

Министерство образования и молодежной политики  
Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Специальность 09.02.07: «Информационные системы и  
программирование»

Разработка информационной системы «Подсистемы для CRM системы  
Медицинского центра»

**Пояснительная записка**

к курсовому проекту

КР-ПР-32-09-2023-ПЗ

Разработал:

Студент гр. ПР-32

\_\_\_\_\_/А.А. Панин

Руководитель

\_\_\_\_\_/Д.О. Гариев

2023

Министерство образования и молодежной политики  
Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Специальность 09.02.07: «Информационные системы и  
программирование»

Разработка информационной системы «Разработка подсистем для CRM  
системы мед. центра»

**Курсовой проект**

КП-ПР-32-09-2023

2023

## ОГЛАВЛЕНИЕ

Введение.....	4
1 Постановка задачи.....	5
1.1 Описание предметной области .....	5
1.2 Требования к программному продукту .....	6
2 Проектирование логической структуры .....	10
3 Техническое проектирование .....	18
3.1 Выбор состава технических программных средств .....	18
3.2 Физическая структура программы .....	19
4 Тестирование .....	22
Заключение .....	23
Список использованной литературы.....	24
Приложение – сценарий тестирования .....	25

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p align="center"><b>КП-ПР-32-09-2023-ПЗ</b></p> <p align="center"><i>Разработка информационной системы «Разработка подсистем для CRM системы мед. центра»</i></p>	Лит	Лист	Листов
	Лит	Изм.	№ докум.	Подп.	Дата			
	Разраб.		Панин А.А.			<p align="center"><b>ЕКТС</b></p>		
	Пров.		Гариев Д.О.					
	Н. контр.		Гариев Д.О.					
	Утв.							

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

“Ваш Доктор” – Это многопрофильная медицинская организация, которая предоставляет услуги в области лечения заболеваний опорно-двигательного аппарата, неврологии, кардиологии, гинекологии и других областях медицины. Для эффективного, удобного и прибыльного ведения бизнеса будут проанализированы устройство и процессы клиники в результате чего буду выявлены все необходимые возможности внедряемой CRM-системы.

# 1 ПОСТАНОВКА ЗАДАЧИ

## 1.1 Описание предметной области

Клиника семейной медицины «Ваш доктор» — это частный медицинский центр для всей семьи, основанный в 2011 году. В данной клинике работают опытные доктора – терапевты, узкие специалисты, диагносты.

Направления клиники:

- детское отделение. Заботимся о здоровье ребенка от зачатия до совершеннолетия. В клинике принимают педиатры, детский невролог и хирург-ортопед;
- взрослое отделение. Первичная консультация и терапия, а также лечение у узких специалистов - кардиолога и гастроэнтеролога;
- диагностика. Качественная диагностика – первый шаг на пути к выздоровлению и крепкому здоровью;
- косметология. Аппаратная и инъекционная косметология;
- медосмотры;
- физиотерапия.

Записаться на прием можно как по телефону, так и онлайн с помощью сайта клиники.

Перед приемом пациента заносят в базу, а если он уже есть в базе, то его записывают на прием.

На данный момент в клинике используется система для управления взаимоотношения с клиентами (CRM), позволяющая обрабатывать данные о врачах, пациентах, приеме и их лечении. В данной системе можно отслеживать всю информацию о пациентах, врачах, а также отслеживать поставки медикаментов на склад и их хранение.

Регистрация пациента осуществляется заявкой онлайн или в самой клинике, администратор вручную вносит пациента в базу, автоматически создаётся и

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Инв. № подл.
Инв. № подл.	Подп. и дата

Лит.	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Лист

5

привязывается к пациенту, амбулаторная карта, регистрирует пациента на приём к нужному ему врачу в определённое согласованное с расписанием время.

Врач видит всех назначенных пациентов к нему на приём и принимает их порядке очереди. После проведения осмотра и оказания лечения, врач выставляет диагноз и направляет на курс лечения и вносит данные о приёме в амбулаторную карту. Амбулаторная карта является документом клиники и не выдаётся пациенту, но пациент получает выписку на основе данных из медицинской карты. После приёма медицинская карта сохраняется в базу и сразу же распечатывается, вкладывается в папку пациента и далее отправляется на склад для хранения.

Медицинская карта заполняется на основании формы N 025/у врачом после окончания приёма и заносится в базу данных. Далее она отправляется на печать, первый экземпляр идёт на руки пациенту, второй отправляется на склад для дальнейшего хранения.

При принятии на работу нового врача в случае наличия всех нужных документов и должного опыта работы, администратор регистрирует его в базу как пользователя с уникальным логином и паролем, описывает его специальность, закрепляет за ним используемые расходуемые медицинские материалы и препараты, составляет его расписание.

В клинике ведётся учёт поставок и хранение медицинских инструментов, расходуемых материалов и препаратов на складе. Ответственный за склад наблюдает и вносит изменения в базу данных о хранящихся на складе медицинских материалах. В связи с не хваткой материалов он делает согласованный заказ нужных клинике материалов. Так же ведёт список поставщиков с возможностью их редактирования.

## 1.2 Требования к программному продукту

Учитывая, что программа была разработана в 2011 году и устарела в плане дизайна и функционала. В связи с этим будет разработан ряд подсистем для более удобного и функционального использования системы.

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Инв. № подл.
Инв. № подл.	Подп. и дата

Лит	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Лист  
6

Подсистемы:

- подсистема для управления складом;
- подсистема для учёта и управления персоналом;
- подсистема для записи клиентов на приём.

Вход в программу должен представлять из себя страницу с 2 полями для ввода пароля и логина и выпадающий список с выбором от лица кого мы входим в программу (врач или администратор). У администратора должен быть фиксированный известный только ему и начальству пароль, имеющий возможность изменения. Логин и пароль врачам создаёт и редактирует администратор.

