

## Содержание

Постановка задачи .....	2
Описание классов.....	2
Взаимосвязь классов.....	31
Описание разработанного приложения (руководство пользователя).....	37
Тестирование.....	46
Заключение.....	49

## Постановка задачи

Разрабатывается программа, которая автоматизирует рабочее место регистратора поликлиники. Задача этой программы облегчить рутинные задачи и увеличить эффективность работы медрегистратора.

Необходимо реализовать: создание, редактирование, поиск и хранение медицинской карты, возможность записи на прием к специалисту, просмотр расписания специалистов.

При создании медицинская карта записывается в создаваемый в ходе запуска программы файл excel, при записи пациента на прием учитывается время работы, а также длительность приема специалиста, при просмотре расписания осуществляется импорт данных о специалистах из файла excel.

При разработке необходимо использовать ООП, инкапсуляцию, наследование, полиморфизм. Язык программирования – C#, технология графического интерфейса – Windows Forms.

## Описание классов

**Class Program** – основной класс, содержащий метод Main, который запускает форму.

(таблица 1)

Таблица 1 – Данные-элементы класса Program

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Методы</b>			
Main	private	void	Запуск формы

**Class Info** - Родительский класс информации о человеке. Содержит свойства и метод для работы с личной информацией человека. (таблица 2)

Таблица 2 - Данные-элементы класса Info

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Свойства</b>			
FirstName	public	string	Имя
SecondName	public	string	Фамилия
MiddleName	public	string	Отчество
DateofBirth	public	DateOnly	Дата рождения
Street	public	string	Улица проживания
HouseNumber	public	string	Номер дома

Apartment	public	int	Номер квартиры
PhoneNumber	public	string	Номер телефона
Post	public	string	Должность
<b>Методы</b>			
Serialisation	public	void	Виртуальная функция для записи в файл дочерних классов.

**Class Patient** - Дочерний класс класса Info. Содержит свойства необходимые для записи информации о пациенте. (таблица 3)

Таблица 3 - Данные-элементы класса Patient

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Свойства</b>			
AnotherAd	public	bool	Индикатор совпадения адреса проживания с пропиской(true – совпадает, false – другой адрес)
AnStreet	public	string	Улица проживания пациента(адрес фактического проживания)
AnHouseNumber	public	string	Номер дома(адрес фактического проживания)
AnApartment	public	int	Квартира(адрес фактического проживания)
InsuranceCompany	public	string	Страховая компания
InsuranceNumber	public	int	Номер медицинского страхового полиса
WorkPlace	public	string	Место работы

*Class Specialist* - Дочерний класс класса Info. Содержит свойства и методы для работы с информацией о специалисте. (таблица 4)

Таблица 4 - Данные-элементы класса Specialist

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Свойства</b>			
Day	public	int	Номер дня недели для просмотра расписания
SpecialistTime	public	int	Выбранный индекс времени во время записи пациентов
SpecialistName	public	int	Выбранный индекс специалиста во время записи пациентов
SpecialisationRecording	public	int	Выбранный индекс специальности во время записи пациентов
Specialisation	public	int	Выбранный индекс специальности во время просмотра расписания
DayofWeek	public	string	Текущий день недели
<b>Методы</b>			
Serialisation()	public	void	Запись класса в файл
SpecialistDeseriialisation(Specialist specialist)	public	Specialist	Чтение класса из файла

**Class MedicalCard** - Дочерний класс класса Patient. Содержит в себе свойства и методы для работы с медицинскими картами пациентов. (таблица 5)

Таблица 5 - Данные-элементы класса MedicalCard

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
createfile	private	static bool	индикатор создания файлов
countcard	private	int	Счетчик медицинских карт
<b>Свойства</b>			
CreateFile	public	bool	Получение поля createfile и запись в него
CountCard	public	int	Получение поля countcard запись в него
<b>Методы</b>			
EddingCard	public	void	Редактирование медицинской карты в файле excel
AddCard(int countPatient, Excel excel)	public	void	Добавление мед карты пациента в excel файл
Editcard(int count, Excel excel)	public	string[,]	Редактирование медицинской карты.
Serialisation()	public	void	Запись класса в файл
CardDeserialisation(MedicalCard card)	public	MedicalCard	Чтение класса из файла

***Class Excel*** - Класс содержащий в себе поля и методы для работы с excel файлами  
(таблица 6)

Таблица 6 - Данные-элементы класса Excel

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
excelapp	private	_Application	Работа с приложением excel
wb	public	Workbook	Работа с книгами excel
ws	public	Worksheet	Работа со страницами excel
<b>Методы</b>			
RedactCells(string[] medcard, int i, int j, int sort)	public	void	Редактирование строки в файле excel
WriteCells(string[] medcard, int i, int j)	public	void	Запись строки в файл
WriteCells(string[] medcard, int i, int j, int sort)	public	void	Запись строки в файл с последующей сортировкой
CreateDefaultFile(string path)	public	void	Создание файла
CreateFile(string path)	public	void	Создание файла с таблицей
ReadCells(int i)	public	String[,]	Получение всей строки из файла
ReadCells(int i, int j)	public	String[,]	Получение необходимого количества клеток в строке
ReadColumns(int rows, int columns, string profession)	public	string	Чтение клетки из файла для определенного специалиста

ReadColumns(int rows, int columns)	public	string	Чтение из клетки
SortColumns(int row)	public	void	Сортировка строки в файле
SortRows(int rows, int columns,int sort)	public	void	Сортировка всех строк в файле по определенному столбцу
SearchTitle(string path)	public	int	Получения номера столбца с определенным заголовком
RowCount()	public	int	Получение количества заполненных строк в файле
ColumnsCount(int row)	public	int	Получение количества заполненных столбцов в файле
Close()	public	void	Закрытие файла

***Class Menu*** - Класс формы главного окна. В нем описываются основные методы и поля, а также взаимодействие с другими окнами. (таблица 7)

Таблица 7 - Данные-элементы класса Menu

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
Medcard	private	MedicalCard	Переменная для работы с медицинской картой
Specialist	private	Specialist	Переменная для работы с

			информацией о специалисте
scheduleStorage	private	ScheduleStorage	Переменная для работы с расписанием специалистов
Recording	private	Recording	Переменная для работы с записью пациентов
Createfile	private	bool	индикатор для создания файлов
<b>Методы</b>			
MedCard_Click(object sender, EventArgs e)	private	void	переход в окно для работы с медицинской картой
CloseMenu_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
CloseMenu_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
CloseMenu_Click(object sender, EventArgs e)	private	void	функция для закрытия программы, с сохранением



			текущего состояния программы
PatientRec_Click(object sender, EventArgs e)	private	void	открытие окна для записи пациентов на прием
Schedule_Click(object sender, EventArgs e)	private	void	открытие окна для просмотра расписания врачей
hideForm_Click(object sender, EventArgs e)	private	void	свернуть текущее окно
hideForm_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
hideForm_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
Menu_MouseMove(object sender, MouseEventArgs e)	private	void	функция для перемещения окна с помощью мыши
Menu_MouseDown(object sender, MouseEventArgs e)	private	void	функция для перемещения окна с помощью мыши

***Class ScheduleStorage*** - Класс формы просмотра расписания специалистов.

(таблица 8)

Таблица 8 - Данные-элементы класса ScheduleStorage

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
Excel	private	Excel	Переменная для работы с файлом excel
menu	private	Menu	Переменная для работы с окном главного меню
countSpecialists	private	int	Переменная для хранения количества специалистов в файле excel
countColumnSpecialisation	private	int	Номер столбца специальностей в excel файле
countColumnSpecialist	private	int	Номер столбца с текущим днем недели в excel файле
specialisationindex	private	int	Хранение номера выбранной специальности в списке
dayindex	private	int	Хранение номера выбранного дня недели в списке
<b>Свойства</b>			
Specialisationindex	public	int	Запись значения в specialisationindex и получение его

Dayindex	public	int	Запись значения в dayindex и получение его
<b>Методы</b>			
BacktoMenu_Click(object sender, EventArgs e)	private	void	вернуться в меню с сохранением текущего состояния
hideForm_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
hideForm_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
hideForm_Click(object sender, EventArgs e)	private	void	Свернуть окно
BacktoMenu_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
upperpanel_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
upperpanel_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши

upperpanel_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
DaysOfWeek_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
DaysOfWeek_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
Specialisation_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
Specialisation_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
Specialists_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
Specialists_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
BacktoMenu_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
Specialisation_SelectedIndexChanged(object sender, EventArgs e)	private	void	описание состояния окна при выборе специализации
DaysOfWeek_SelectedIndexChanged(object sender, EventArgs e)	private	void	описания состояния окна при выборе дня недели

***Class Recording*** - Класс формы записи пациентов на прием. (таблица 9)

Таблица 9 - Данные-элементы класса Recording

Элементы класса	Спецификатор доступа	Тип данных	Назначение
Поля			
excel	private	Excel	Переменная для работы с файлом excel
recordmenu	private	Menu	Переменная для работы с окном главного меню
excel2	private	Excel	Переменная для работы с файлом excel
formtimer	private	Timer	Таймер для проверки текущего окна
specialisation	private	string	Переменная для хранения информации о специальности
starttime	private	string	Хранение времени начала рабочего дня специалиста
endtime	private	string	Хранение времени окончания рабочего дня специалиста
timerecording	private	int	Хранение номера строки занятого времени текущего специалиста
countSpecialists	private	int	Хранения количества специалистов в файле excel

countColumnSpecialist	private	int	Хранение порядкового номера столбца дня недели в excel файле
countColumnSpecialisation	private	int	Хранение порядкового номера столбца специальностей в excel файле
specialistIndex	private	int	Хранение индекса выбранной специальности в списке
specialistIndex	private	int	Хранение индекса выбранного специалиста в списке
timeIndex	private	int	Хранение индекса выбранного времени записи в списке
currentDay	private	string	Хранение текущего дня недели
recordedTime	private	List	Список для хранения записанного времени
date	private	DateTime	Хранение текущей даты
<b>Свойства</b>			
SpecialisationIndex	public	int	Запись значения в specialisationIndex и получение его

