Software Design Document

“Информационная система больницы”

Заказчик

Проработчик курса

«Технологии разработки программного обеспечения»

Савенков Иван Николаевич

Автор документа

Студентка 2 курса Физики-технического факультета ДонНУ

Группа ИВТ-1

Дегтярева Дарина Андреевна

2021

Оглавление

[1.Введение 3](#_Toc87523508)

[1.1. Цель 3](#_Toc87523509)

[1.2. Описание продукта 3](#_Toc87523510)

[2. Пользовательские интерфейсы 4](#_Toc87523511)

[3.Описание классов 11](#_Toc87523512)

[3.1. Логика приложения 11](#_Toc87523513)

[3.2. Классы графического интерфейса 11](#_Toc87523514)

# 1.Введение

## 1.1. Цель

Одной из актуальных проблем любого вида человеческой деятельности является проблема обработки увеличивающегося потока информации.

Перспективным способом решения данной проблемы является автоматизация работы с информацией, в частности, создание компьютерных баз данных, позволяющих хранить, систематизировать и обрабатывать данные.

Целью данной работы является создание информационной системы больницы (частной клиники). Она предназначена для хранения информации о врачах, пациентах клиники, вспомогательной информации о распорядке работы больницы и приеме больных, с возможностью внесения данных, выборки и изменения данных, вывода информации в необходимом формате.

## 1.2. Описание продукта

Больница - это такая организация, которая работает с очень большим объемом информации, как о сотрудниках, так и о пациентах. Врачам всегда следить за данными о своих пациентах, о курсе лечения больных. А руководству и бухгалтерии необходимо быть в курсе событий о своих сотрудниках. Для этого нужна общая система, включающая всю необходимую информацию. Программа является очень актуальной на сегодняшний день, она автоматизирует работу и предоставляет пользователю (оператору) понятный и дружественный интерфейс.

Мощность базы данных обусловлена возможностью ее постоянного пополнения новыми данными, причем в неограниченном количестве информации. Это является очень удобным для пользователя. Таким образом, создание информационной системы, обладающей такими свойствами, задача достаточно актуальная и полезная.

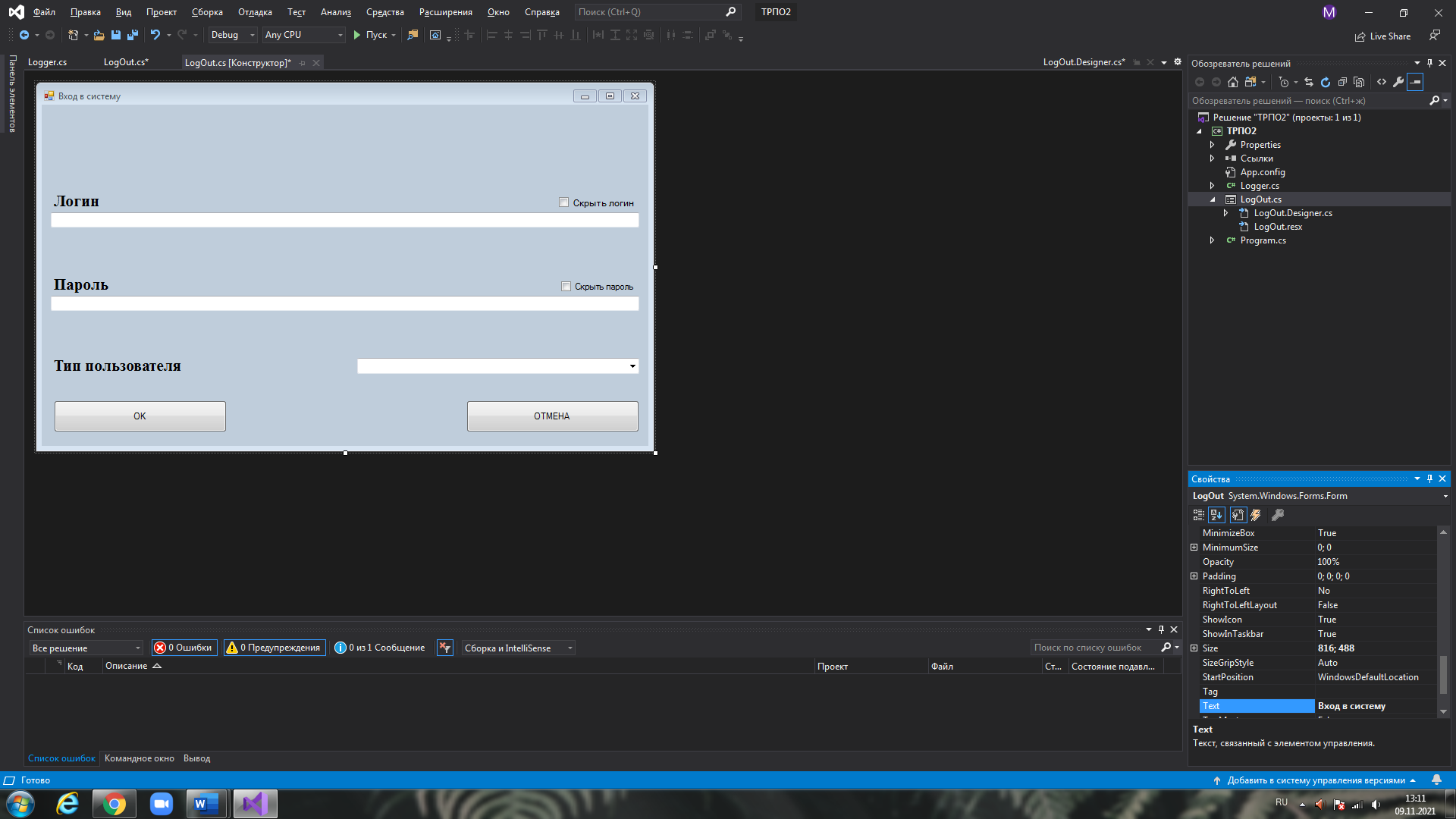
В последнее время информационные технологии стали неотъемлемой частью нашей жизни.