Все нужные кнопки или вкладки должны находиться в одном месте (в одной панели) и располагаться в порядке важности от более необходимых, например, “приёмы” и до менее необходимых, “выход”. Панель нужных элементов должна располагаться на каждой странице для более удобной навигации по приложению. Должно быть реализовано сохранения интерфейса при переходе с одной страницы на другую, если на одной странице будут заполнены поля и не нажата кнопка сохранения, при переходе на другую страницу и последующем возвращении данные должны остаться на своих местах. При, удалении и сохранение должно появляться диалоговое окно для подтверждения действия с кнопками “Да” и “Отмена”.

Должны быть реализованы функции по сортировке и поиску всех возможных данных, связанных с врачами, пациентами, приёмами, материалами и записями в медицинской карте.

Обеспечить связь пациентов с их личной медицинской картой, врачей с расписанием, связь пациентов и врачей(приём), связь врача и медицинской карты.

После приёма у врача появляется окно с потраченными им материалами на конкретного человека за весь приём, они должны списаться со склада.

Подсистема для управления складом обязана представлять из себя отдельную страницу, содержащую, список материалов и имеющая возможность их добавления, редактирования и удаления. Так же страница должна содержать

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата						Лист 7
Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ					

список поставщиков, состоящий из названия компании, телефонного номера, электронной почты, типа доставляемого товара, в котором также имеется возможность добавления, удаления и редактирования.

Для этой подсистемы должны быть созданы следующие поля для дальнейшего заполнения данными:

- наименование материала – строковый тип;
- тип материала – строковый тип;
- единица измерения – строковый тип;
- количество – целочисленный тип;
- срок годности – формат даты;
- цена – Денежный формат.

Подсистема для учёта и управления персоналом должна распространяться на две уникальные страницы. Первая страница должна содержать всех врачей содержащихся в базе данных для это в ней нужно создать следующие поля для дальнейшего заполнения данными:

- фιο – строковый тип;
- номер телефона – строковый тип;
- электронная почта – строковый тип;
- специальность – строковый тип;
- Login – строковый тип;
- Password – строковый тип;
- таблица с расписанием – ссылочный тип.

Вторая страница должна быть уникальной для каждого врача, содержать список приёмов, назначенных к конкретному врачу в конкретное время, который заполняет администратор. На этой странице у врач должны быть реализованы следующие функции возможность выбрать человека, назначенного к нему, сделать запись в медицинскую карту сохранить её и распечатать. После окончания одного приёма, когда врач выбирает следующего человека на приём, прошлая запись должна удалиться. Данные медицинской карте должны храниться в базе данных

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата	Инв. № инв.	Подп. и дата	Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ	Лист
													8



для этого в ней нужно создать следующие поля для дальнейшего заполнения данными:

- дата заполнения медицинской карты – дата и время;
- жалобы – строковый тип;
- диагноз – строковый тип;
- курс лечения – строковый тип;
- комментарий врача – строковый тип;
- лечащий врач – ссылочный тип;
- лечение – строковый тип.

Подсистема для записи клиентов на приём должна содержаться на отдельной странице, которую видит и использует только администратор. На странице должен отображаться список пациентов и кнопка с возможностью создания записи на приём, после нажатия на которую должно открыться окно, на котором можно выбрать врача пациента и время начала приёма. При вводе даты приёма должно отобразиться сообщение, показывающее занято ли на данное время. Данные о приёме должны временно храниться в базе данных для этого потребуются следующие поля:

- фιο пациента – ссылочный тип;
- фιο врача – ссылочный тип;
- дата начала приёма – дата и время.

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата						
Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ					Лист
										9

## 2 ПРОЕКТИРОВАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

Исходной точкой построения логической схемы данных является представление предметной области в виде одного или нескольких отношений, и на каждом шаге проектирования производится некоторый набор схем отношений (Рисунок 1).