SpecialistIndex	public	int	Запись значения в specialistIndex и получение его
TimeIndex	public	int	Запись значения в timeIndex и получение его
CurrentDayofWeek	public	string	Запись значения в currentDay и получение его
<b>Методы</b>			
SpecialistCB_SelectedIndexChanged(object sender, EventArgs e)	private	void	функция описывающая состояние окна при выборе специалиста
hideForm_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
hideForm_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
hideForm_Click(object sender, EventArgs e)	private	void	Свернуть окно
BacktoMenu_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку

Recording_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
Recording_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
upperpanel_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
upperpanel_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
BacktoMenu_MouseLeave(object sender, EventArgs e)	private	void	вернуться в главное меню
BacktoMenu_Click(object sender, EventArgs e)	private	void	вернуться в меню с сохранением текущего состояния
SpecialisationCB_SelectedIndexChanged(object sender, EventArgs e)	private	void	функция описывающая состояние окна при выборе специальности
RecordingButton_Click(object sender, EventArgs e)	private	void	запись на выбранное время
FormTimer_Tick(object sender, EventArgs e)	private	void	проверка всего окна на корректность
private void Recording_Load(object sender, EventArgs e)	private	void	Описывает события при загрузке текущего окна

***Class CardMenu*** - Класс формы меню работы с медицинскими картами. Содержит поля и методы для открытия окон с поиском/ записью/ редактированием медицинской карты (таблица 10)



Таблица 10 - Данные-элементы класса CardMenu

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
medcard	private	MedicalCard	Переменная для работы с медицинской картой
menu	private	Menu	Переменная для работы с окном главного меню
<b>Методы</b>			
AddCard_Click(object sender, EventArgs e)	private	void	открытие окна с добавлением медицинской карты
EditCard_Click(object sender, EventArgs e)	private	void	открытие окна с редактированием медицинской карты
Menu_MouseMove(object sender, MouseEventArgs e)	private	void	функция для перемещения окна с помощью мыши
Menu_MouseDown(object sender, MouseEventArgs e)	private	void	функция для перемещения окна с помощью мыши
SearchMedCard_Click(object sender, EventArgs e)	private	void	открытие окна с поиском медицинской карты
hideForm_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
hideForm_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе

			определенные события при отведении мыши от кнопки
hideForm_Click(object sender, EventArgs e)	private	void	Свернуть текущее окно
BackToMenu_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
BackToMenu_MouseClick(object sender, MouseEventArgs e)	private	void	вернуться в главное меню
BackToMenu_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки

**Class PatientForm** - Класс формы создания пациенту медицинской карты (таблица 11)

Таблица 11 - Данные-элементы класса PatientForm

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
medicalCard	private	MedicalCard	Переменная для работы с медицинской картой

menu	private	CardMenu	Переменная для работы с окном медицинских карт
excel	private	Excel	Переменная для работы с файлом
formtimer	private	Timer	Таймер для проверки текущего окна
savedata	private	bool	индикатор для создания файлов
<b>Методы</b>			
AddSaveData()	public	void	Добавление сохраненных ранее данных
PatientPost_KeyPress(object sender, KeyPressEventArgs e)	private	void	переход в окно для работы с медицинской картой
PatientWork_KeyPress(object sender, KeyPressEventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
AnHouseNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
AnStreet_KeyPress(object sender, KeyPressEventArgs e)	private	void	функция для закрытия программы, с сохранением

			текущего состояния программы
Apartment_KeyPress(object sender, KeyPressEventArgs e)	private	void	открытие окна для записи пациентов на прием
HouseNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	открытие окна для просмотра расписания врачей
Street_KeyPress(object sender, KeyPressEventArgs e)	private	void	свернуть текущее окно
DoB_KeyPress(object sender, KeyPressEventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
MName_KeyPress(object sender, KeyPressEventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
FName_KeyPress(object sender, KeyPressEventArgs e)	private	void	функция для перемещения окна с помощью мыши
adressbox_CheckedChanged(object sender, EventArgs e)	private	void	функция для перемещения окна с помощью мыши
addingcard_Click(object sender, EventArgs e)	private	void	добавление медицинской карты в файл
SName_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений

AnApartment_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
InsNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
PatientPhoneNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
InsCompany_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
BacktoMenu_Click(object sender, EventArgs e)	private	void	возвращение в окно для работы с медицинскими картами
PatientForm_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
PatientForm_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
upperpanel_MouseMove(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
upperpanel_MouseDown(object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
ClearForm_Click(object sender, EventArgs e)	private	void	Очистить форму
SName_Validating(object sender, System.ComponentModel.CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной

FName_Validating(object sender, System.ComponentModel.CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
MName_Validating(object sender, System.ComponentModel.CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
Street_Validating(object sender, System.ComponentModel.CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
AnStreet_Validating(object sender, System.ComponentModel.CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
PatientWork_Validating(object sender, System.ComponentModel.CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
PatientPost_Validating(object sender, System.ComponentModel.CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
FormTimer_Tick(object sender, EventArgs e)	private	void	проверка всего окна на корректность заполнения
hideForm_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
hideForm_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе

			определенные события при отведении мыши от кнопки
hideForm_Click(object sender, EventArgs e)	private	void	Свернуть окно
BacktoMenu_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
BackToMenu_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
PatientForm_Load(object sender, EventArgs e)	private	void	Описывает события при загрузке текущего окна

***Class Editingcards*** - Класс формы редактирования медицинской карты. (таблица 12)

Таблица 12 - Данные-элементы класса Editingcards

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
medical	private	MedicalCard	Переменная для работы с медицинской картой
cardmenu	private	CardMenu	Переменная для работы с окном медицинских карт

excel	private	Excel	Переменная для работы с файлом
formtimer	private	Timer	Таймер для проверки текущего окна
cardnumber	private	int	Переменная для хранения введенного номера карты
<b>Методы</b>			
DataForEditing_Click(object sender, EventArgs e)	private	void	получение данных для редактирования по номеру карты
CardNumberforEd_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
editingcard_Click(object sender, EventArgs e)	private	void	применение изменений в медицинской карте
EditSName_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditFName_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditMName_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditDoB_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditStreet_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений



EditHouseNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditApartment_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditAnStreet_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditAnHouseNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditAnApartment_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditInsCompany_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditInsNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditPatientPhoneNumber_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditPatientWork_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
EditPatientPost_KeyPress(object sender, KeyPressEventArgs e)	private	void	ограничения для вводимых значений
Editadressbox_CheckedChanged(object sender, EventArgs e)	private	void	добавления фактического адреса проживания, если он

			соответствует адресу по прописке
Editingcards_MouseMove( object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
Editingcards_MouseDown (object sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
upperpanel_MouseMove(ob ject sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
upperpanel_MouseDown(ob ject sender, MouseEventArgs e)	private	void	перемещение окна с помощью мыши
EditLastCard_Click(object sender, EventArgs e)	private	void	получение данных последней добавленной карты
StepBackCM_Click(object sender, EventArgs e)	private	void	вернуться в окно для работы с медицинскими картами
EditSName_Validating(obje ct sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
EditFName_Validating(obje ct sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
EditMName_Validating(obj ect sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной

EditStreet_Validating(object sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
EditAnStreet_Validating(object sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
EditPatientWork_Validating(object sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
EditPatientPost_Validating(object sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
CardNumberforEd_TextChanged(object sender, EventArgs e)	private	void	проверка вводимого номера карты
FormTimer_Tick(object sender, EventArgs e)	private	void	проверка всего окна на корректное заполнение
hideForm_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
hideForm_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки

hideForm_Click(object sender, EventArgs e)	private	void	свернуть окно
StepBackCM_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
StepBackCM_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
Editingcardcs_Load(object sender, EventArgs e)	private	void	Описывает события при загрузке текущего окна

***Class SearchCard*** - Класс формы поиска медицинской карты. (таблица 13)

Таблица 13 - Данные-элементы класса SearchCard

Элементы класса	Спецификатор доступа	Тип данных	Назначение
<b>Поля</b>			
excel	private	Excel	Переменная для работы с файлом excel
cardMenu	private	CardMenu	Переменная для работы с окном главного меню
medicalCard	private	MedicalCard	Переменная для работы с файлом excel
formtimer	private	Timer	Таймер для проверки текущего окна

countmas	private	Lsit	Список для хранения найденных медицинских карт
rowCount	private	int	Количество медицинских карт в excel файле
<b>Методы</b>			
searchingcard_Click(object sender, EventArgs e)	private	void	поиск карты по введенным данным
SSName_KeyPress(object sender, KeyPressEventArgs e)	private	void	Ограничения для вводимых значений
SFName_KeyPress(object sender, KeyPressEventArgs e)	private	void	Ограничения для вводимых значений
SMName_KeyPress(object sender, KeyPressEventArgs e)	private	void	Ограничения для вводимых значений
SDoB_KeyPress(object sender, KeyPressEventArgs e)	private	void	Ограничения для вводимых значений
SSName_Validating(object sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
SFName_Validating(object sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
SMName_Validating(object sender, CancelEventArgs e)	private	void	если введена строка с маленькой буквы, делает первую букву заглавной
FormTimer_Tick(object sender, EventArgs e)	private	void	Проверка текущего окна на корректность

