**Министерство образования, науки и молодежной политики Нижегородской области**

**Государственное бюджетное профессиональное образовательное учреждение**

**«Нижегородский Губернский колледж»**

Методическая комиссия «Информатика и вычислительная техника»

Допущен к защите:

преподаватель

\_\_\_\_\_\_\_\_\_\_\_\_В.А. Авакян,

\_\_\_\_\_\_\_\_\_\_\_\_Ю.С. Мамшева,

«21» марта 2022 г.

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

ПМ.01 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ

Студент: Кузнецова О.Н.

Специальность, группа: 09.02.07, 43П

Нижний Новгород

2023 г.

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc130157808)

[**1 «РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ»** 5](#_Toc130157809)

[**1.1 Проектирование системы и работа с базой данных** 5](#_Toc130157810)

[**1.2 Разработка приложения** 9](#_Toc130157811)

[**1.3 Тестирование** 19](#_Toc130157812)

[**2 «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»** 21](#_Toc130157813)

[**2.1 Анализ аспектов среды разработки и выбор платформы** 21](#_Toc130157814)

[**2.2 Проектирование и разработка мобильного приложения** 24](#_Toc130157815)

[**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ** 29](#_Toc130157816)

# **ВВЕДЕНИЕ**

Программа профессионального модуля является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности 09.02.07 «Информационные системы и программирования».

В части освоения основного вида деятельности «Разработка модулей программного обеспечения для компьютерных систем» и соответствующих профессиональных компетенций (ПК):

ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;

ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием;

ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств;

ПК 1.4. Выполнять тестирование программных модулей;

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода;

ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

Цели и задачи профессионального модуля – требования к результатам освоения профессионального модуля. С целью овладения указанным видом профессиональной деятельности и соответствующим профессиональными компетенциями обучающийся в ходе освоения профессионального модуля должен:

Иметь практический опыт:

* В разработке кода программного продукта на основе готовой спецификации на уровне модуля;
* В использовании инструментальных средств на этапе отладки программного продукта;
* В проведении тестирования программного модуля по определенному сценарию;
* В использовании инструментальных средств на этапе отладки программного продукта;
* В разработке мобильных приложений.

Уметь:

* Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
* Создавать программу по разработанному алгоритму как отдельный модуль;
* Выполнять отладку и тестирование программы на уровне модуля;
* Осуществлять разработку кода программного модуля на современных языках программирования;
* Уметь выполнять оптимизацию и рефакторинг программного кода;
* Оформлять документацию на программные средства.

Знать:

* Основные этапы разработки программного обеспечения;
* Основные принципы технологии структурного и объектно-ориентированного программирования;
* Способы оптимизации и приемы рефакторинга;

Основные принципы отладки и тестирования программных продуктов.

# **1 «РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ»**

## **1.1 Проектирование системы и работа с базой данных**

Предметная область – лабораторные анализы и их результаты. Нужно предусмотреть функционал для разных пользователей, а именно для администратора и лаборанта. Администратор имеет доступ к просмотру и редактированию всех данных. У лаборанта есть доступ только к ограниченному набору данных, а возможность редактирования присутствует только в таблице результатов. На начальном этапе были предусмотрены данные, благодаря которым можно было воссоздать структуру. В итоге в системе получилось 4 основных блока (таблицы): сотрудники (все данные о них, в том числе данные для авторизации), пациенты (данные, которые использовались при записи результатов, а также для авторизации в мобильном приложении), услуги (список анализов и их цены, которые также нужны для описания результатов) и сами результаты, которые объединяют данные с трёх таблиц (кто сдавал анализ, кто его проводил, когда и какой результат), также присутствует таблица ролей для идентификации входящего клиента.

Для того, чтобы импортировать данные в базу данных необходимо их привести к третьей нормальной форме. Для этого при помощи функций Excel данные приводятся к одному виду, повторяющиеся данные выносятся в отдельные таблички, а также исправляются синтаксические ошибки. По окончанию редактирования два документа с пользователями и сервисами выглядят следующим образом.

