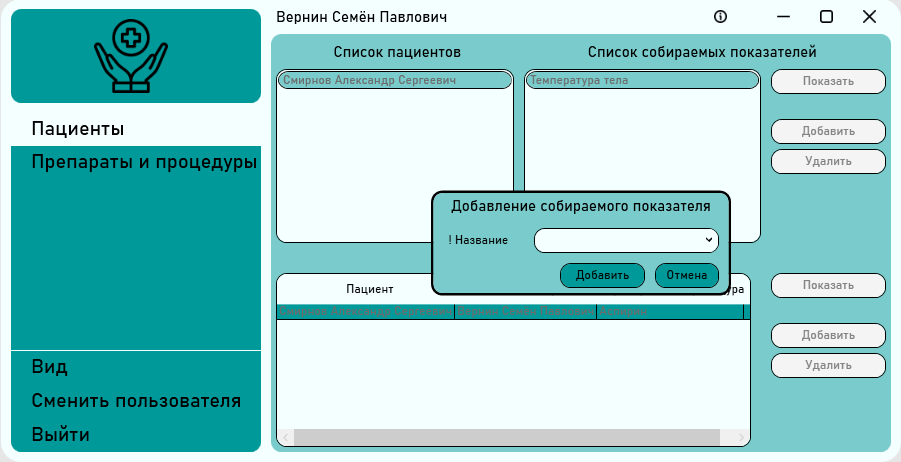


Рисунок Б19 – Форма добавления собираемого показателя

* Нажмите на строку списка пациентов, которому вы хотите добавить собираемый показатель.
* Нажмите верхнюю кнопку «Добавить».
* Нажмите на поле «Название».
* Выберите значение из списка.
* Нажмите кнопку «Добавить», чтобы добавить собираемый показатель.
* Нажмите на кнопку «Отмена» для закрытия формы.

1. Удаление собираемого показателя

* Нажмите на строку списка пациентов, собираемый показатель которого вы хотите удалить.
* Нажмите на строку списка собираемых показателей, который вы хотите удалить.
* Нажмите на верхнюю кнопку «Удалить».
* Подтвердите удаление нажав на кнопку «Да».