Начальным этапом создания системы является изучение, анализ и моделирование деятельности организации для возможного улучшения и оптимизации методов работы.

# 2. Пользовательские интерфейсы

Интерфейс разработан с разрешением экрана не менее 1366x768. Программа содержит следующие пользовательские интерфейсы

1. Окно «Входа в систему»



Функции окна:

1. Textbox1 – поле для ввода логина
2. Label1 – “Логин”
3. CheckBox1 (скрыть логин) – контроль для отображения логина или замены символов на \*
4. Textbox2 – поле для ввода логина
5. Label2 – “Пароль”
6. CheckBox2 (скрыть пароль) - контроль для отображения пароля или замены символов на \*
7. Label3 – “Тип пользователя”
8. ComboBox1 –позволяет выбрать тип пользователя : администратор или киноман
9. Button “Ок” – при нажатии открывается окно администратора или пользователя в зависимости от верности введенных логина и пароль и выбранного типа пользователя. Иначе появляется MessageBox с информацией об ошибке входа

10) Button “Отмена” – закрывает приложение

2. Главное окно администратора(регистратура)



1. Элемент MenuStrip1, который содержит пункты:

-Программа

-Пациенты

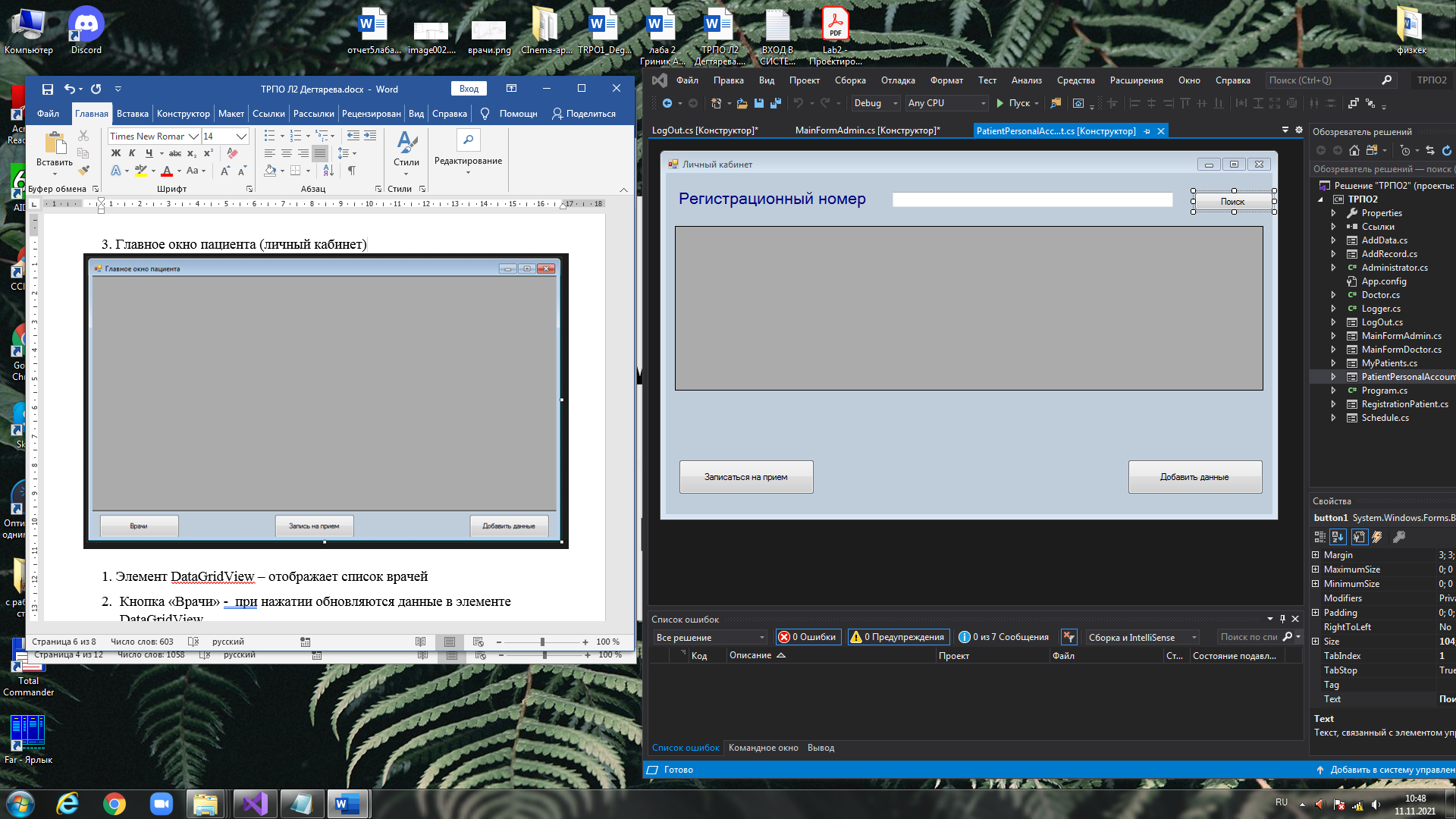
Каждый пункт содержит свои подпункты

Программы -> Выход: осуществляется закрытие приложения

Пациенты -> Регистрация пациента; Личный кабинет пациента

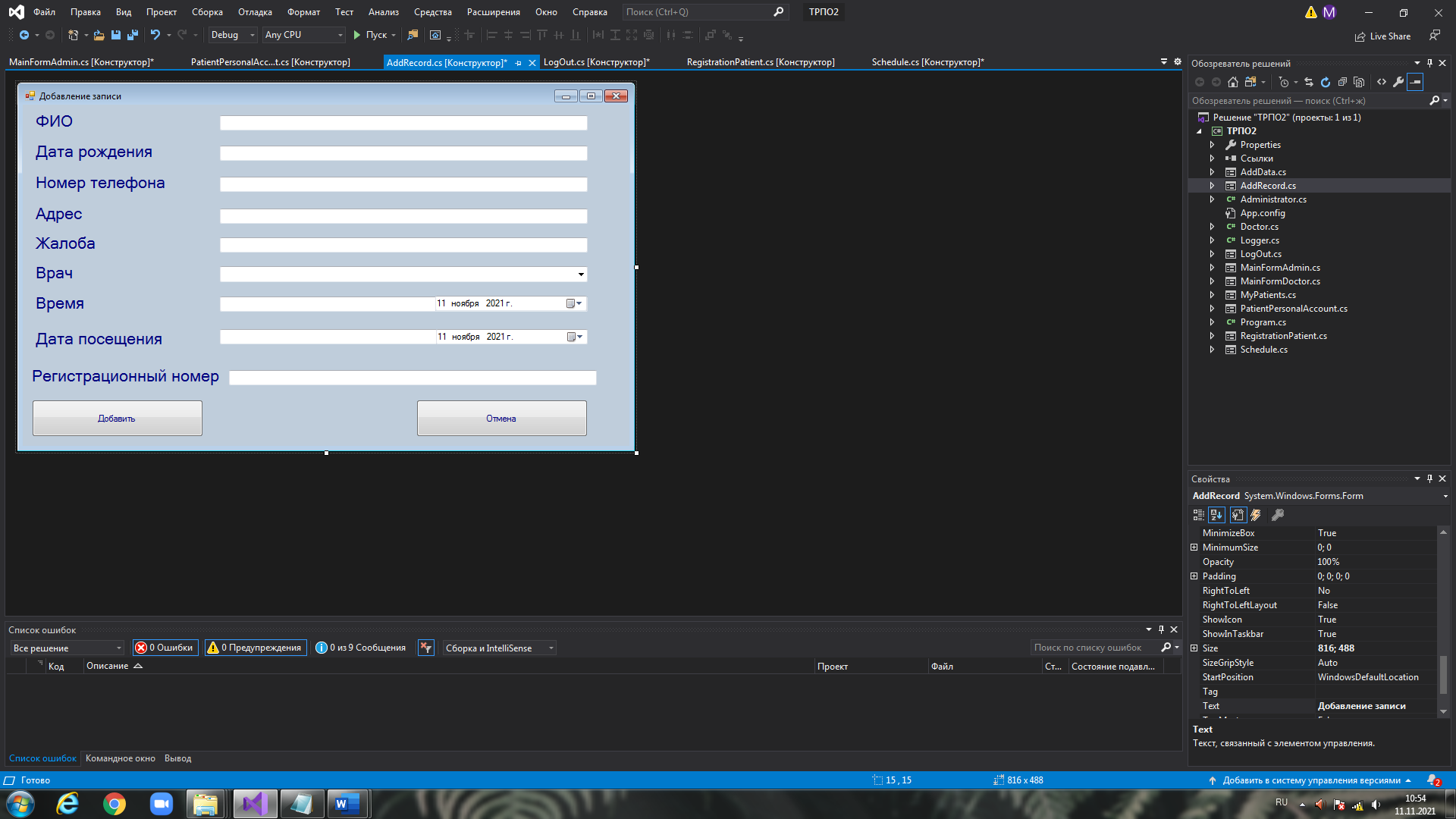
1. Button «Добавление запись» - при нажатии открывается окно «Добавление записи»
2. Button “Удаление записи” – необходимо выбрать необходимую запись в DataGridView и нажать на кнопку, после чего запись будет удалена из системы. Если запись в DataGridView не выбрана, то ничего не произойдет

3. Главное окно пациента (личный кабинет)