SearchCard_Load(object sender, EventArgs e)	private	void	Описывает события при загрузке текущего окна
hideForm_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
hideForm_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки
hideForm_Click(object sender, EventArgs e)	private	void	Свернуть окно
BacktoMenu_MouseEnter(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при наведении мыши на кнопку
BacktoMenu_MouseLeave(object sender, EventArgs e)	private	void	функция содержащая в себе определенные события при отведении мыши от кнопки

## Взаимосвязь классов

Класс Info – абстрактный базовый класс. Класс содержит в себе свойства и метод для работы с информацией о специалисте/пациенте. Классы Patient и Specialist унаследованы от него, класс MedicalCard унаследован от класса Patient. В классах Patient и Specialist определены дополнительные свойства необходимые для пациента и специалиста соответственно. В классе MedicalCard описаны дополнительные поля, свойства и методы необходимые для добавления, редактирования и записи в файл данных. (рисунок 1)

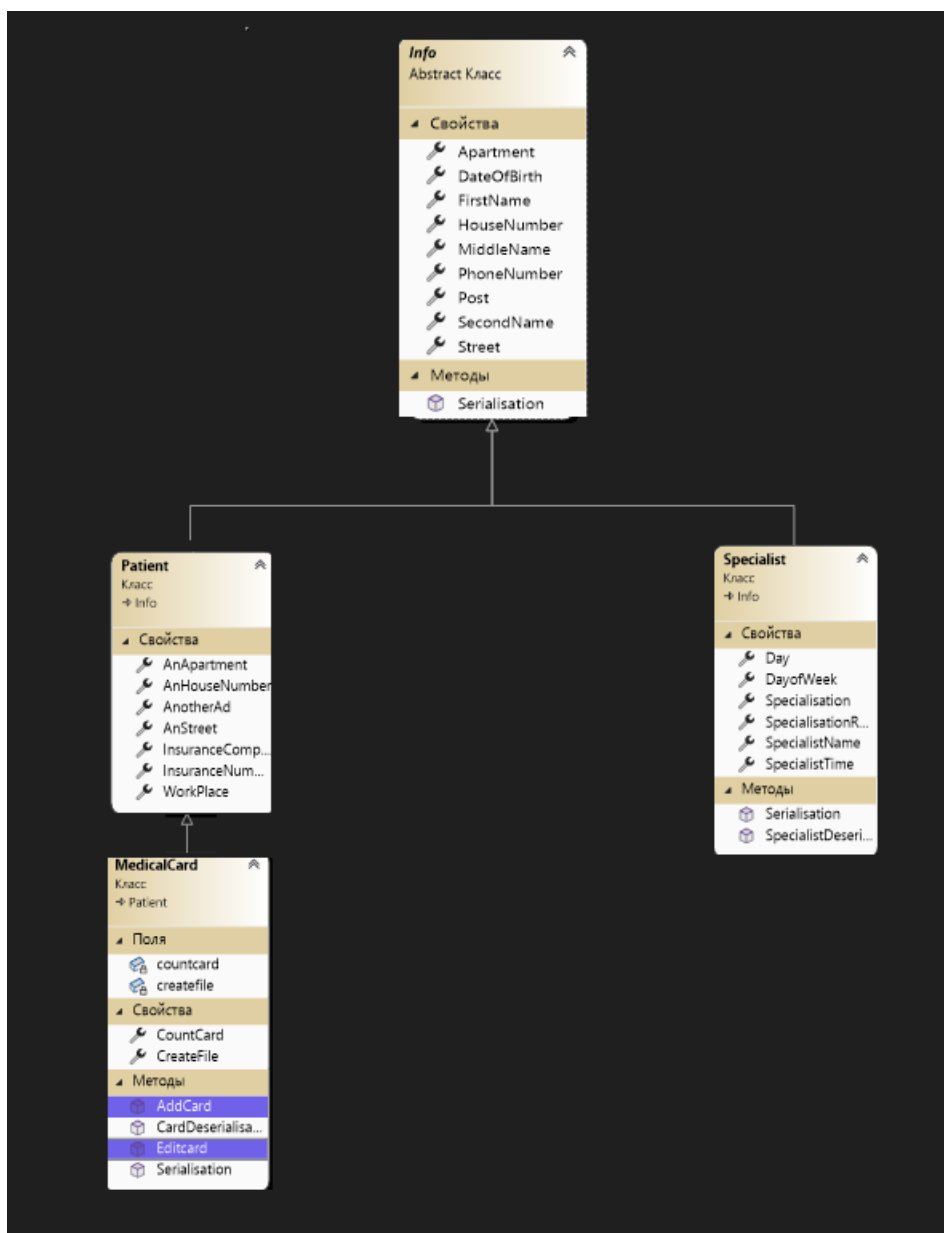


Рисунок 1 – Диаграмма классов Info, Patient, Specialist, MedicalCard.

Класс Excel предназначен для работы с файлами excel – открытие, закрытие, создание файла, запись данных в файл, чтение данных из файла. (рисунок 2)

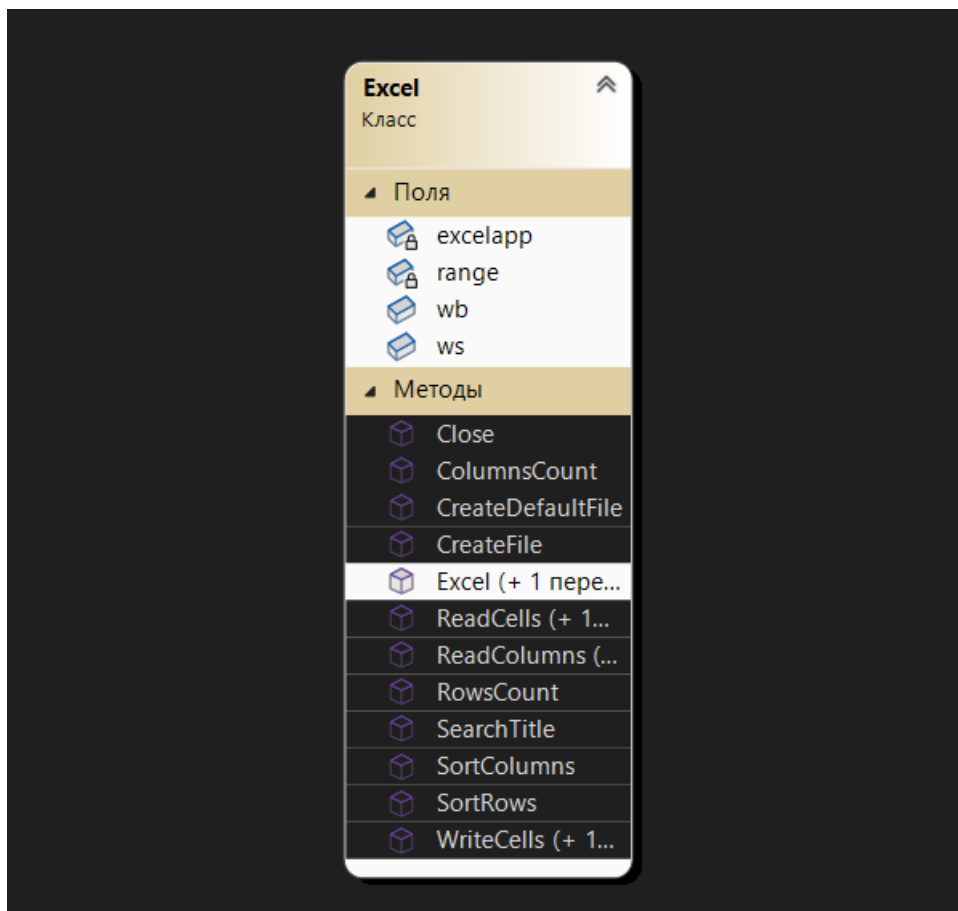


Рисунок 2 – Диаграмма класса Excel.

Класс Cardmenu предназначен для отображения окна меню медицинских карт. Содержит поля и методы для открытия окон с поиском/ записью/ редактированием медицинской карты. (рисунок 3)

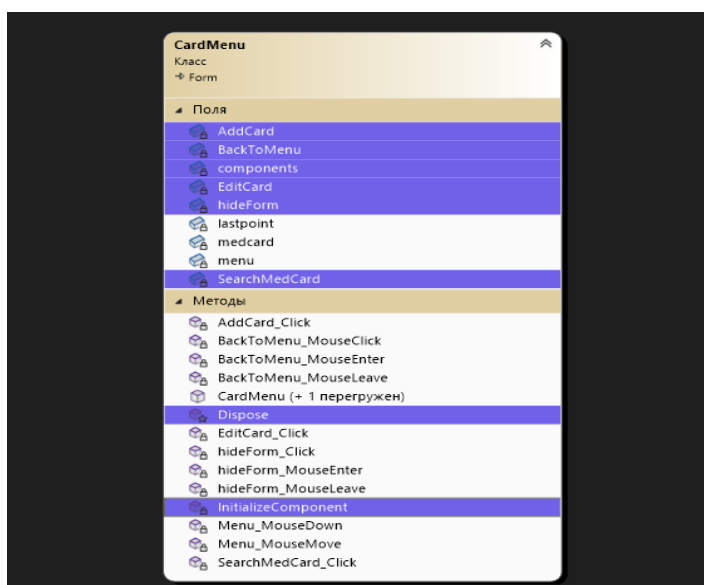


Рисунок 3 – Диаграмма класса CardMenu.