1. Добавление назначения

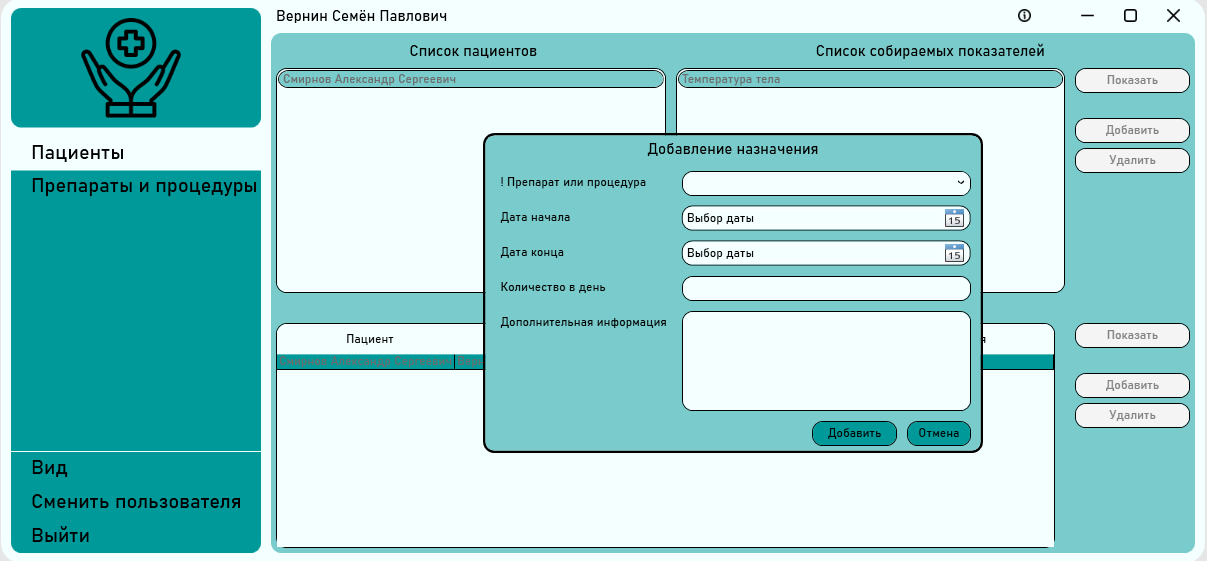


Рисунок Б20 – Форма добавления назначения

* Нажмите на строку списка пациентов, которому вы хотите добавить назначение.
* Нажмите нижнюю кнопку «Добавить».
* Переходя между текстовыми полями, с помощью нажатий левой кнопки мыши или клавиши Tab, заполните информацию о назначении.
* Нажмите кнопку «Добавить», чтобы добавить назначение.
* Нажмите на кнопку «Отмена» для закрытия формы.

1. Удаление назначения

* Нажмите на строку списка пациентов, назначение которого вы хотите удалить.
* Нажмите на строку таблицы назначений, которую вы хотите удалить.
* Нажмите на нижнюю кнопку «Удалить».
* Подтвердите удаление нажав на кнопку «Да».

1. Просмотр расписания назначения

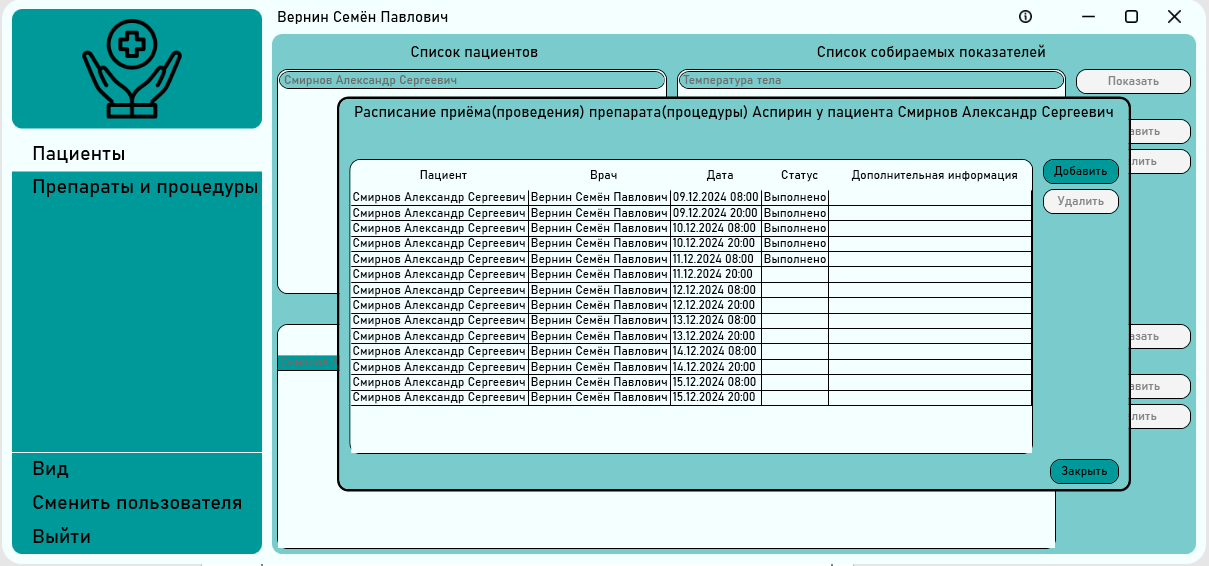


Рисунок Б21 – Форма просмотра расписания назначения

* Нажмите на строку списка пациентов, у которого вы хотите просмотреть расписание назначения.
* Нажмите на строку таблицы назначений, расписание которого вы хотите просмотреть.
* Нажмите на нижнюю кнопку «Показать»
* Нажмите на кнопку «Добавить», чтобы добавить новую запись расписания (Пункт 45).
* Нажмите на кнопку «Закрыть», чтобы закрыть форму.

1. Добавление записи расписания

* Просмотреть расписание нужного назначения (Пункт 44).