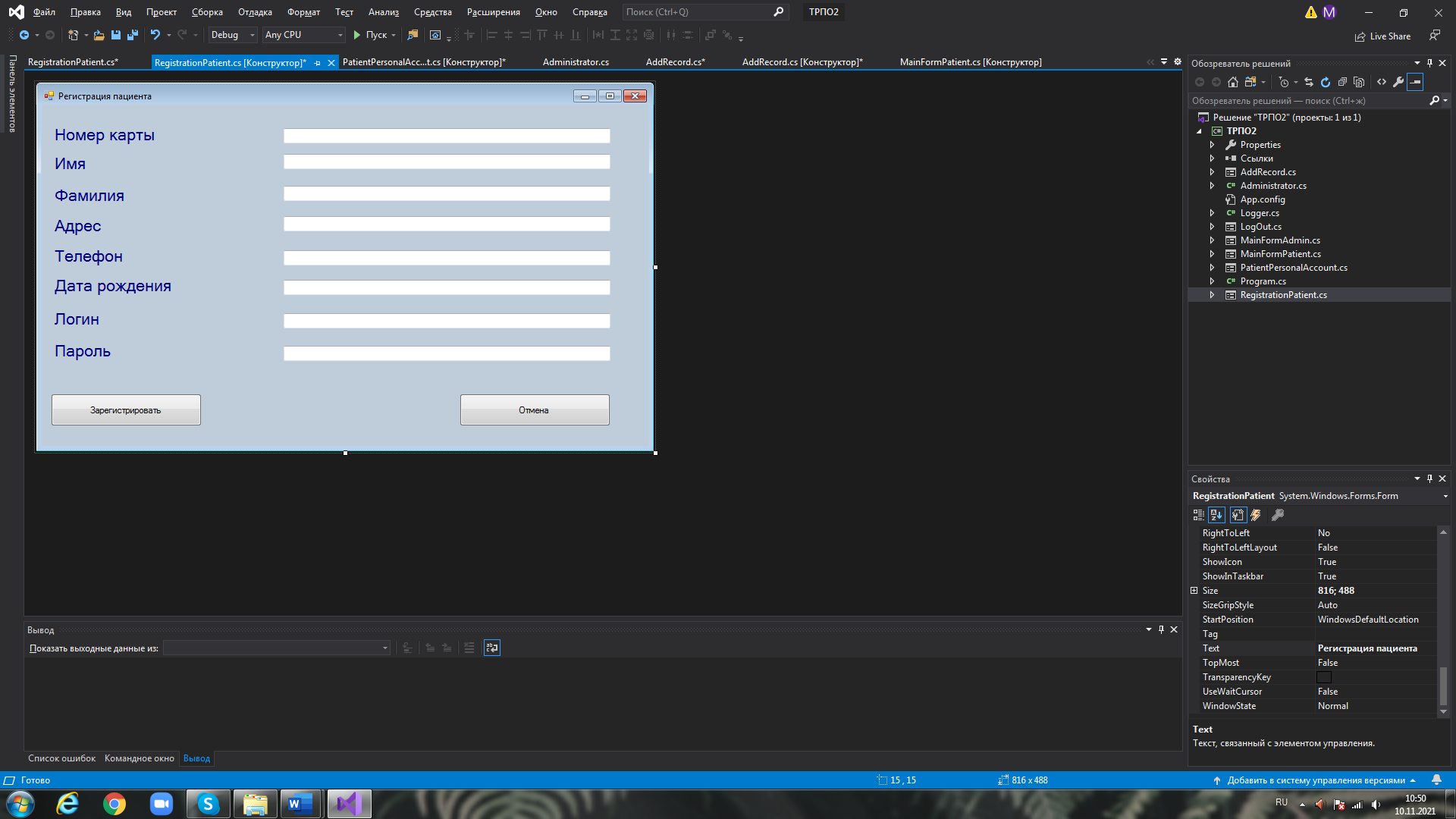


1. С помощью TextBox1 происходит поиск пациента, зарегистрированного на регистратуре.
2. Элемент DataGridView – отображает список приемов
3. Кнопка «Запись на прием» - при нажатии открывается окно «Запись на прием»
4. Кнопка «Добавить данные» - при нажатии открывается окно «Добавление данных о пациенте/анализов»

4.Окно «Добавление записи»