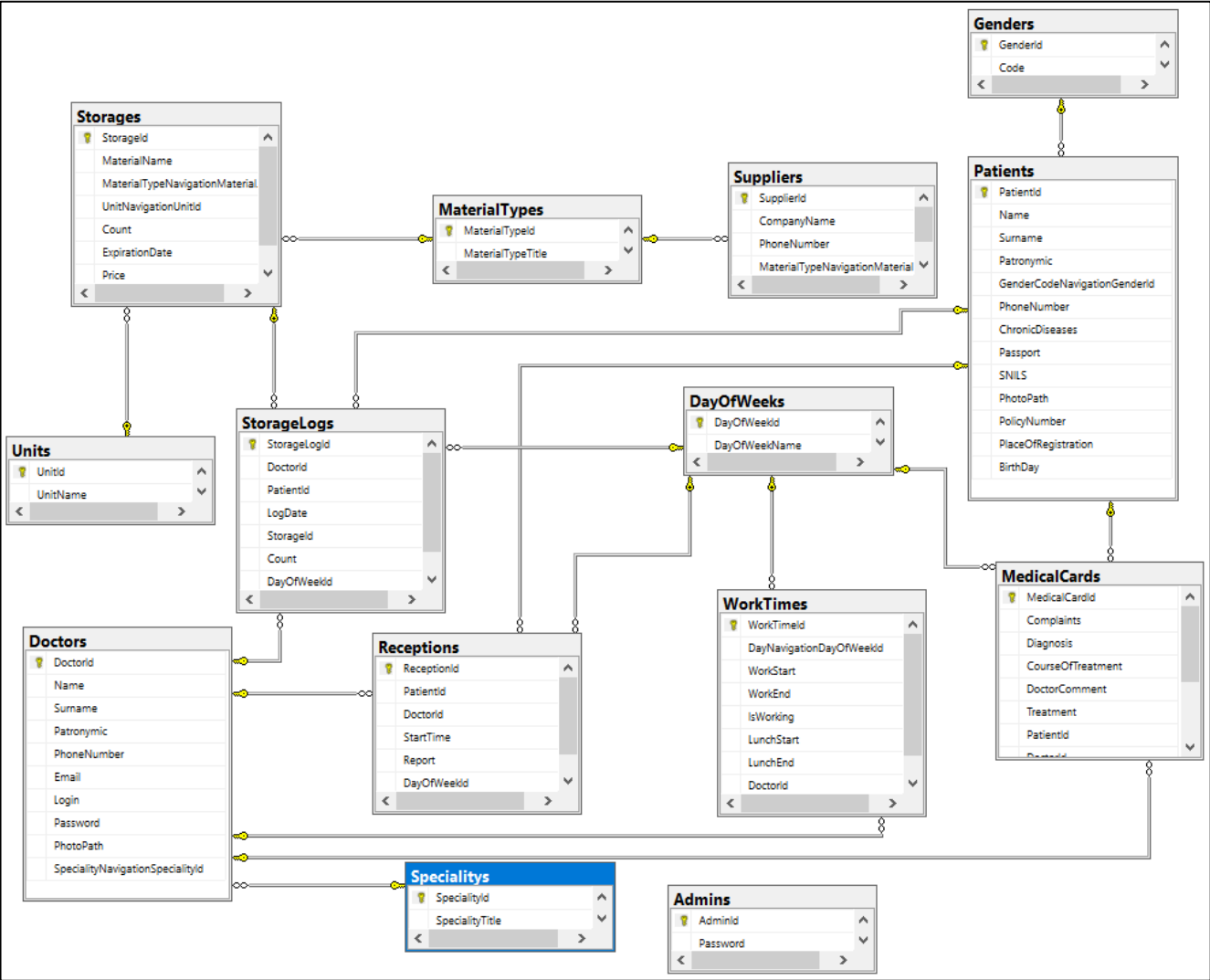


Рисунок 1 - Структура базы данных

Сущности и их связи для функционирования подсистемы управления складом и хранения в них данных.

Главная таблица “Storages” (главная в данной подсистеме сущность “Склад”) связана с таблицей “MaterialTypes” (связанная с главной вспомогательная

Инв. № подл.	Подп. и дата	Взам. инв. №	Подп. и дата	Инв. № дубл.

Лит.	Изм.	№ докум.	Подп.	Дата

сущность) связью один-ко-многим, к каждой записи в “Storages” идёт единственный “MaterialTypeId” (уникальный идентификатор) – первичный ключ (id) типа материала из таблицы “MaterialTypes”, к каждому id привязано название типа материала. Так же таблица “Units” (Связанная с главной вспомогательная сущность “Единицы измерения”) связана с таблицей “Storages” связью один-ко-многим с помощью id к каждому из которых привязано наименование единицы измерения. В таблице “Suppliers” (сущность “Поставщики”) по такому же принципу осуществлена связь с таблицей “MaterialTypes”, таблица также имеет связь один-ко-многим. Таким образом подсистема для управления складом представляет из себя главную сущность “Storages” связанную двумя вспомогательными таблицами “Units” и “MaterialTypes” связями один-ко-многим и сущность “Suppliers” связанную с одной вспомогательной таблицей “Suppliers” связью один-ко-многим. Поля StorageId, MaterialTypeId, SupplierId, UnitId – первичные ключи из каждой сущности. Поля MaterialTypeNavigationMaterialTypeId, UnitNavigationUnitId – составные ключи, исходящие из вспомогательных таблиц в главные сущности.

Сущность “MaterialTypes” содержит в себе конкретный тип материала со своим уникальным индексом, который в этой сущности не повторяется, так же содержит следующие, условленные с заказчиком заполненные заранее разработчиком поля (название типа материала).

Поля:

- MaterialTypeTitle – NVARCHAR.

Сущность “Units” содержит в себе конкретную единицу измерения со своим уникальным индексом, который в этой сущности не повторяется, так же содержит следующие заполненные разработчиком поля (наименование единицы измерения).

- UnitName – NVARCHAR.

Сущность “Storages” содержит в себе конкретный материал со своим уникальным индексом, который в этой сущности не повторяется, так же содержит поля для заполнения (название, тип материала, единица измерения, количество, срок годности, цена).

Поля:

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата						Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ	Лист	11

- MaterialName – NVARCHAR;
- MaterialTypeNavigationMaterialTypeId - составной ключ;
- UnitNavigationUnitId – составной ключ;
- Count – INT;
- ExpirationDate – DateTime;
- Price – Decimal (Money).

Сущность “Suppliers” содержит в себе конкретного поставщика со своим уникальным индексом, который в этой сущности не повторяется, так же содержит поля для заполнения (название компании, номер телефона, тип материала).

Поля:

- CompanyName – NVARCHAR;
- PhoneNumber – NVARCHAR;
- MaterialTypeNavigationMaterialTypeId – составной ключ.

Сущности и их связи для функционирования подсистемы для записи клиента на приём и временного хранения в них данных.

Главная таблица “Receptions” (главная в данной подсистеме сущность “Приёмы”) связана с таблицей Patients (сущность “Пациенты”) связью один-ко-многим, на каждую запись идёт один уникальный пациент из множества других. Так же таблица “Receptions” связана с таблицей Doctors (сущность “Врачи”) связью один-ко-многим, то есть это значит что на каждую запись идёт один уникальный врач из множества других.

Поле ReceptionId – первичный ключ, Поля PatientId и DoctorId – составные ключи.