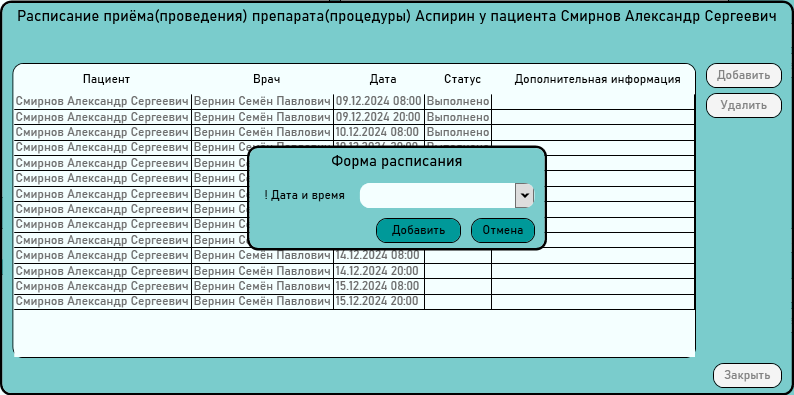


Рисунок Б22 – Форма добавления записи расписания назначения

* Нажать на поле «Дата и время».
* Выбрать или ввести значение.
* Нажмите кнопку «Добавить», чтобы добавить запись расписания назначения.
* Нажмите на кнопку «Отмена» для закрытия формы без добавления.

3.5 Медицинский персонал

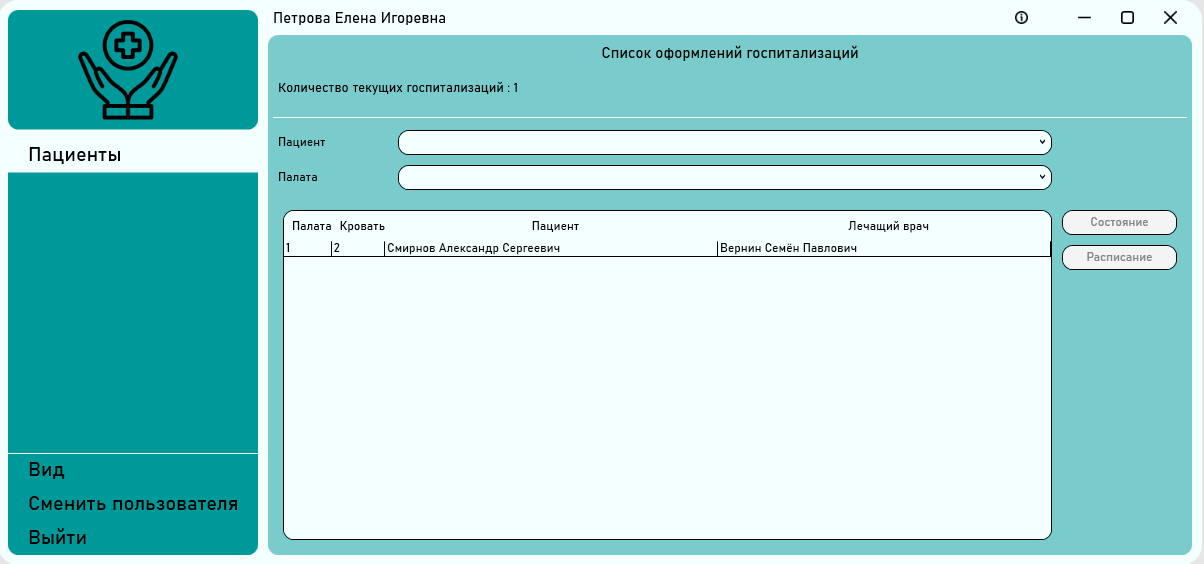


Рисунок Б23 – Страница «Пациенты» у медицинского персонала

1. Фильтрация пациентов – аналогично фильтрации сотрудников (Пункт 7)