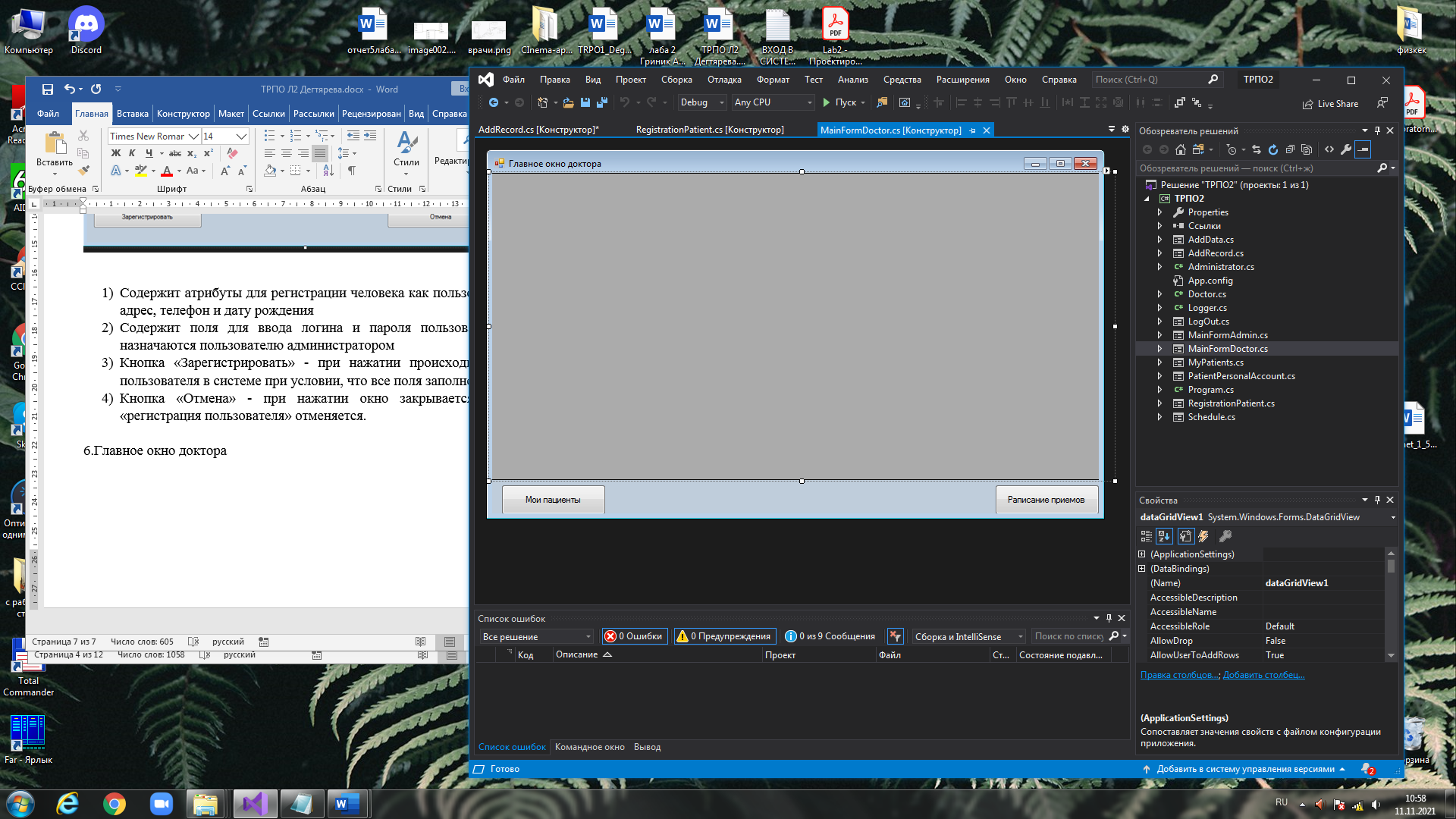


1. Содержит личные данные пациента: ФИО, дату рождения, номер телефона, адрес, на что он жалуется, выбор лечащего врача и дату посещения и textbox для ввода соответствующих значений
2. Кнопка «Добавить» - при нажатии запись добавляется в информационную систему больницы, что все поля заполнены
3. Кнопка «Отмена» - при нажатии окно закрывается, а операция «Добавление записи» отменяется.

5. Окно «Регистрация пользователя»  


1. Содержит атрибуты для регистрации человека как пользователя – ФИО, адрес, телефон и дату рождения
2. Содержит поля для ввода логина и пароля пользователя, которые назначаются пользователю администратором
3. Кнопка «Зарегистрировать» - при нажатии происходит регистрация пользователя в системе при условии, что все поля заполнены
4. Кнопка «Отмена» - при нажатии окно закрывается, и операция «регистрация пользователя» отменяется.