Класс Editingcards предназначен для открытия окна редактирования медицинской карты. (рисунок 4)

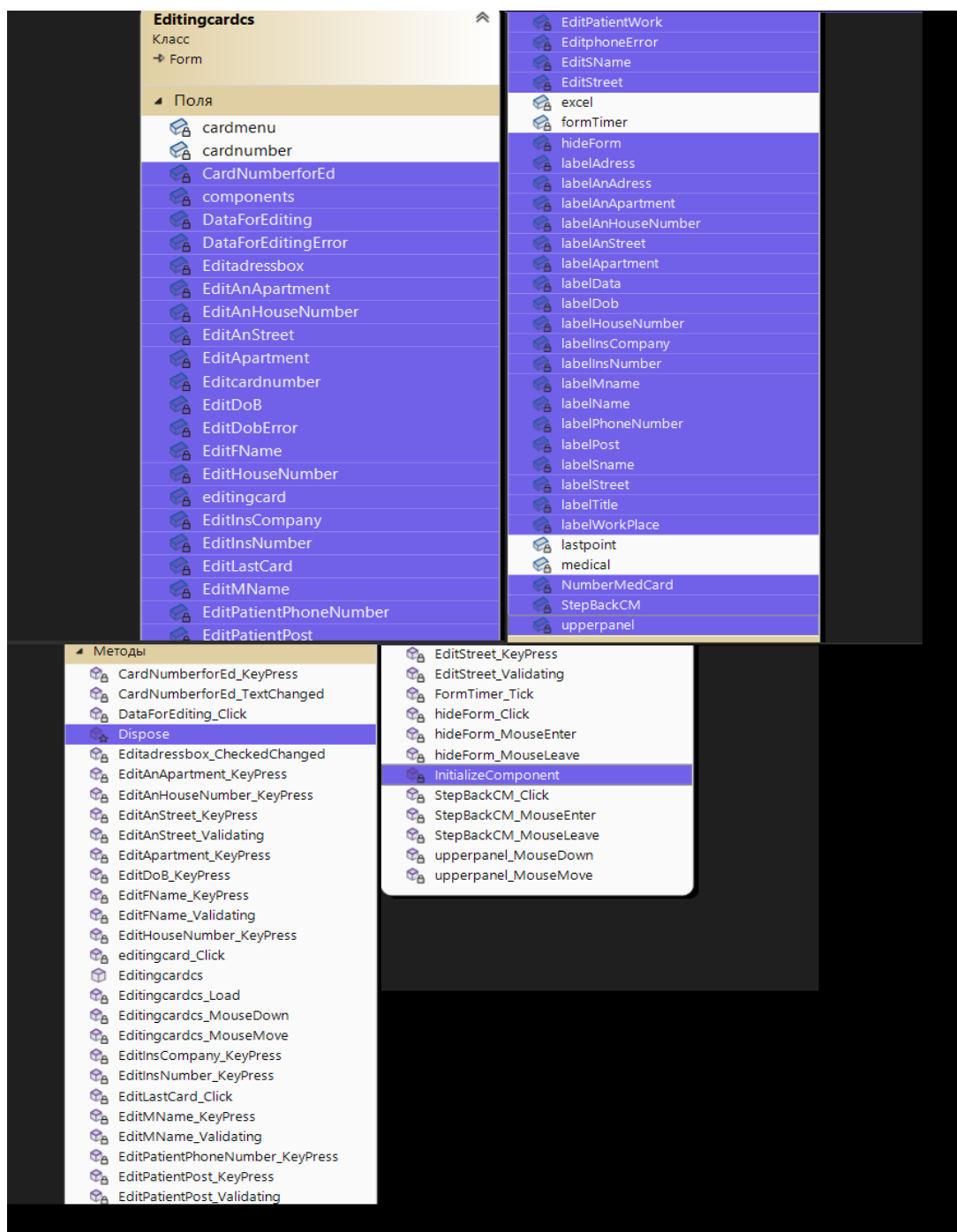


Рисунок 4 – Диаграмма класса Editingcards.

Класс Menu предназначен для открытия главного меню. В нем описываются основные методы и поля, а также взаимодействие с другими окнами. (рисунок 5)

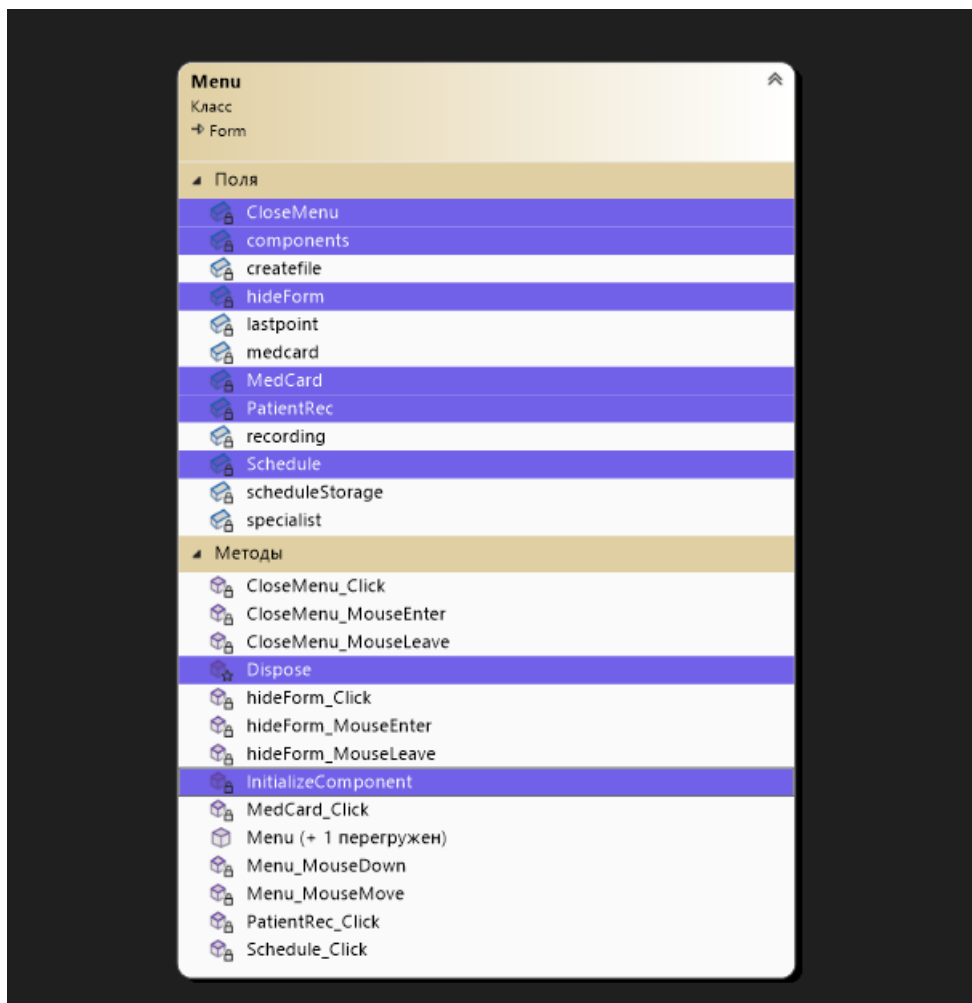


Рисунок 5 – Диаграмма класса Menu.

Класс Recording предназначен для открытия окна для записи пациентов на прием.  
(рисунок 6)