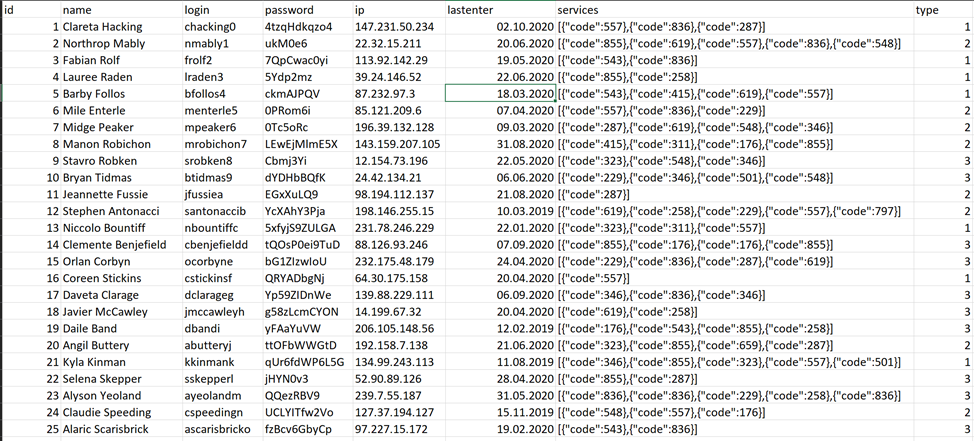
[](https://user-images.githubusercontent.com/100830195/222380718-5447104b-d74c-4d28-b385-acddedfbaf36.png)

Рисунок 1.1.1 – Таблица пользователей

[](https://user-images.githubusercontent.com/100830195/222381297-d273c37d-04cf-4086-b3a1-cb18d5c6554f.png)

Рисунок 1.1.2 – Таблица сервисов

После приведения данных к правильной форме, была создана база данных и занесены все данные. Были созданы следующие таблицы.

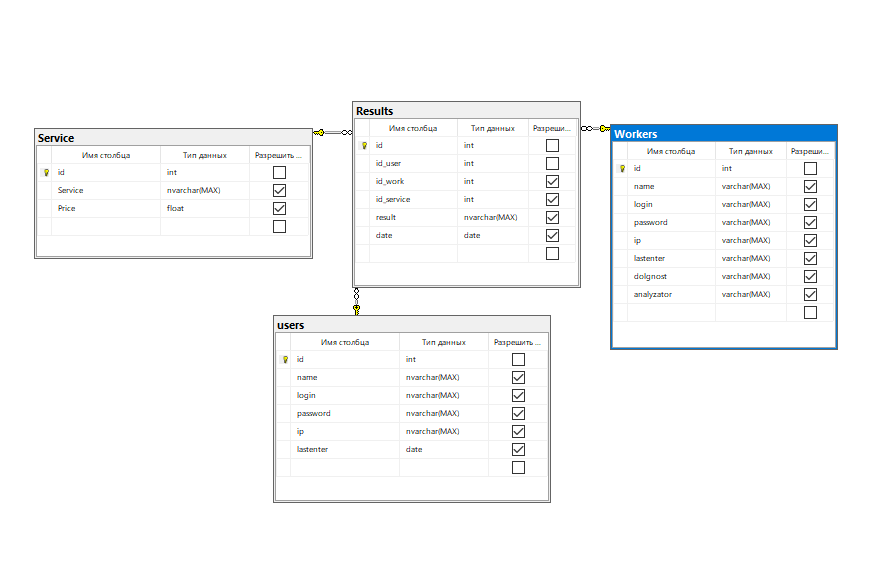
[](https://user-images.githubusercontent.com/100830195/222416719-50fdd53c-b680-49c0-8502-561f2accd175.png)

Рисунок 1.1.3 – Диаграмма базы данных

После заполнения таблиц данными, они стали выглядеть следующим образом

[](https://user-images.githubusercontent.com/100830195/222417110-59a3b9d1-c156-4a99-9664-0bd5ad2cef6a.png)

Рисунок 1.1.4 – Данные таблицы услуг

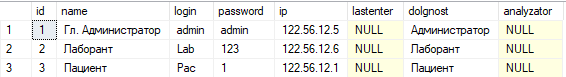
[](https://user-images.githubusercontent.com/100830195/222417139-e4cb2692-4bed-408c-8dbd-60ade0bc48e4.png)

Рисунок 1.1.5 – Данные таблицы работников

[](https://user-images.githubusercontent.com/100830195/222417162-f3d5124c-0f8f-4ef2-b2fd-70813df6e4b3.png)