Таблица “Receptions” содержит в себе конкретную запись на приём со своим уникальным индексом, который в этой сущности не повторяется, так же содержит поля для заполнения данными (пациент, доктор, время начала приёма).

Поля:

- PatientId – составной ключ;
- DoctorId – составной ключ;

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ	Лист
											12

- StartTime – DateTime.

Сущности и их связи для функционирования подсистемы учёта и управления персоналом и хранения в них данных.

Главная таблица “Doctors” (главная в данной подсистеме сущность “Врачи”) связывается с таблицей “Specialitys” (связанная с главной вспомогательная сущность) связью один-ко-многим к каждой записи (врачу) идёт единственный “SpecialityId” (Уникальный ключ сущности “специальность”) – первичный ключ (id) специальности из таблицы “Specialitys”, к каждому id привязано название специальности. Таблица “Receptions” (сущность “Приём”) связана с таблицей той же связью один-ко-многим, то есть на каждый приём назначен как конкретный пациент с уникальным индексом, так и врач со своим уникальным индексом. Так же сущность “Doctors” имеет связь один-ко-многим с сущностью “MedicalCards” (сущность “медицинская карта”), уникальный индекс конкретного доктора (принимающего пациента) добавляется к каждой записи в “MedicalCards” индекс доктора в медицинской карте может повторяться множество раз. В итоге мы имеем подсистему, состоящую из главной таблицы “Doctors” связанной с вспомогательной таблицей “Specialitys” связью один-ко-многим, точно такие же связи она имеет и с таблицами “MedicalCards” и “Receptions”.

Сущность “Doctors” содержит в себе конкретного врача со своим уникальным индексом, который в этой сущности не повторяется, так же содержит поля для заполнения (имя, фамилию, отчество, номер телефона, электронная почта, логин, пароль, путь до фотографии, специальность).

Поля:

- Name – NVARCHAR;
- Surname – NVARCHAR;
- Patronymic – NVARCHAR;
- PhoneNumber – NVARCHAR;
- Email – NVARCHAR;
- Login – NVARCHAR;

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.						Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ	Лист 13

- Password – NVARCHAR;
- PhotoPath – NVARCHAR;
- SpecialityNavigationSpecilityId – составной ключ.

Сущность “Specialitys” содержит в себе конкретную специальность со своим уникальным индексом, который в этой сущности не повторяется, так же содержит поля для заполнения (наименование специальности).

Поля:

- SpecialityTitle – NVARCHAR.

Сущность “MedicalCrads” содержит в себе конкретную запись в медицинской карте со своим уникальным индексом, который в этой сущности не повторяется, так же содержит поля для заполнения (дата заполнения, жалобы, диагноз, курс лечения, комментарий врача, лечащий врач, лечение).

Поля:

- DateOfCompletion – DateTime;
- Complaints – NVARCHAR;
- Diagnosis – NVARCHAR;
- CourseOfTreatment – NVARCHAR;
- DoctorComment – NVARCHAR;
- Treatment – NVARCHAR;
- TimeOfStart – NVARCHAR;
- TimeOfCompletion – NVARCHAR;
- DayOfWeekId – составной ключ;
- PatientId – составной ключ;
- DoctorId – составной ключ.

Диаграммы прецедентов. Основное назначение диаграммы — описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему.

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.