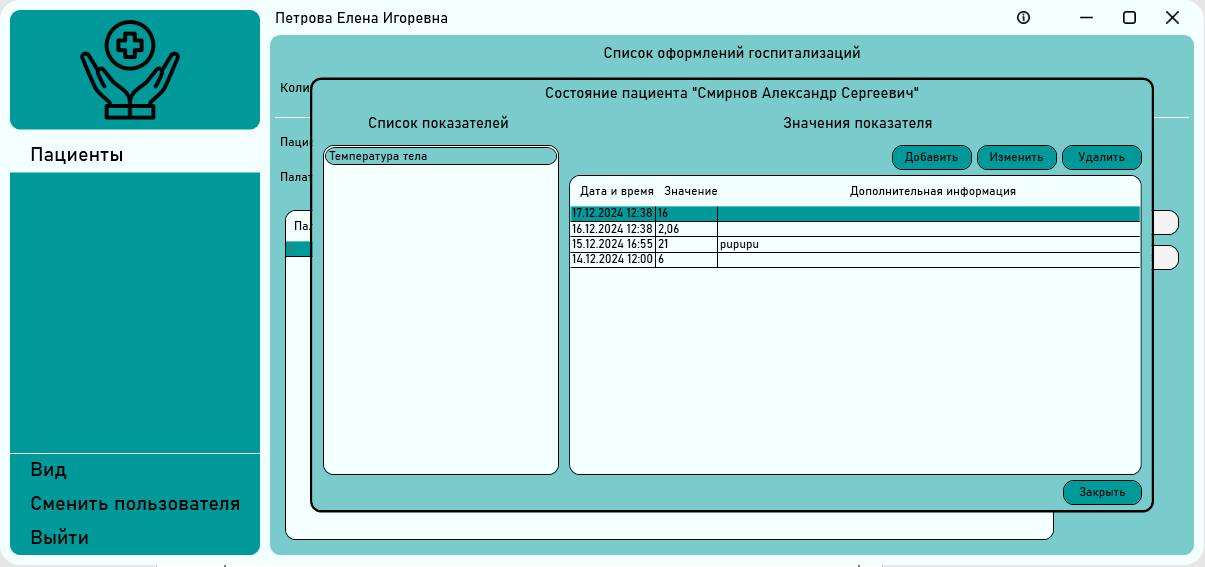


Рисунок Б24 – Форма состояния пациента

1. Добавление значения показателя состояние пациента

* Нажмите на строку таблицы пациентов, которому вы хотите добавить запись о состоянии.
* Нажмите на кнопку «Состояние»
* Нажмите на строку списка показателей, для которого вы хотите добавить запись о значении.
* Нажмите на кнопку «Добавить».