Рисунок 1.1.6 – Данные таблицы пользователей

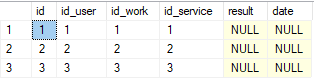
[](https://user-images.githubusercontent.com/100830195/222417181-5b50fed7-3310-4cd1-9d31-a41cc8cdfc17.png)

Рисунок 1.1.7 – Данные таблицы результатов анализов

## **1.2 Разработка приложения**

Первым делом было разработано окно регистрации и окно входа:

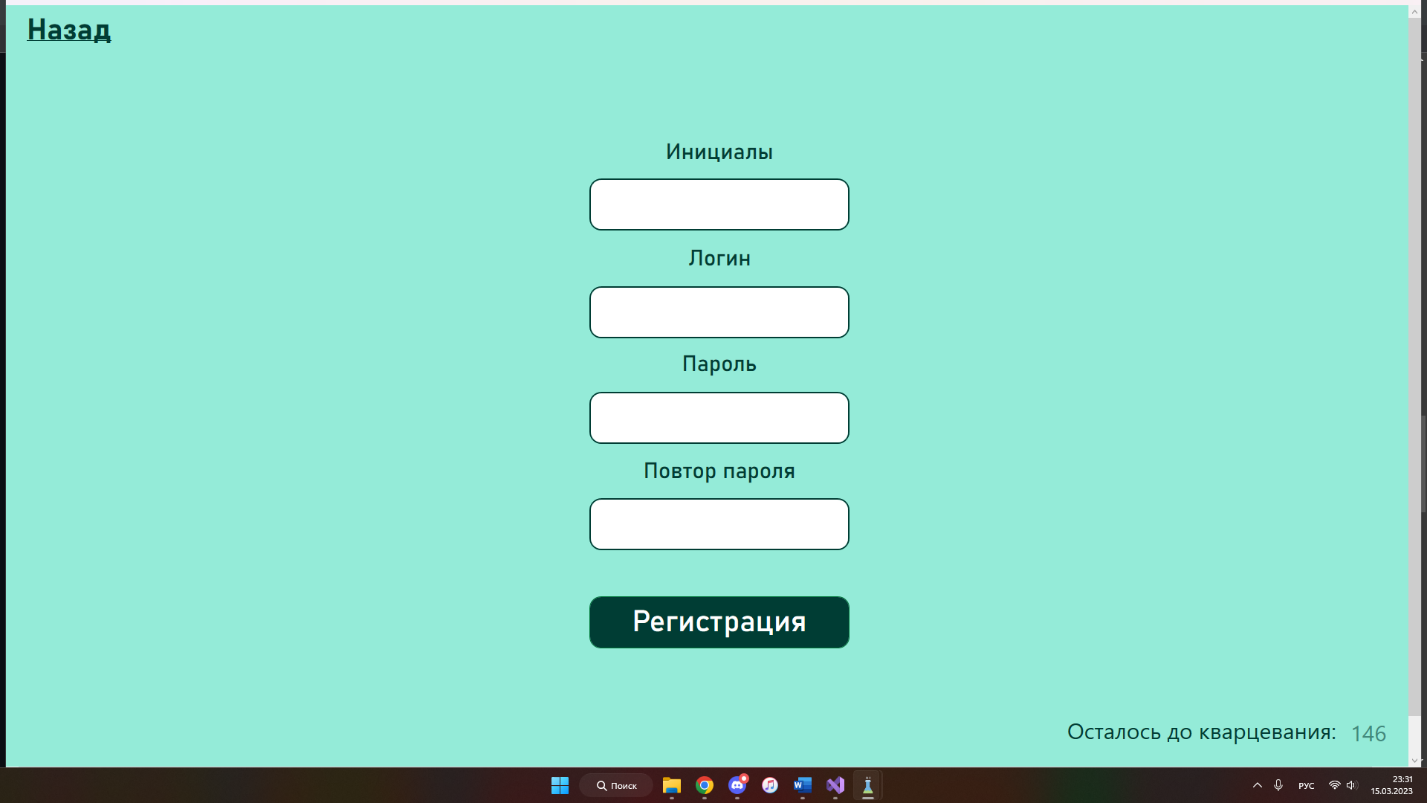
[](https://user-images.githubusercontent.com/100990884/225435041-ef9116a0-8d24-48b4-85c0-5d30744c6c33.png)