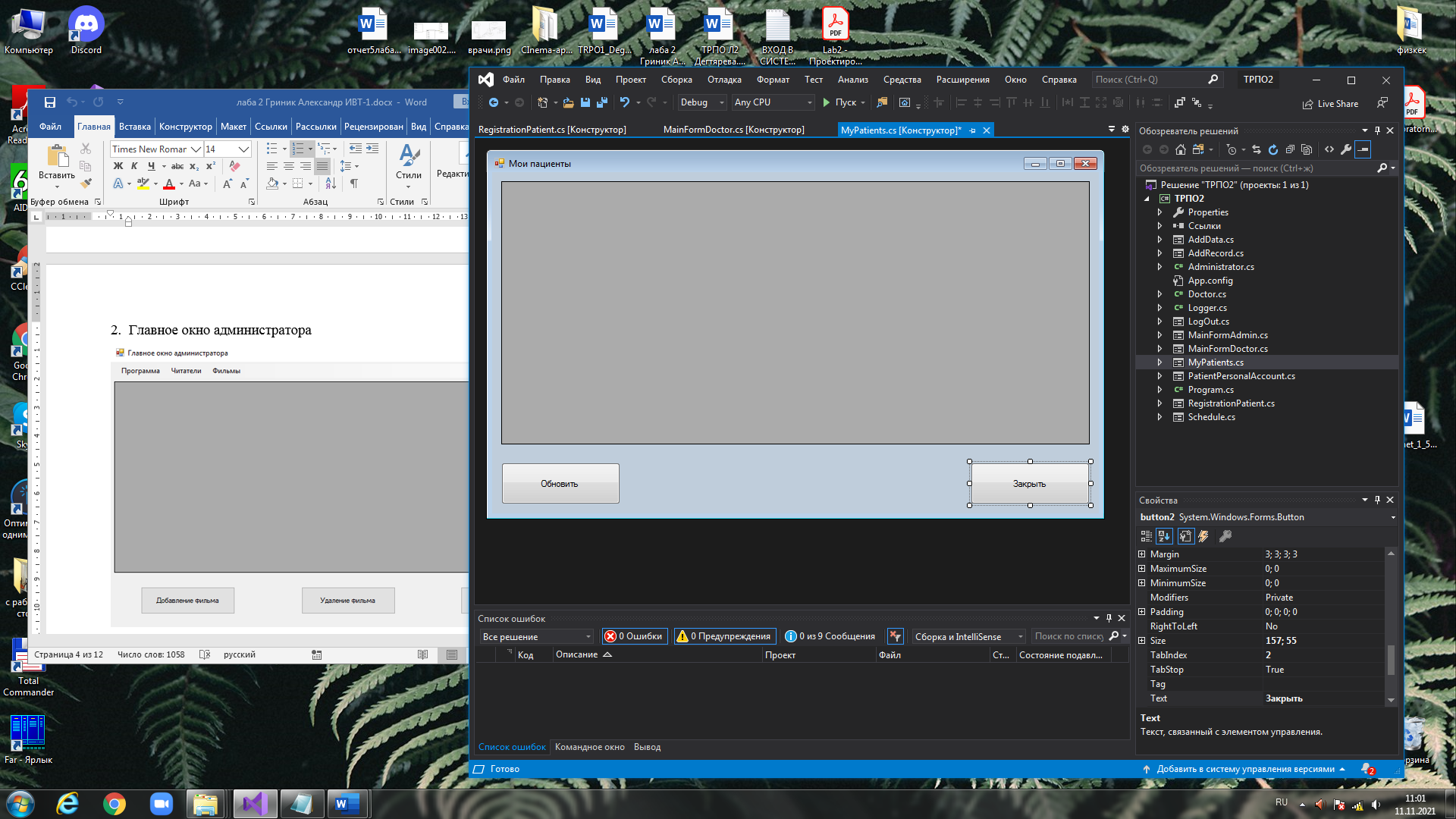
6.Главное окно доктора



1) Содержит элемент DataGridView – отображает расписание приемов при нажатии на кнопку «Расписание приемов»

2) Кнопка «Мои пациенты» делает переход на форму «Мои пациенты»

7. Форма «Мои пациенты»

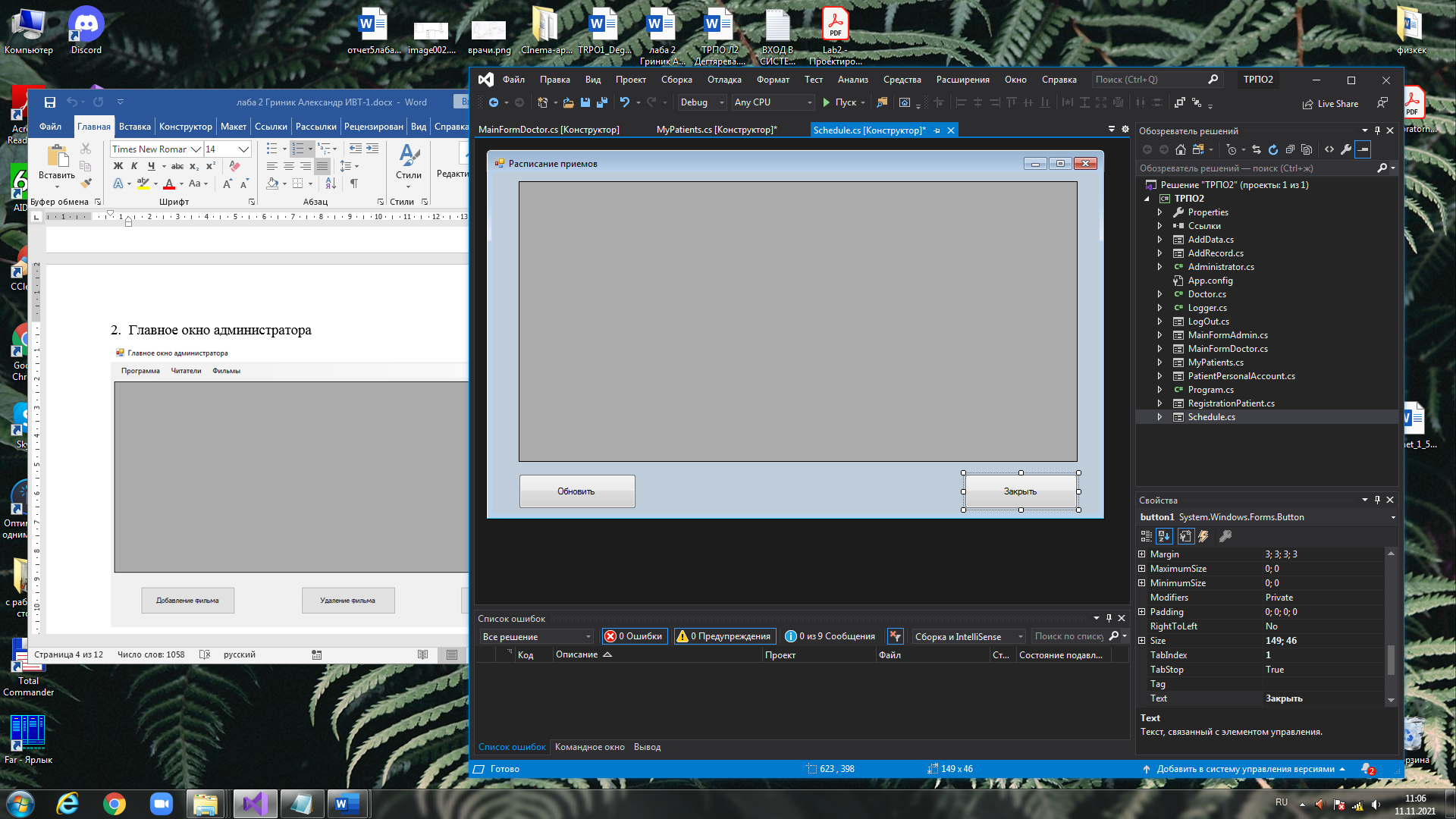


1) Содержит элемент DataGridView, отображающий пациентов, проходящих лечение у данного врача, перенесенные из личного кабинета пациента/регистратуры