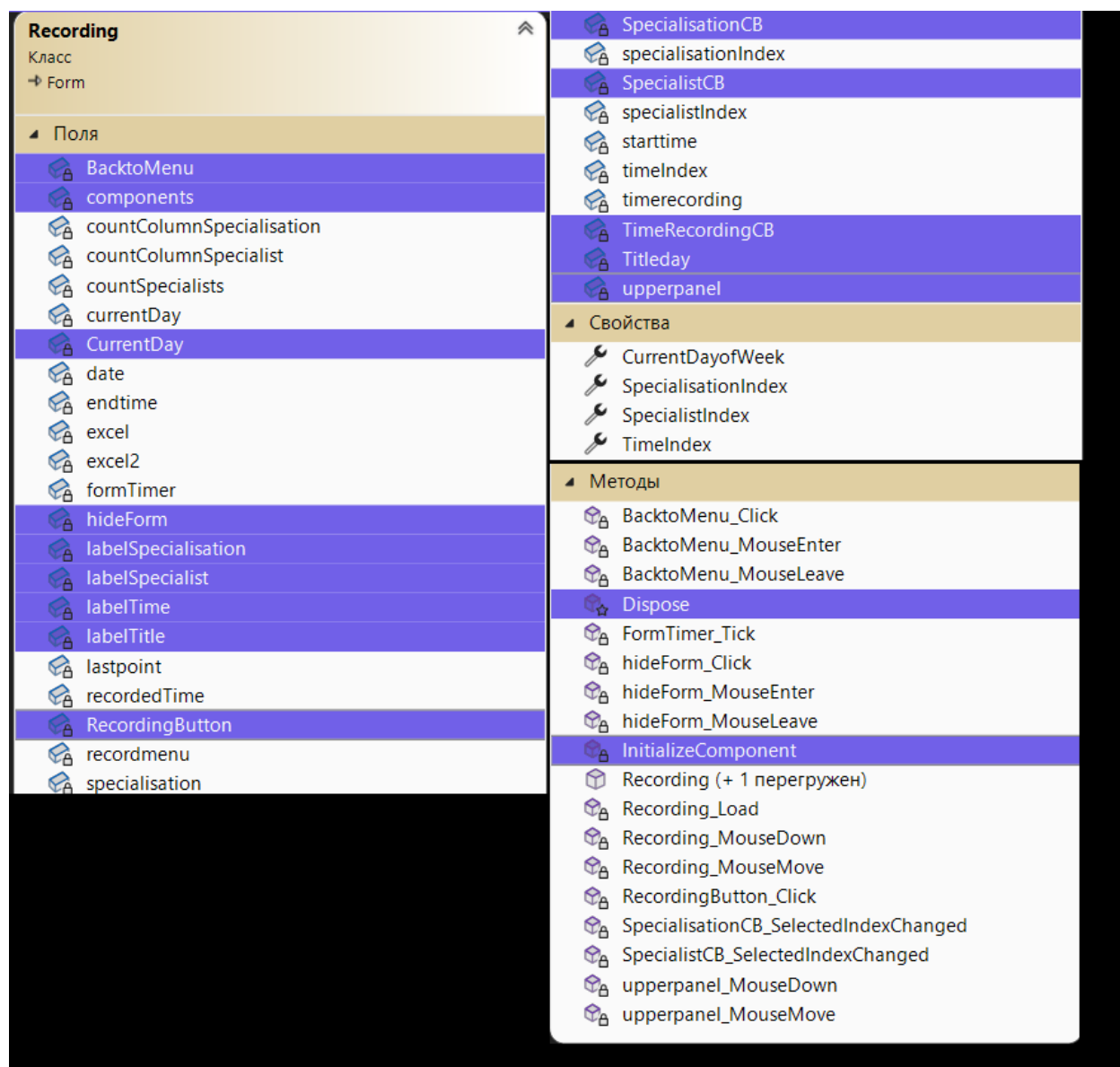


Рисунок 6 – Диаграмма класса Recording.

Класс SearchCard предназначен для открытия окна поиска медицинской карты (рисунок 7)

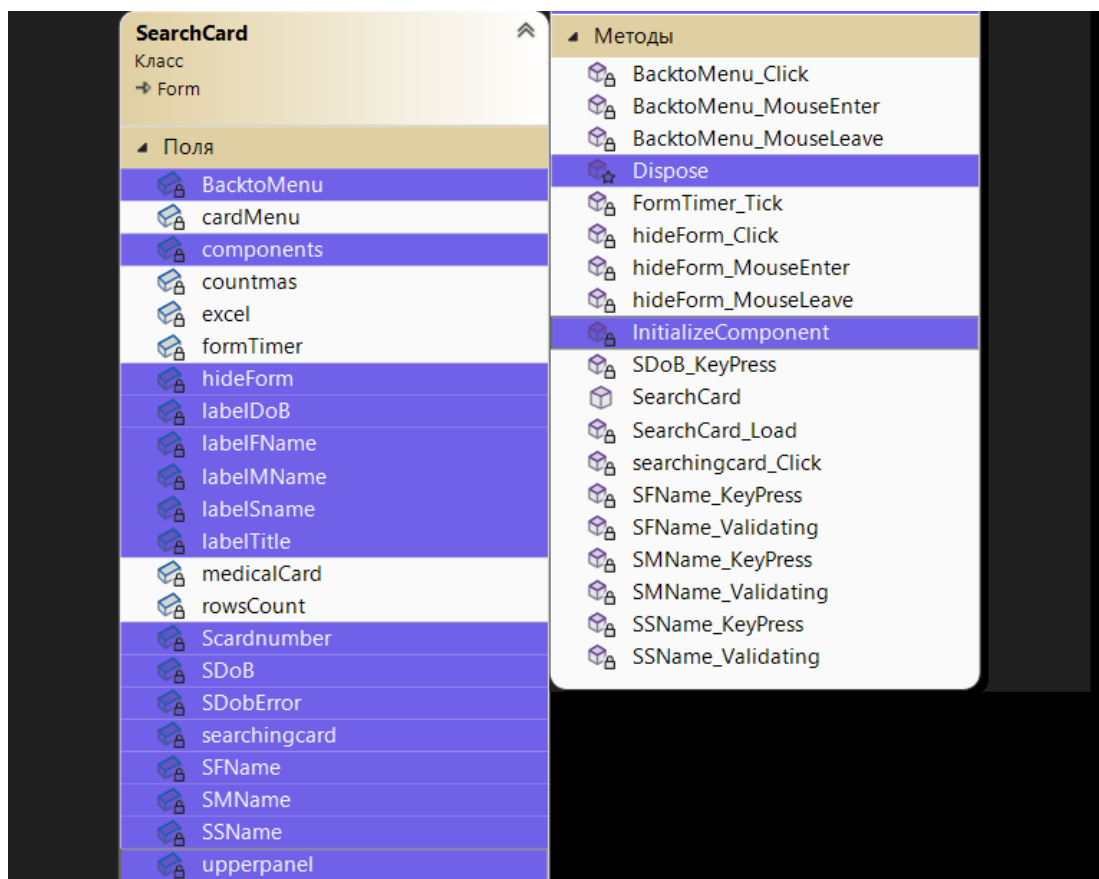


Рисунок 7 – Диаграмма класса SearchCard.

Класс ScheduleStorage предназначен для открытия окна просмотра расписания специалистов (рисунок 8)

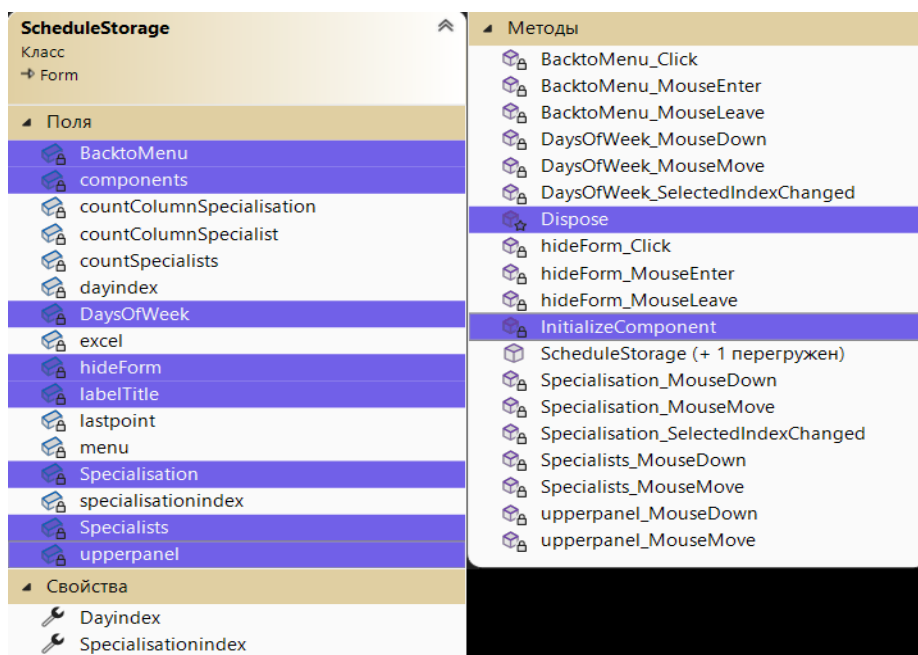


Рисунок 8 – Диаграмма класса ScheduleStorage.

### **Описание разработанного приложения (руководство пользователя).**

Данное приложение предназначено для облегчения работы регистратора поликлиники. Регистратору предоставляется возможность просматривать время работы специалиста в любой день недели, что поможет быстро ответить пациенту, в случае обращения в регистратуру. Возможность записи на прием с последующим сохранением времени, на которое проведена запись позволяет в быстром доступе получить незанятое время определенного специалиста. Создание медицинской карты пациенту позволит в дальнейшем обратиться к этой медицинской карте для внесения изменений. Поиск медицинской карты по имени, фамилии, отчеству и дате рождения позволяет просмотреть всю информацию о пациенте и соответствующий ему номер медицинской карты.