Лит	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Лист

14

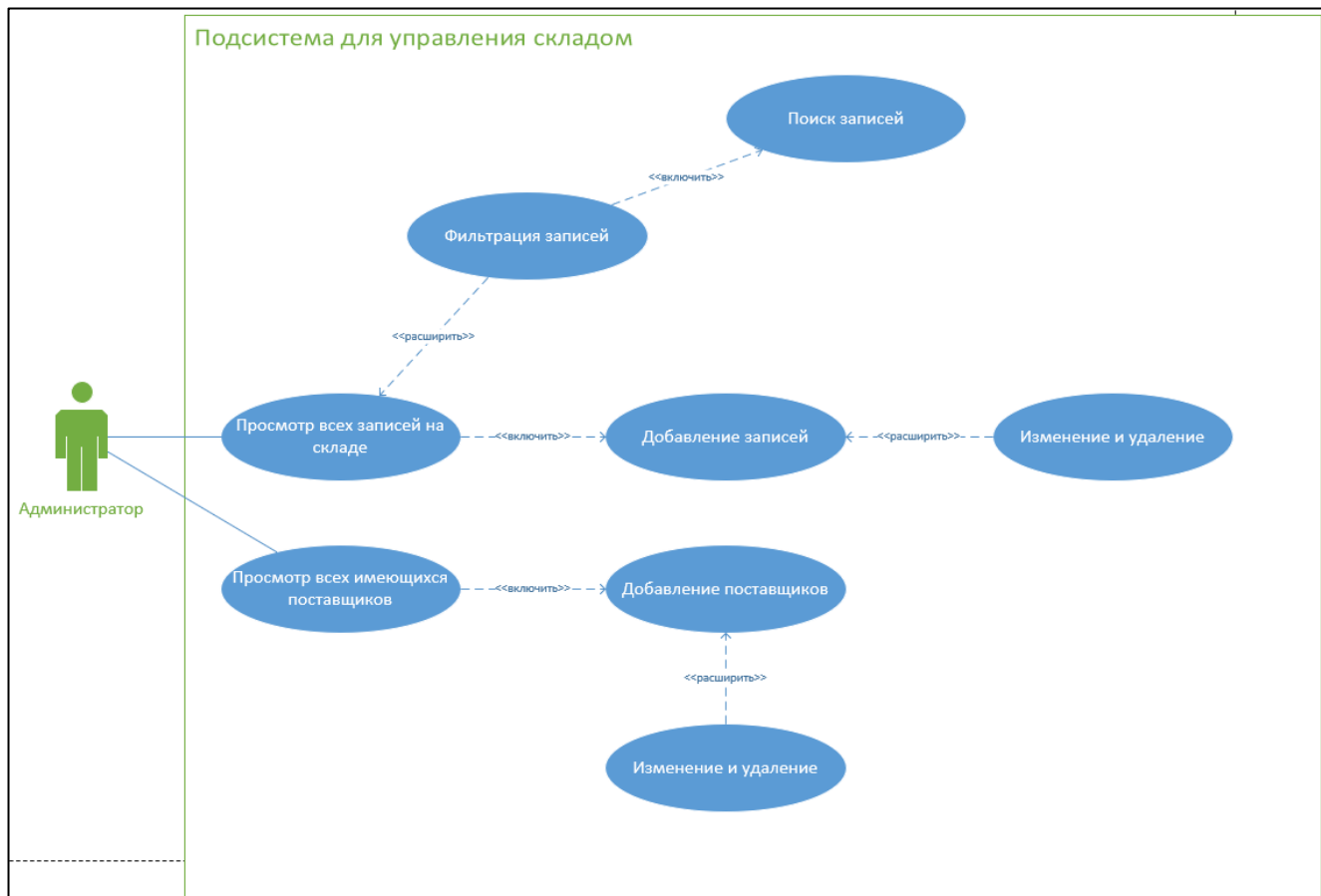


Рисунок 2 - Диаграмма прецедентов "Управление складом"

Диаграмма прецедентов для подсистемы управления складом (Рисунок 2) представляет из себя субъект (администратор) и все его возможности в данной подсистеме.

Администратор – субъект, осуществляющий управление складом.

Администратор имеет возможность просматривать все складские записи расширяем просмотр всех записей на складе возможностью фильтрации записей и их добавления. В фильтрацию включаем возможность поиска записей по параметрам. Добавление расширяем возможностью изменения и удаления записей.

Администратор может просматривать всех поставщиков. В возможность просмотра поставщиков включаем добавление поставщиков её расширяем изменением и удалением.

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

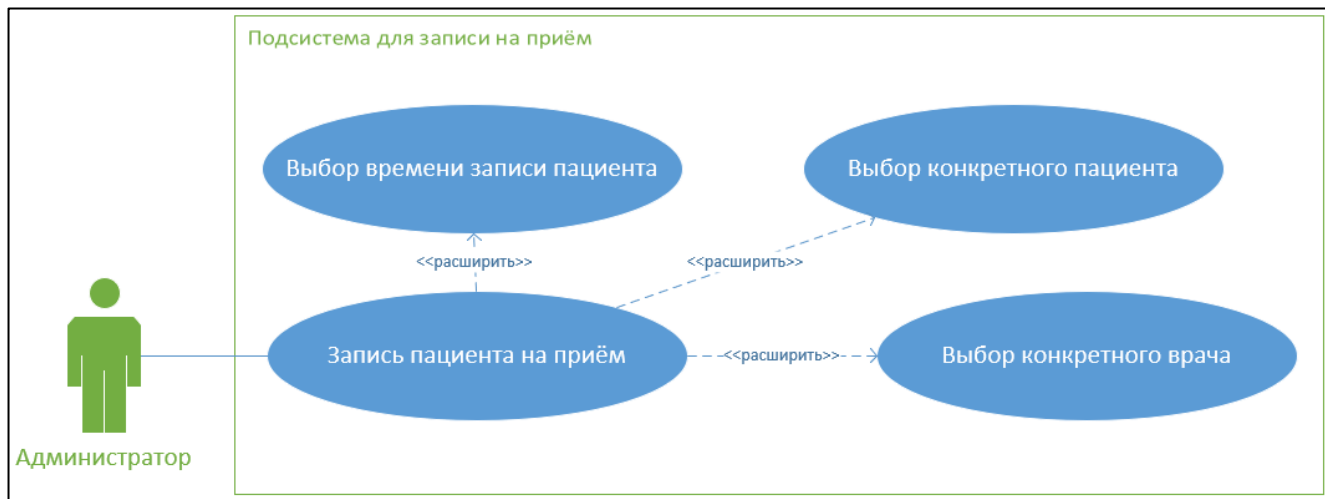


Рисунок 3 - Диаграмма прецедентов "Запись на приём"

Диаграмма прецедентов для подсистемы “запись на приём” (Рисунок 3) представляет из себя субъект (администратор) и все его возможности в данной подсистеме.

Администратор – субъект, осуществляющий запись на приём.

В данной подсистеме субъект обладает возможностью записи пациента на приём расширится эта возможность выбором времени записи, конкретного пациента и конкретного врача.



Рисунок 4 - Диаграмма прецедентов "Управление персоналом"

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Лит.	Изм.	№ докум.	Подп.	Дата



Диаграмма прецедентов для подсистемы “учёта и управления персоналом” (Рисунок 4) представляет из себя 2 субъекта (врач и администратор) и все их возможности в этой подсистеме.

Врач – субъект, осуществляющий принятие пациента на приём и заполняющий медицинскую карту.

Администратор – субъект, осуществляющий добавление врачей и пациентов.

Оба субъекта могут печатать медицинскую карту.

В данной подсистеме у субъекта “Врач” имеются возможности просмотра назначенных приёмов, просмотра списка пациентов, печать медицинских карт. Просмотр списка пациентов расширяется изменением данных пациентов. В просмотр назначенных приёмов включается принятие приёма и запись данных в медицинскую карту.

У Субъекта “Администратор” имеются возможности просмотра списка пациентов, просмотра списка врачей, печати медицинских карт. Просмотр списка клиентов включает в себя добавление новых пациентов, которое расширяется возможностью их изменения. Просмотр списка врачей включает в себя возможность добавления новых врачей, которая расширяется возможностью их изменения и их удаления.