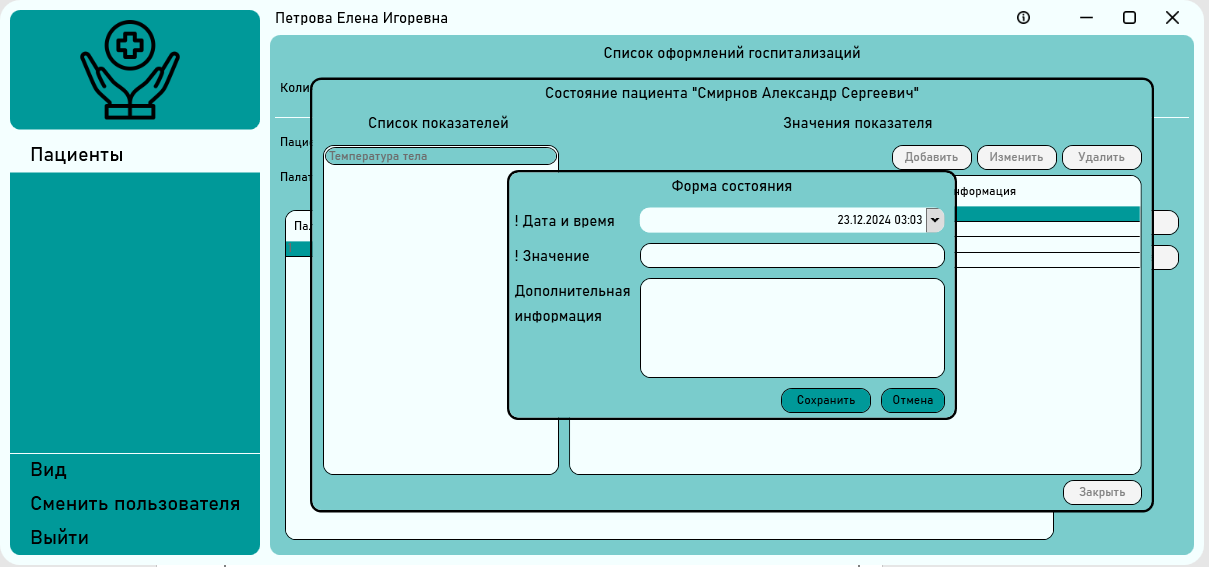


Рисунок Б24 – Форма добавления значения показателя

* Переходя между текстовыми полями, с помощью нажатий левой кнопки мыши или клавиши Tab, заполните информацию о значении показателя состояния пациента.
* Нажмите кнопку «Сохранить», чтобы добавить запись значения показателя состояния пациента.
* Нажмите на кнопку «Отмена» для закрытия формы без добавления.
* Нажмите на кнопку «Закрыть», чтобы закрыть форму.

1. Изменение значения показателя состояние пациента

* Нажмите на строку таблицы пациентов, которому вы хотите добавить запись о состоянии.
* Нажмите на кнопку «Состояние»
* Нажмите на строку списка показателей, для которого вы хотите добавить запись о значении.
* Нажмите на строку таблицы значений показателей, которую вы хотите изменить.
* Нажмите на кнопку «Изменить».
* Переходя между текстовыми полями, с помощью нажатий левой кнопки мыши или клавиши Tab, измените информацию о значении показателя состояния пациента.
* Нажмите кнопку «Сохранить», чтобы добавить запись значения показателя состояния пациента.
* Нажмите на кнопку «Отмена» для закрытия формы без добавления.
* Нажмите на кнопку «Закрыть», чтобы закрыть форму.

1. Изменение значения показателя состояние пациента

* Нажмите на строку таблицы пациентов, которому вы хотите добавить запись о состоянии.
* Нажмите на кнопку «Состояние»
* Нажмите на строку списка показателей, для которого вы хотите добавить запись о значении.
* Нажмите на строку таблицы значений показателей, которую вы хотите изменить.
* Нажмите на кнопку «Удалить».
* Подтвердите удаление нажав на кнопку «Да».
* Нажмите на кнопку «Закрыть», чтобы закрыть форму.