При запуске программы пользователя встречает главное меню (рисунок 9)

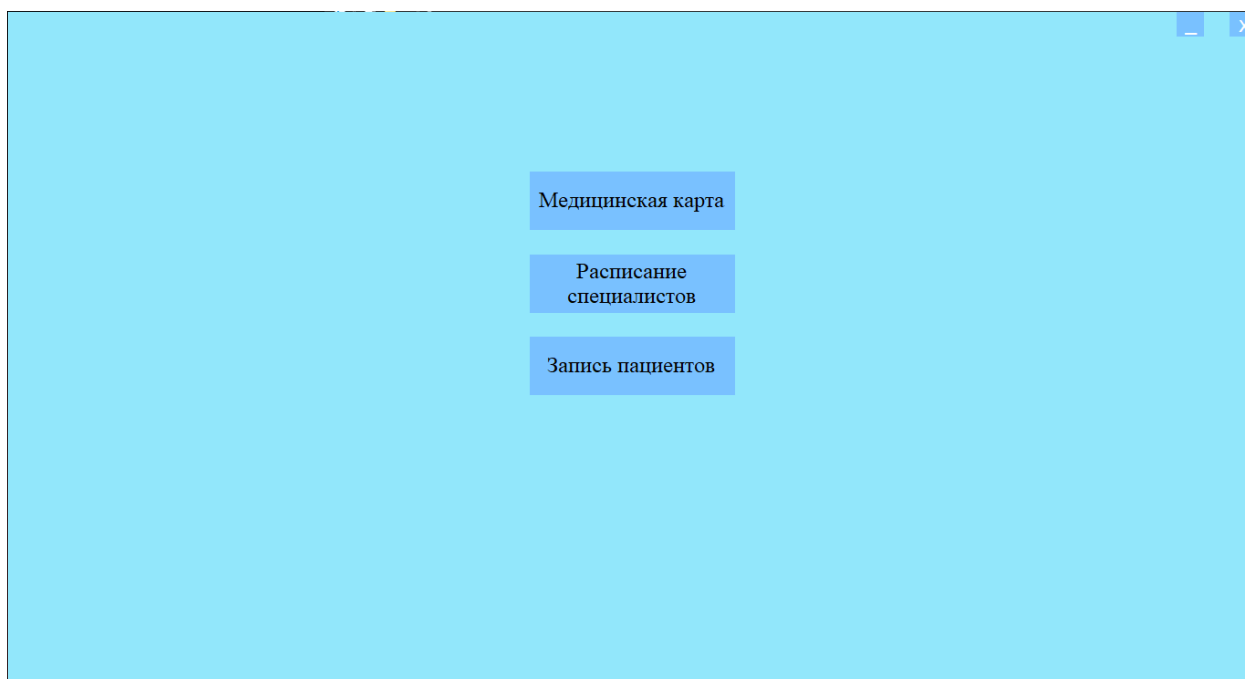


Рисунок 9 – Главное меню.

При нажатии на кнопку «Медицинская карта» открывается меню медицинских карт (рисунок 10)

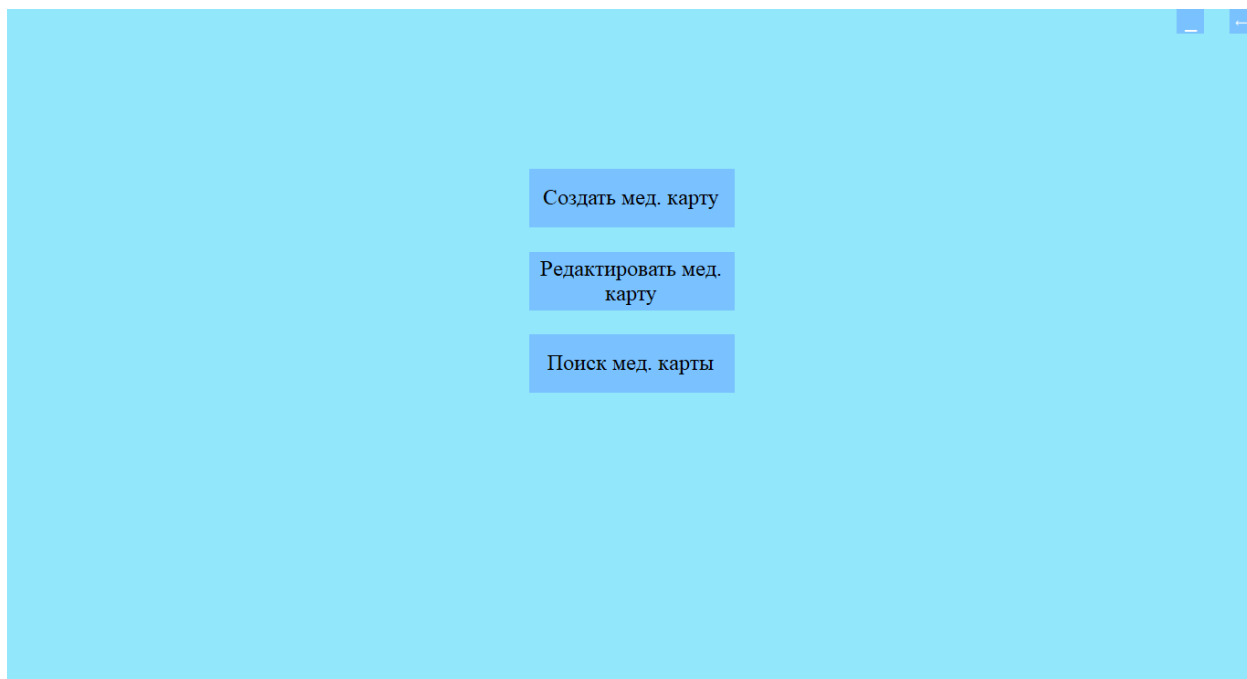


Рисунок 10 – Меню медицинских карт.

При нажатии на кнопку «Создать мед. карту» открывается окно создания медицинской карты, где есть поля для ввода значений и номер добавляемой медицинской карты, кнопка «Добавить» станет доступной после полного заполнения полей (рисунок 11)

Рисунок 11 – Окно создания медицинской карты.

Если адрес проживания по прописке совпадает с фактическим адресом проживания пациента, нужно нажать на белый квадрат «как в прописке» для автоматического заполнения полей «Адрес проживания» (рисунок 12)

Создание медицинской карты

Медицинская карта № 1

Фамилия

Имя

Отчество

Дата рождения (дд.мм.гггг)

Адрес прописки: ул.  дом  кв.

Адрес проживания: ул.  дом  кв.

☐ как в прописке

Очистить форму

Рисунок 12 – Автозаполнение фактического адреса проживания.

После полного заполнения формы можете добавить медицинскую карту, нажав кнопку «Добавить», вылезет окно подтверждения добавления (рисунок 13)

Создание медицинской карты

Медицинская карта № 1

Фамилия

Имя

Отчество

Дата рождения (дд.мм.гггг)

Адрес прописки: ул.  дом  кв.

Адрес проживания: ул.  дом  кв.

☒ как в прописке

Страховая компания  № полиса

Телефон

Место работы  Должность

Добавить

Подтверждение ввода

Добавить медицинскую карту?

Да Нет

Рисунок 13 – Подтверждение добавления.

В любой момент можно очистить все заполненные поля, нажав кнопку «Очистить форму» (рисунок 14)

The image shows two identical screenshots of a web form titled 'Создание медицинской карты' (Creating a medical card). The form is for 'Медицинская карта № 1' (Medical card No. 1). It contains several input fields: 'Фамилия' (Surname) with 'Иванов' entered, 'Имя' (Name) with 'Иван' entered, 'Отчество' (Patronymic), 'Дата рождения (дд.мм.гггг)' (Date of birth), 'Адрес прописки' (Registered address) with fields for street, house, and apartment, 'Адрес проживания' (Residence address) with similar fields, a checkbox 'как в прописке' (as in registration), 'Страховая компания' (Insurance company), '№ полиса' (Policy number), 'Телефон' (Phone), 'Место работы' (Place of work), and 'Должность' (Position). There are two blue buttons: 'Очистить форму' (Clear form) and 'Добавить' (Add). An orange arrow points to the 'Очистить форму' button in the top screenshot.

Рисунок 14 – Очистка данных.

Для того, чтобы свернуть текущее или вернуться в предыдущее окно (в случае если это главное меню – закрыть приложение) обратите внимание на верхний правый угол окна (рисунок 15)

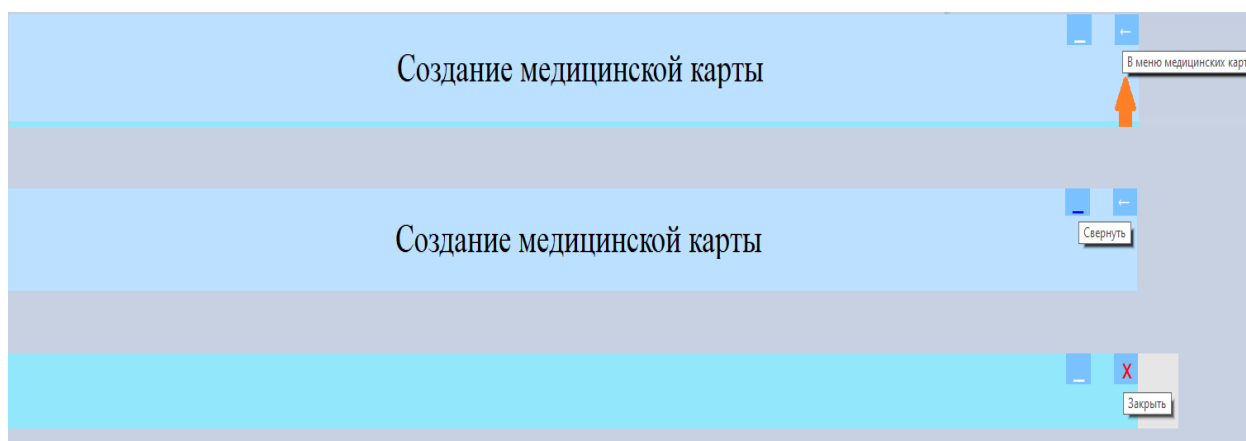


Рисунок 15 – Управление окном.



При нажатии на кнопку «Редактировать мед.карту» в меню медицинских карт вы попадете в окно редактирования медицинских карт (рисунок 16)

Редактирование медицинской карты

Медицинская карта №

Введите номер медицинской карты

Получить данные

Редактировать последнюю добавленную карту

Фамилия

Имя

Отчество

Дата рождения (дд.мм.гггг)

Адрес прописки: ул. дом кв.