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата	Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ	Лист
											17

## 3 ТЕХНИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

### 3.1 Выбор состава технических программных средств

На сегодняшний момент язык программирования C# один из самых мощных, быстро развивающихся и востребованных языков в IT-отрасли. В настоящий момент на нем пишутся самые различные приложения: от небольших десктопных программ до крупных веб-порталов и веб-сервисов, обслуживающих ежедневно миллионы пользователей.

C# уже не молодой язык, как и вся платформа .NET уже прошел большой путь. Первая версия языка вышла вместе с релизом Microsoft Visual Studio .NET в феврале 2002 года. Текущей версией языка является версия C# 11, которая вышла 8 ноября 2022 года вместе с релизом .NET 7.

C# является языком с Си-подобным синтаксисом и близок в этом отношении к C++ и Java.

C# является объектно-ориентированным и в этом плане много перенял у Java и C++. Так же, как и эти языки, C# поддерживает полиморфизм, наследование, перегрузку операторов, статическую типизацию. Объектно-ориентированный подход позволяет решить задачи по построению крупных, но в тоже время гибких, масштабируемых и расширяемых приложений. И C# продолжает активно развиваться, и с каждой новой версией появляется все больше интересного функционала. Поэтому для этого проекта и был выбран данный язык.

Для построения клиентского приложения был выбран WPF. Windows Presentation Foundation — аналог WinForms, система для построения клиентских приложений Windows с визуально привлекательными возможностями взаимодействия с пользователем.

В основе WPF лежит векторная система визуализации, не зависящая от разрешения устройства вывода и созданная с учётом возможностей современного графического оборудования. WPF предоставляет из себя средства для создания

Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.	КП-ПР-32-09-2023-ПЗ					Лист
										18
Лит	Изм.	№ докум.	Подп.	Дата						

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

Лит	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Лист  
19

КП-ПР-32-09-2023-ПЗ

Лист

19

Для простого и более удобного построения пользовательского интерфейса



AddSupplierWindow содержит все поля для заполнения данных о поставщике и кнопку его добавления в базу данных, так же это окно используется для изменения поставщика.

AdminWindow содержит все кнопки переходов на страницы админа и фрейм в который эти страницы выводятся и кнопку выхода.

DoctorAddWindow содержит все поля для заполнения данных о враче и кнопку его добавления в базу данных, так же это окно используется для изменения врача.

DoctorAddToMedicalCardWindow содержит все поля пациента без возможности их изменения и поля для заполнения медицинской карты кнопку завершения приёма и кнопку списания со склада.

DoctorWindow содержит информацию о враче, а так же все кнопки переходов на страницы этого врача и фрейм в который эти страницы выводятся и кнопку выхода.

ReceptionWindow содержит информацию о пациенте, все поля для заполнения данных о приёме и кнопкой записи на приём.

StorageAddWindow содержит все поля для заполнения данными о записи в складе и кнопку добавления в базу, так же это окно используется для изменения записи в складе.

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

Лит	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

## 4 ТЕСТИРОВАНИЕ

Тестирование в белом ящике.

Тестирование в белом ящике (White Box Testing) — это метод тестирования программного обеспечения, при котором тестировщик имеет доступ к исходному коду программы и может проверять его работу на уровне отдельных функций и процедур.

Основным преимуществом тестирования в белом ящике является возможность более глубокого и точного тестирования программы. Тестировщик может проверить каждую строку кода на наличие ошибок и недочетов, а также оценить покрытие кода тестами. Это позволяет выявлять ошибки на более ранних этапах разработки, когда их исправление проще и дешевле.

Также тестирование в белом ящике позволяет оптимизировать код программы, улучшить его производительность и устранить узкие места, что в свою очередь повышает качество и надежность программы.

Однако, следует учитывать, что тестирование в белом ящике требует от тестировщика хорошего знания языка программирования, на котором написана программа, а также опыта работы с исходным кодом. Кроме того, этот метод тестирования не может полностью заменить тестирование в черном ящике (Black Box Testing), которое позволяет проверить работу программы на уровне ее функциональности и соответствия требованиям заказчика.

Все тесты полей и функций расположены в приложении (Приложение — сценарий тестирования).

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата						
Лит.	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ					Лист
										22

## ЗАКЛЮЧЕНИЕ

В заключении курсовой работы по разработке CRM системы для медицинского центра можно отметить, что данная система является необходимым инструментом для улучшения качества обслуживания пациентов и оптимизации работы медицинского персонала. В процессе разработки были рассмотрены основные требования и функциональные возможности CRM системы, а также проанализированы существующие аналоги.

В результате была разработана CRM система, которая позволяет автоматизировать процессы записи на прием, ведения медицинских карт, складского учёта. Она обладает удобным интерфейсом и интуитивно понятными функциями, что позволяет медицинскому персоналу более качественно выполнять свою работу.

В дальнейшем, для улучшения функциональности системы, можно рассмотреть возможность добавления модулей для анализа данных и статистики, а также интеграцию с другими медицинскими системами. В целом, разработанная CRM система является эффективным инструментом для улучшения работы медицинского центра и повышения уровня обслуживания пациентов.