2) Кнопка «Обновить» - обновляет данные

3) Кнопка «Закрыть» - делает выход из формы.

8. Форма «Расписание приемов»

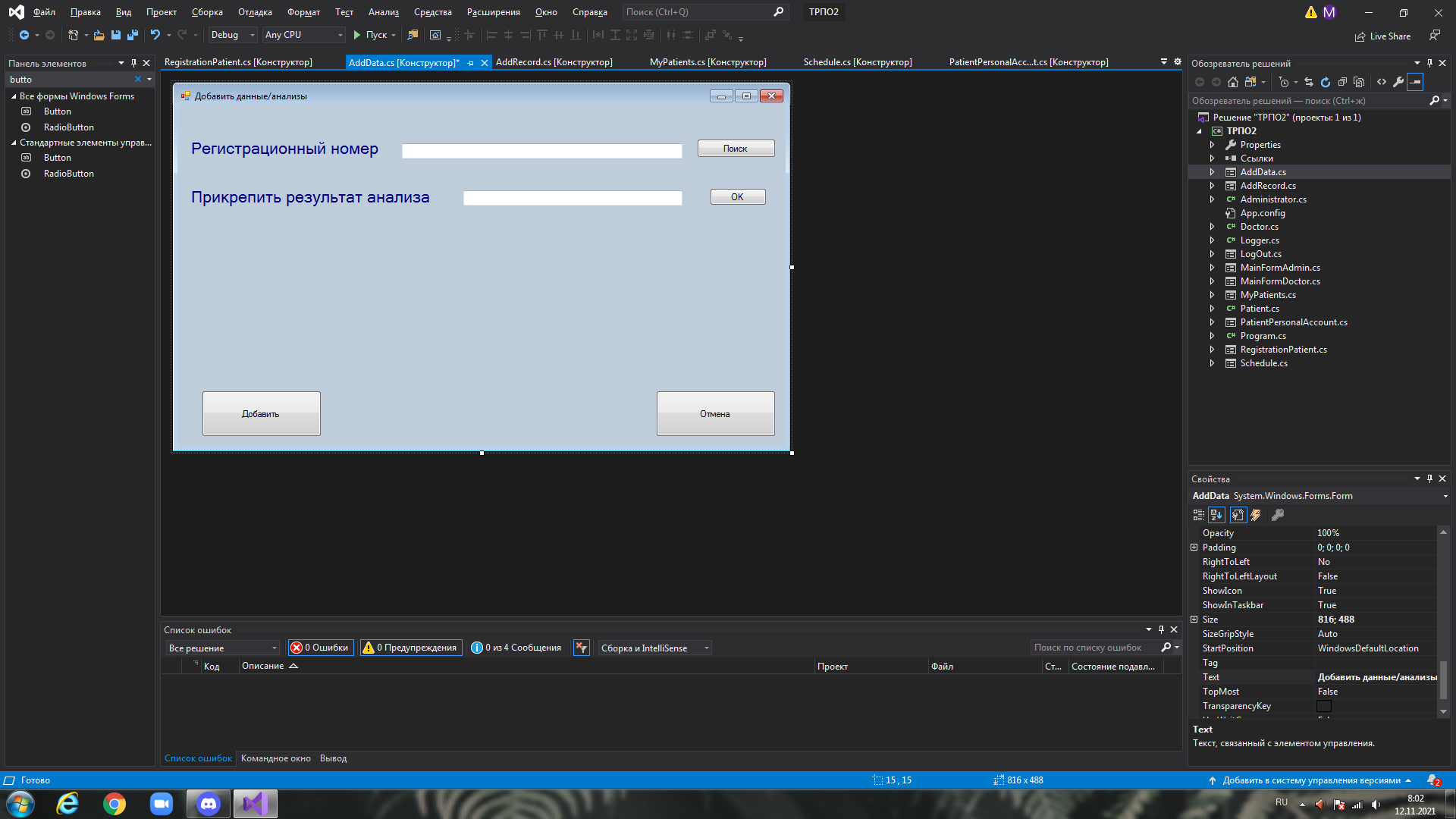


1) Содержит элемент DataGridView, отображающий записи пациентов, осуществленные предварительно из личного кабинета пациента/регистратуры

2) Кнопка «Обновить» - обновляет данные

3) Кнопка «Закрыть» - делает выход из формы.

9. Форма добавить данные/анализы



1. Форма содержит поиск пациента по регистрационному номеру

2. Также есть TextBox для прикрепления результата анализа, для врача и дальнейшего планирования лечения.