Адрес проживания: ул. дом кв.

☐ как в прописке

Страховая компания № полиса

Телефон

Место работы Должность

Редактировать

Рисунок 16 – Редактирование медицинской карты.

Здесь можно получить последнюю добавленную карту нажав на кнопку «Редактировать последнюю добавленную карту» (рисунок 17)

Редактирование медицинской карты

Медицинская карта № 1

Введите номер медицинской карты

Получить данные

Редактировать последнюю добавленную карту

Фамилия Иванов

Имя Василий

Отчество Евгеньевич

Дата рождения (дд.мм.гггг) 20.12.2001

Адрес прописки: ул. Крылова дом 44а кв. 23

Адрес проживания: ул. Крылова дом 44а кв. 23

☐ как в прописке

Страховая компания Росгосстрах № полиса 429032131

Телефон 8-900-100-10-10

Место работы Нгту Должность Студент

Редактировать

Рисунок 17 – Редактирование последней добавленной карты.

Нажав кнопку «Редактировать» медицинская карта будет отредактирована, а форма очистится

Также можно добавить медицинскую карту для редактирования по ее номеру введя номер в поле и нажав на кнопку «Получить данные» (рисунок 18)

The image shows two screenshots of a web application window titled "Редактирование медицинской карты" (Editing medical card). The top screenshot shows the initial state where the user is prompted to enter the card number. The bottom screenshot shows the card details after clicking "Получить данные" (Get data).

**Top Screenshot: Adding card by number**

- Title: Редактирование медицинской карты
- Section: Медицинская карта №
- Fields: Фамилия, Имя, Отчество, Дата рождения (дд.мм.гггг)
- Input: Введите номер медицинской карты (1)
- Buttons: Получить данные, Редактировать последнюю добавленную карту

**Bottom Screenshot: Card details**

- Title: Редактирование медицинской карты
- Section: Медицинская карта № 1
- Fields: Фамилия (Иванов), Имя (Василий), Отчество (Евгеньевич), Дата рождения (дд.мм.гггг) (20.12.2001)
- Address: Адрес прописки: ул. Крылова, дом 44а, кв. 23; Адрес проживания: ул. Крылова, дом 44а, кв. 23
- Insurance: Страховая компания (Росгосстрах), № полиса (429032131)
- Contact: Телефон (8-900-100-10-10)
- Work: Место работы (Нгу), Должность (Студент)
- Buttons: Редактировать

Рисунок 18 – добавление медицинской карты для редактирования по ее номеру.

При нажатии на кнопку «Поиск мед. карты» в меню медицинских карт вы попадете в окно поиска медицинских карт (рисунок 19)

The image shows a web application window titled "Поиск медицинской карты" (Search medical card). It contains search fields for personal data and a search button.

**Search Fields:**

- Фамилия
- Имя
- Отчество
- Дата рождения (дд.мм.гггг)

**Buttons:** Поиск

Рисунок 19 – Окно поиска мед. карты.

Для того чтобы найти медицинскую карту заполните все поля и нажмите кнопку «Поиск» (рисунок 20)

Фамилия

Имя

Отчество

Дата рождения (дд.мм.гггг)

**Поиск**

**Информация о пациенте** X

1/1

Медицинская карта №1  
 Фамилия: Иванов  
 Имя: Василий  
 Отчество: Евгеньевич  
 Дата рождения: 20.12.2001  
 Адрес проживания по прописке:  
 Улица: Крылова, Дом: 44а, Квартира: 23  
 Фактический адрес проживания:  
 Улица: Крылова, Дом: 44а, Квартира: 23  
 Страховая компания: Росгосстрах  
 Номер страховки: 429032131  
 Номер телефона: 8-900-100-10-10  
 Место работы: Нгту  
 Должность: Студент

**OK**

Рисунок 20 – Поиск мед. карты.

Из главного меню нажав кнопку «Расписание специалистов» вы попадете в окно с расписанием специалистов (рисунок 21)

Расписание работы врачей		
Понедельник	Андролог	
Вторник	Артролог	
Среда	Врачи-МРТ	
Четверг	Гинеколог-Акушер	
Пятница	Дерматовенеролог	
Суббота	Диабетолог	
Воскресенье	Кардиолог	
	Косметолог	
	Логопед	
	ЛОР	
	Маммолог	
	Мануальный терапевт	
	Миколог	
	Нарколог	
	Невролог	
	Нейрохирург	
	Ортопед	
	Офтальмолог	
	Паразитолог	
	Проктолог	
	Терапевт	
	Травматолог	
	УЗИ-специалист	
	Уролог	
	Физиотерапевт	
	Фтизиатр	
	Хирург	

Рисунок 21 – Окно расписания специалистов.

Для просмотра расписания выберите день недели нажав на нужный день мышью, затем выберите специальность. Здесь вы увидите ФИО специалиста его рабочее время в этот день и в скобках номер кабинета, в котором он принимает. (рисунок 22)

Расписание работы врачей		
Понедельник	Андролог	№ участка 3 Бернадо Алексей Григорьевич 14:00-20:00 (216)
Вторник	Артролог	№ участка 2 Буркатская Татьяна Сергеевна 14:00-20:00 (212)
Среда	Врач-МРТ	№ участка 1 Ильин Иван Егорович 14:00-20:00 (214)
Четверг	Гинеколог-Акушер	№ участка 1 Седова Анна Евгеньевна 8:00-15:00 (214)
Пятница	Дерматовенеролог	№ участка 2 Сушкова Галина Борисовна 8:00-15:00 (212)
Суббота	Диабетолог	№ участка 3 Юн Валентина Викторовна 8:00-15:00 (216)
Воскресенье	Кардиолог	
	Косметолог	
	Логопед	
	Лор	
	Маммолог	
	Мануальный терапевт	
	Миколог	
	Нарколог	
	Невролог	
	Нейрохирург	
	Ортопед	
	Офтальмолог	
	Паразитолог	
	Проктолог	
	Терапевт	
	Травматолог	
	УЗИ-специалист	
	Уролог	
	Физиотерапевт	
	Фтизиатр	
	Хирург	

Рисунок 22 – Расписание специалистов.

Из главного меню нажмите кнопку «Запись пациентов», вы попадете в окно записи пациентов на прием. (рисунок 23)

Запись на прием		
Направление	Специалист	Время записи
<input type="text"/>	<input type="text"/>	<input type="text"/>
<div>Записать</div> <div>День недели Вторник</div>		

Рисунок 23 – Окно записи пациентов на прием.

Наведясь на стрелку нажмите левой кнопкой мыши и выберите нужную специальность, аналогично с выбором специалиста и времени записи (рисунок 24)

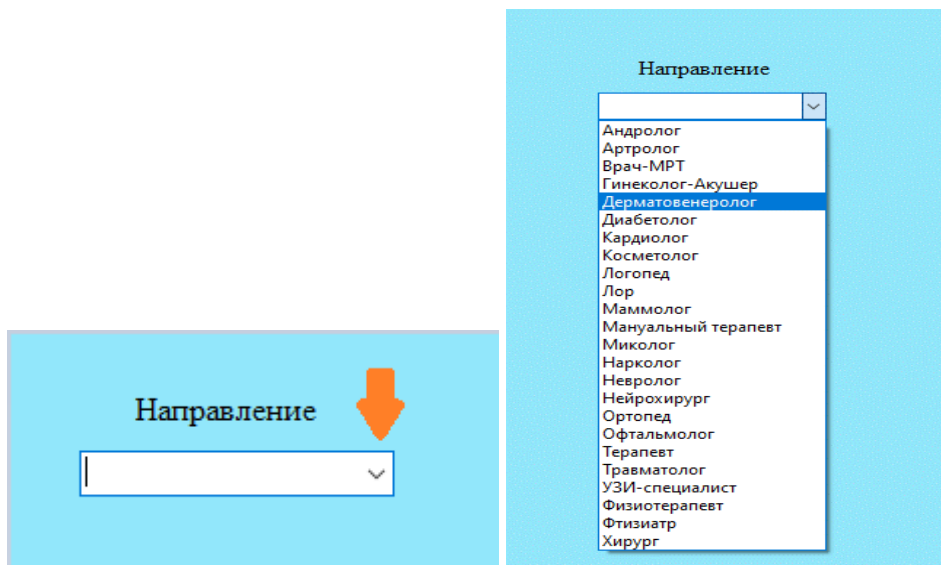


Рисунок 24 – Подготовка к записи пациента.