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата	КП-ПР-32-09-2023-ПЗ					Лист
										23
Лит	Изм.	№ докум.	Подп.	Дата						

## СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Сайт о программировании, Metanit [Электронный ресурс], - Режим доступа: <https://metanit.com/sharp/efcore/> (Дата обращения 07.03.2023).
2. Сайт о программировании, Metanit [Электронный ресурс], - Режим доступа: <https://metanit.com/sharp/wpf/5.14.php> (Дата обращения 07.03.2023).
3. Документация Microsoft по .NET [Электронный ресурс], - Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/> (Дата обращения 18.03.2023).
4. Форум о программировании Stackoverflow [Электронный ресурс], - Режим доступа: <https://stackoverflow.com/questions/tagged/c%23> (Дата обращения 01.03.2023).
5. Форум программистов и сисадминов Киберфорум [Электронный ресурс], - Режим доступа: <https://www.cyberforum.ru/wpf-silverlight/> (Дата обращения 27.03.23).
6. Медицинская клиника «Ваш доктор» [Электронный ресурс], - Режим доступа: <https://ksmvd.ru> (Дата обращения 26.02.2023).

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата		Лист
						24
Лит	Изм.	№ докум.	Подп.	Дата	КП-ПР-32-09-2023-ПЗ	



## ПРИЛОЖЕНИЕ – СЦЕНАРИЙ ТЕСТИРОВАНИЯ

Поле	Описание
Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddDoc-UI-1
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Проверка ввода всех полей в окне добавления или изменения врачей
Краткое изложение теста	Если все данные заполнены верно и под полями нет ошибок, то при нажатии на кнопку добавление или изменения врач должен успешно добавиться в базу данных
Этапы теста	<ol style="list-style-type: none"> <li>1. Войти в программу через админа;</li> <li>2. Нажать на кнопку “Врачи”;</li> <li>3. Нажать на кнопку добавить;</li> <li>4. Заполнить все поля корректными данными;</li> <li>5. Нажать на кнопку добавить или изменить.</li> </ol>

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	

Лит	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Лист

25

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Тестовые данные	Все поля заполнены корректными данными
Ожидаемый результат	Врач добавился в базу данных или изменился
Фактический результат	Врач добавился в базу данных или изменился
Предварительное условие	
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddDoc-UI-2
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

КП-ПР-32-09-2023-ПЗ

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Заголовок/название теста	Проверка ввода всех полей в окне добавления или изменения врачей
Краткое изложение теста	Если все данные заполнены не верно и под полями есть ошибки, то при нажатии на кнопку добавление врач не должен добавиться в базу данных и выведется сообщение что поля не заполнены корректно.
Этапы теста	<ol style="list-style-type: none"> <li>1. Войти в программу через админа;</li> <li>2. Нажать на кнопку “Врачи”;</li> <li>3. Нажать на кнопку добавить;</li> <li>4. Заполнить все поля корректными данными;</li> <li>5. Нажать на кнопку добавить или изменить;</li> </ol>
Тестовые данные	Не все поля заполнены корректными данными
Ожидаемый результат	Врач не добавился в базу данных или не изменился и вывелось сообщение о том, что не все поля заполнено
Фактический результат	Врач не добавился в базу данных или не изменился и вывелось сообщение о том, что не все поля заполнено
Предварительное условие	
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
------	----------

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	Подп. и дата

Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddSupplier-UI-1
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Проверка ввода всех полей в окне добавления или изменения поставщиков
Краткое изложение теста	Если все данные заполнены верно и под полями нет ошибок, то при нажатии на кнопку добавление или изменения поставщик должен успешно добавиться в базу данных
Этапы теста	<ol style="list-style-type: none"> <li>1. Войти в программу через админа;</li> <li>2. Нажать на кнопку “Поставщики”;</li> <li>3. Нажать на кнопку добавить или изменить;</li> <li>4. Заполнить все поля корректными данными;</li> <li>5. Нажать на кнопку добавить или изменить.</li> </ol>
Тестовые данные	Все поля заполнены корректными данными
Ожидаемый результат	Поставщик добавился в базу данных или изменился

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

КП-ПР-32-09-2023-ПЗ

Инв. № подл.	Подп. и дата
	Взам. инв. №
	Инв. № дубл.
	Подп. и дата
	Инв. № подл.

Фактический результат	Поставщик добавился в базу данных или изменился
Предварительное условие	
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddSupplier-UI-2
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Проверка ввода всех полей в окне добавления или изменения поставщиков

Лит	Изм.	№ докум.	Подп.	Дата

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	Взам. инв. №

Краткое изложение теста	Если все данные заполнены не верно и под полями есть ошибки, то при нажатии на кнопку добавление поставщик не должен добавиться в базу данных и выведется сообщение что поля не заполнены корректно.
Этапы теста	<ol style="list-style-type: none"> <li>1. Войти в программу через админа;</li> <li>2. Нажать на кнопку “Склад”;</li> <li>3. Нажать на кнопку добавить или изменить;</li> <li>4. Заполнить все поля корректными данными;</li> <li>5. Нажать на кнопку добавить или изменить;</li> </ol>
Тестовые данные	Не все поля заполнены корректными данными
Ожидаемый результат	Поставщик не добавился в базу данных или не изменился и вывелось сообщение о том, что не все поля заполнены корректно
Фактический результат	Поставщик не добавился в базу данных или не изменился и вывелось сообщение о том, что не все поля заполнены корректно
Предварительное условие	
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
------	----------

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	

Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddStorage-UI-1
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Проверка ввода всех полей в окне добавления или изменения записи на складе
Краткое изложение теста	Если все данные заполнены верно и под полями нет ошибок, то при нажатии на кнопку добавление или изменения запись склада должен успешно добавиться в базу данных
Этапы теста	<ol style="list-style-type: none"> <li>1. Войти в программу через админа;</li> <li>2. Нажать на кнопку “Склад”;</li> <li>3. Нажать на кнопку добавить или изменить;</li> <li>4. Заполнить все поля корректными данными;</li> <li>5. Нажать на кнопку добавить или изменить.</li> </ol>
Тестовые данные	Все поля заполнены корректными данными

Лит	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Лист

31

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Ожидаемый результат	Запись склада добавилась в базу данных или изменилась
Фактический результат	Запись склада добавилась в базу данных или изменилась
Предварительное условие	
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddStorage-UI-2
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий

Лит	Изм.	№ докум.	Подп.	Дата



Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	Инв. № инв.