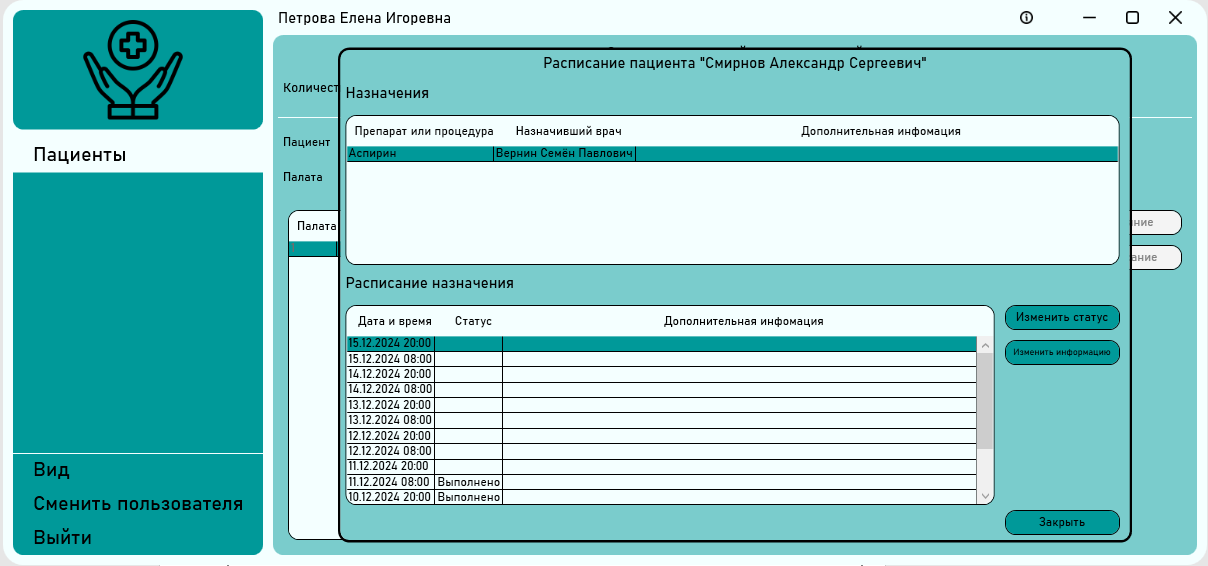


Рисунок Б25 – Форма расписания назначения

1. Изменение статуса записи расписания

* Нажмите на строку таблицы пациентов, которому вы хотите изменить статус записи расписания.
* Нажмите на кнопку «Расписание»
* Нажмите на строку списка назначений, у которого вы хотите изменить статус у записи расписания.
* Нажмите на строку таблицы записей расписания, у которой вы хотите изменить статус.
* Нажмите на кнопку «Изменить статус».
* Статус поменяется на противоположный (пустое значение приравнивается к статусу «Не выполнено»).
* Нажмите на кнопку «Закрыть», чтобы закрыть форму.

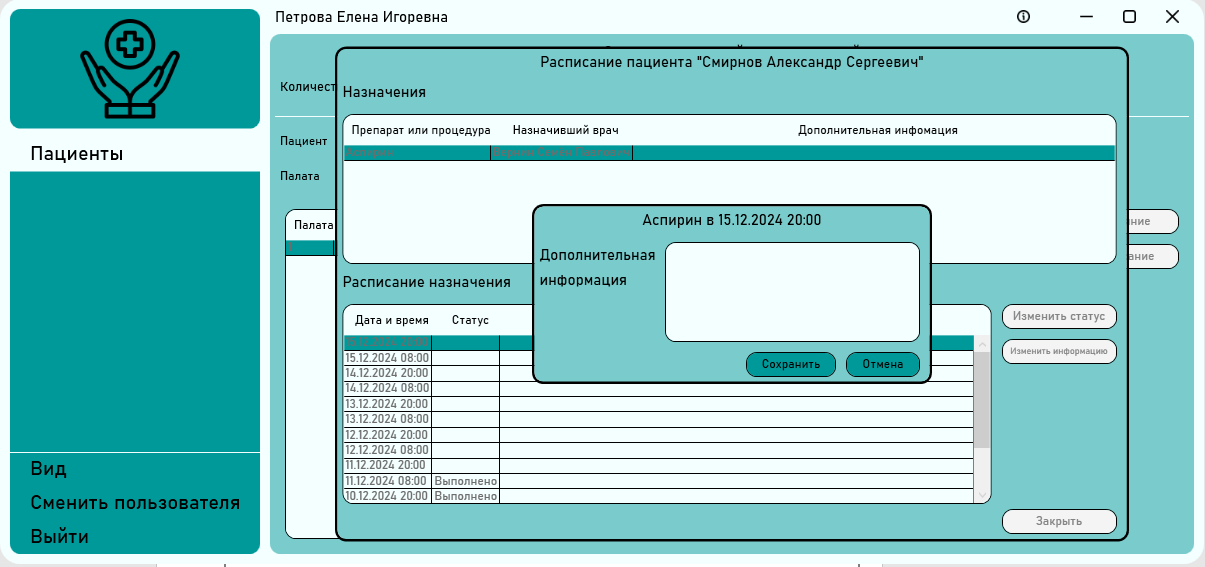


Рисунок Б26 – Форма добавления информации для записи расписания назначения

1. Добавление или изменения дополнительной информации для записи расписания

* Нажмите на строку таблицы пациентов, которому вы хотите изменить статус записи расписания.
* Нажмите на кнопку «Расписание»
* Нажмите на строку списка назначений, у которого вы хотите изменить статус у записи расписания.
* Нажмите на строку таблицы записей расписания, у которой вы хотите изменить статус.
* Нажмите на кнопку «Изменить информацию».
* В открывшейся форме нажать на текстовое поле «Дополнительная информация» (Рисунок Б26).
* Ввести нужную информацию.
* Нажмите кнопку «Сохранить», чтобы добавить запись значения показателя состояния пациента.
* Нажмите на кнопку «Отмена» для закрытия формы.
* Нажмите на кнопку «Закрыть», чтобы закрыть форму.