После того, как выбран специалист и время записи, нажмите кнопку «Записать» для записи пациента и подтвердите запись (рисунок 25)

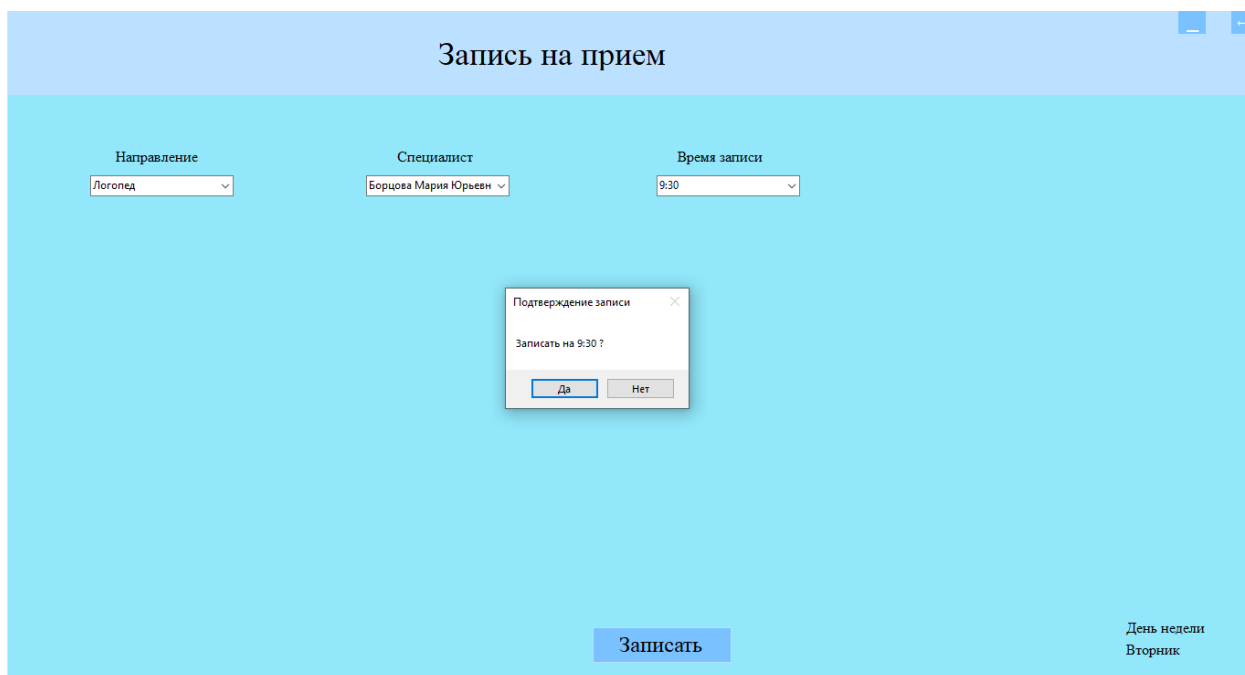


Рисунок 25 – Запись пациента на прием

## Тестирование

Тестирование программы показало корректную работу, а также правильную реакцию на исключительные ситуации.

При попытке поиска пациента, если до этого нигде не добавлялся пациент выведется информационное сообщение (рисунок 26)

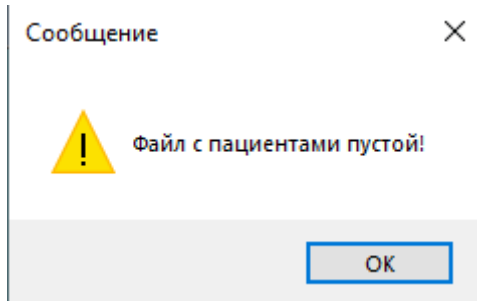


Рисунок 26 – Сообщение об отсутствии пациентов.

Если пациент не найден среди имеющихся выведется информационное сообщение (рисунок 27)

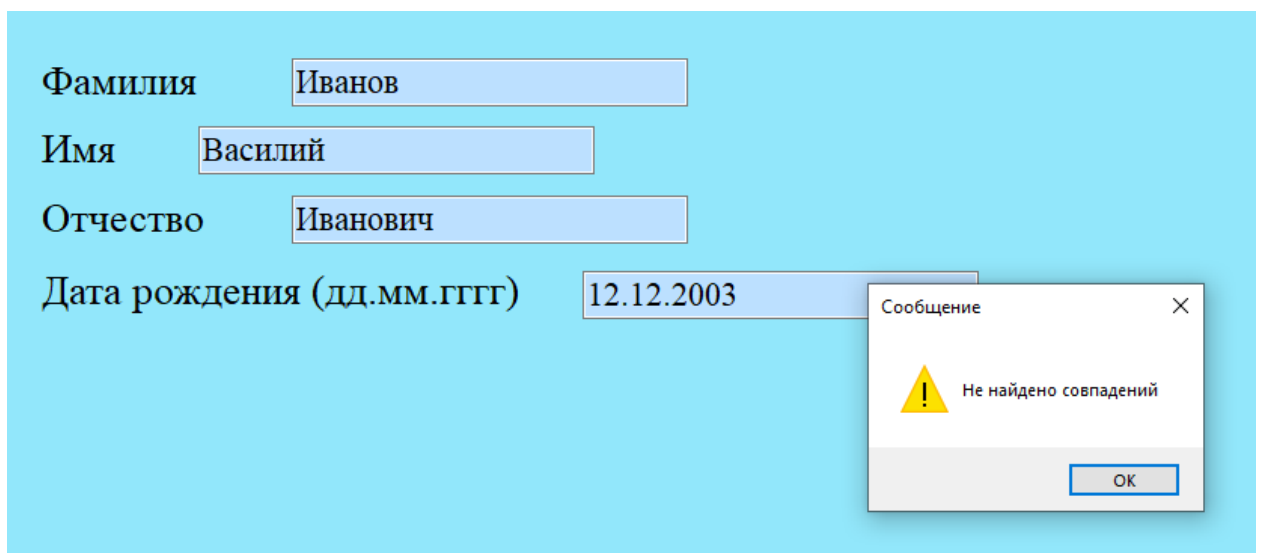


Рисунок 27 – Сообщение о том, что пациент не найден.

При вводе данных для добавления/редактирования/поиска мед. карты нельзя нажать кнопку добавить/редактировать/поиск, если что-то заполнено некорректно. Для добавления и редактирования медицинской карты - это дата рождения и номер телефона (рисунок 28, 29).

The screenshot shows a form for adding a medical card. The following fields contain data that triggers validation errors:

- Дата рождения (дд.мм.гггг):** 20.12.2023. A red error message below the field reads: "Введите реальную дату рождения" (Enter a real date of birth).
- Телефон:** 8-913-987-2323. A red error message below the field reads: "Неверный формат ввода номера телефона. Пример ввода: 8-900-100-10-10" (Incorrect phone number input format. Example input: 8-900-100-10-10).

Other fields are filled with: Фамилия: Чайка, Имя: Павел, Отчество: Андреевич, Адрес прописки: ул. Немировича-Данченко, дом 134, кв. 28, Адрес проживания: ул. Немировича-Данченко, дом 134, кв. 28, Страховая компания: Росгосстрах, № полиса: 429012314, Место работы: Нгту, Должность: Студент. Buttons include "Очистить форму" (Clear form) and "Добавить" (Add).

Рисунок 28 – Исключительная ситуация при вводе информации для добавления медицинской карты.

The screenshot shows a form for editing a medical card, titled "Медицинская карта № 1". The following fields contain data that triggers validation errors:

- Дата рождения (дд.мм.гггг):** 19.10.200. A red error message below the field reads: "Некорректный ввод даты рождения. Пример ввода: 19.10.2003" (Incorrect date of birth input. Example input: 19.10.2003).
- Телефон:** 8-901-910-32-3. A red error message below the field reads: "Неверный формат ввода номера телефона. Пример ввода: 8-900-100-10-10" (Incorrect phone number input format. Example input: 8-900-100-10-10).

Other fields are filled with: Фамилия: Иванов, Имя: Василий, Отчество: Игоревич, Адрес прописки: ул. Крылова, дом 11, кв. 13, Адрес проживания: ул. Крылова, дом 11, кв. 13, Страховая компания: КАСКО, № полиса: 25151515, Место работы: Нгту, Должность: Студент. Buttons include "Получить данные" (Get data), "Редактировать последнюю добавленную карту" (Edit the last added card), and "Редактировать" (Edit).

Рисунок 29 – Исключительная ситуация при вводе информации для редактирования медицинской карты.

Для поиска карты некорректный введен может быть дата рождения (рисунок 30)

Фамилия

Имя

Отчество

Дата рождения (дд.мм.гггг)

Некорректный ввод даты рождения. Пример ввода: 18.10.2003

Рисунок 30 – Исключительная ситуация при вводе информации для поиска мед карты.

Проверка даты рождения выдает несколько видов предупреждений. Ограничение на минимальный год рождения выбирался в соответствие со средним возрастом долгожителей России – 120 лет (т.е текущий год минус 120 лет) (рисунок 31)

<input type="text" value="19.10.200 "/>	<input type="text" value="19.13.2003"/>
<small>Некорректный ввод даты рождения. Пример ввода: 18.10.2003</small>	<small>Введите существующую дату</small>
<input type="text" value="19.12.1902 "/>	<input type="text" value="19.12.2023 "/>
<small>Введите реальную дату рождения</small>	<small>Введите реальную дату рождения</small>

Рисунок 31 – Виды предупреждений при вводе даты рождения.

Проверка номера телефона выдает два вида предупреждений. В России телефонные коды операторов связи начинаются с «9» (рисунок 32)

<input type="text" value="8-901-910-32-3"/>	<input type="text" value="8-701-910-32-23"/>
<small>Неверный формат ввода номера телефона. Пример ввода 8-900-100-10-10</small>	<small>Неверный оператор телефона</small>

Рисунок 32 – Виды предупреждений при вводе номера телефона.



### **Заключение**

При написании программы были использованы принципы ООП, а именно инкапсуляция, наследование и полиморфизм.

Написанная программа обладает простым и понятным интерфейсом, корректно реагирует на исключительные ситуации, обладает функционалом, необходимым для упрощения деятельности регистратора поликлиники. Программный продукт соответствует поставленной задаче и может использоваться в медицинских учреждениях.