Заголовок/название теста	Проверка ввода всех полей в окне добавления или изменения записей на складе
Краткое изложение теста	Если все данные заполнены не верно и под полями есть ошибки, то при нажатии на кнопку добавление запись склада не должна добавиться в базу данных и выведется сообщение что поля не заполнены корректно.
Этапы теста	<ol style="list-style-type: none"> <li>1. Войти в программу через админа;</li> <li>2. Нажать на кнопку “Склад”;</li> <li>3. Нажать на кнопку добавить или изменить;</li> <li>4. Заполнить все поля корректными данными;</li> <li>5. Нажать на кнопку добавить или изменить;</li> </ol>
Тестовые данные	Не все поля заполнены корректными данными
Ожидаемый результат	Запись склада не добавилась в базу данных или не изменилась и вывелось сообщение о том, что не все поля заполнены корректно
Фактический результат	Запись склада не добавилась в базу данных или не изменилась и вывелось сообщение о том, что не все поля заполнены корректно
Предварительное условие	
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
------	----------

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

КП-ПР-32-09-2023-ПЗ

Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddMedCard-UI-1
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Проверка ввода всех полей в окне добавления записи в мед. карту
Краткое изложение теста	Если все данные заполнены верно и под полями нет ошибок, то при нажатии на кнопку завершить запись мед. карты должна успешно добавиться в базу данных
Этапы теста	<ol style="list-style-type: none"> <li>1. Войти в программу через врача;</li> <li>2. Нажать на кнопку “Записи”</li> <li>3. Нажать на записи кнопку принять;</li> <li>4. Заполнить все поля корректными данными;</li> <li>5. Нажать на кнопку завершить.</li> </ol>
Тестовые данные	Все поля заполнены корректными данными
Ожидаемый результат	Запись мед. карты добавилась в базу данных

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

Фактический результат	Запись мед. карты добавилась в базу данных
Предварительное условие	В базе есть записи на приём
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	AddMedCard-UI-2
Приоритет тестирования (Низкий/Средний/Высокий)	Высокий
Заголовок/название теста	Проверка ввода всех полей в окне добавления записи в мед. карту
Краткое изложение теста	Если все данные заполнены не верно и под полями есть ошибки, то при нажатии на кнопку завершить

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата
Инв. № подл.	Подп. и дата

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

КП-ПР-32-09-2023-ПЗ

Лист

35

Инв. № подл.	Подп. и дата
	Взам. инв. №
	Инв. № дубл.
	Подп. и дата
	Инв. № подл.

	запись мед. карты не должна добавиться в базу данных и выведется сообщение что поля не заполнены корректно.
Этапы теста	1. Войти в программу через врача; 2. Нажать на кнопку “Записи” 3. Нажать на записи кнопку принять; 4. Заполнить все поля корректными данными; 5. Нажать на кнопку завершить.
Тестовые данные	Не все поля заполнены корректными данными
Ожидаемый результат	Запись мед. карты не добавилась в базу данных и вывелось сообщение о том, что не все поля заполнены корректно
Фактический результат	Запись мед. карты не добавилась в базу данных и вывелось сообщение о том, что не все поля заполнены корректно
Предварительное условие	В базе есть записи на приём
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Поле	Описание
Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0

Лит.	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Инв. № подл.	Подп. и дата
Инв. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	Search-UI-1
Приоритет тестирования (Низкий/Средний/Высокий)	Средний
Заголовок/название теста	Проверка поиска
Краткое изложение теста	Вывод необходимых элементов при вводе данных в строку поиска
Этапы теста	1. Авторизация; 2. Перейти на окно, на котором имеется поиск; 3. Заполнить строку поиска
Тестовые данные	В базе данных имеется человек с фамилией Иванов, в строку поиска вводим Ива
Ожидаемый результат	Выводится Иванов и все остальные элементы, содержащие «Ива» в фамилии
Фактический результат	Выводится Иванов и все остальные элементы, содержащие «Ива» в фамилии
Предварительное условие	В базу добавлены элементы, подходящие под строку, введенную в поле поиска
Статус (Зачет/Незачет)	Зачет

Лит	Изм.	№ докум.	Подп.	Дата
-----	------	----------	-------	------

КП-ПР-32-09-2023-ПЗ

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

Примечания/комментарии	
------------------------	--

Поле	Описание
Название проекта	Разработка информационной системы «Подсистемы для CRM системы Медицинского центра»
Рабочая версия	1.0
Имя тестирующего	Панин Андрей Александрович
Дата(ы) теста	13.04.2023
Тестовый пример#	Search-UI-1
Приоритет тестирования (Низкий/Средний/Высокий)	Средний
Заголовок/название теста	Проверка поиска
Краткое изложение теста	Проверка вывода, если в базе не имеется элементов, содержащих строку, введенную в строку поиска
Этапы теста	1. Авторизация; 2. Перейти на окно, на котором имеется поиск; 3. Заполнить строку поиска

Тестовые данные	В базе данных нет людей с фамилией, содержащей «Ива», в строку поиска вводим Ива
Ожидаемый результат	Ничего не выводится
Фактический результат	Ничего не выводится
Предварительное условие	В базе данных нет элементов, содержащих строку, введенную в поле поиска, в фамилии
Статус (Зачет/Незачет)	Зачет
Примечания/комментарии	

Инв. № подл.	Подп. и дата	Взам. инв. №	Подп. и дата	Инв. № дубл.

Лит	Изм.	№ докум.	Подп.	Дата

КП-ПР-32-09-2023-ПЗ

Лист

39