# 3.Описание классов

## 3.1. Логика приложения

Таблица 3.1 – Классы и файлы

|  |  |
| --- | --- |
| Название файла | Описание класса |
| Administator.cs | Класс, описывающий администратора(=регистратуру) больницы |
| Logger.cs | Класс для логирования действий пользователей |
| Program.cs | Класс для работы приложения |
| Doctor.cs | Класс, описывающий пользователей (докторов) ИС больницы |
| Patient.cs | Класс, описывающий пользователей(пациентов) ИС больницы |

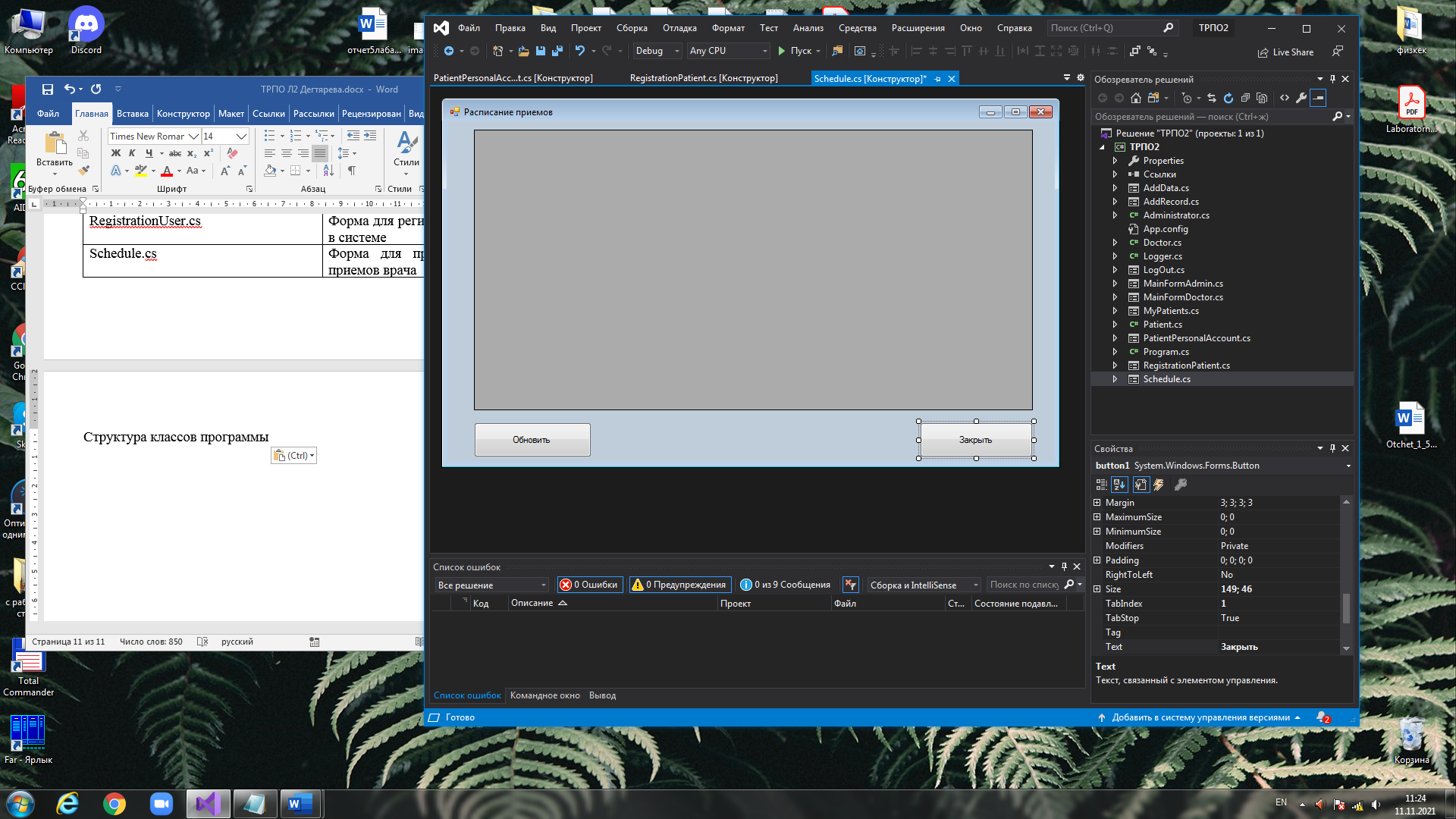
## 3.2. Классы графического интерфейса

Так как в языке программирования С# все окна и графические элементы форм являются отдельными классами, в таблице 4.2 представлен класс окон и пользовательских элементов управления.

Таблица 3.2 – Классы отвечающие за графический интерфейс

|  |  |
| --- | --- |
| AddData.cs | Форма для добавления данных пациента о своем лечении |
| AddRecord.cs | Форма для добавления записи к врачу |
| LogOut.cs | Формы для осуществления входа в систему администратором/пациентом/врачом |
| MainFormAdmin.cs | Главная форма для администратора, откуда он проводит дальнейшие операции |
| MainFormDoctor.cs | Главная форма для доктора, откуда он проводит дальнейшие операции |
| MyPatients.cs | Форма для просмотра пациентов, проходящих лечение у конкретного врача |
| PatientPersonalAccount.cs | Личный кабинет пациента |
| RegistrationUser.cs | Форма для регистрации пользователя в системе |
| Schedule.cs | Форма для просмотра расписания приемов врача |

Структура классов программы



Общая схема связности классов