Рисунок 1.2.1 – Форма регистрации

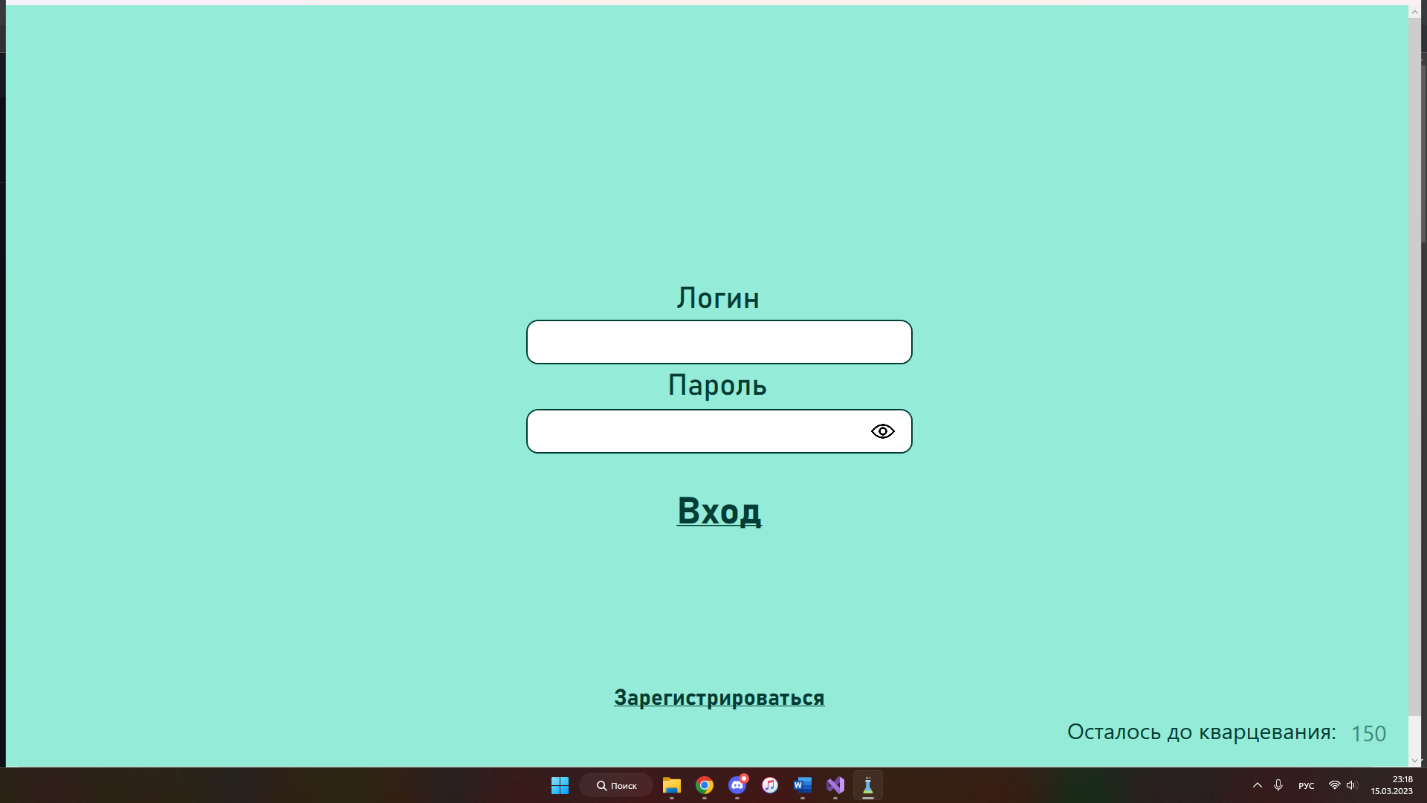
[](https://user-images.githubusercontent.com/100990884/225432482-5d3c98f8-5d98-4cf5-b793-ab9e5ef8ff5e.png)

Рисунок 1.2.2 – Форма авторизации

В строке пароля добавлена кнопка настройки видимости текста для удобства пользователей:

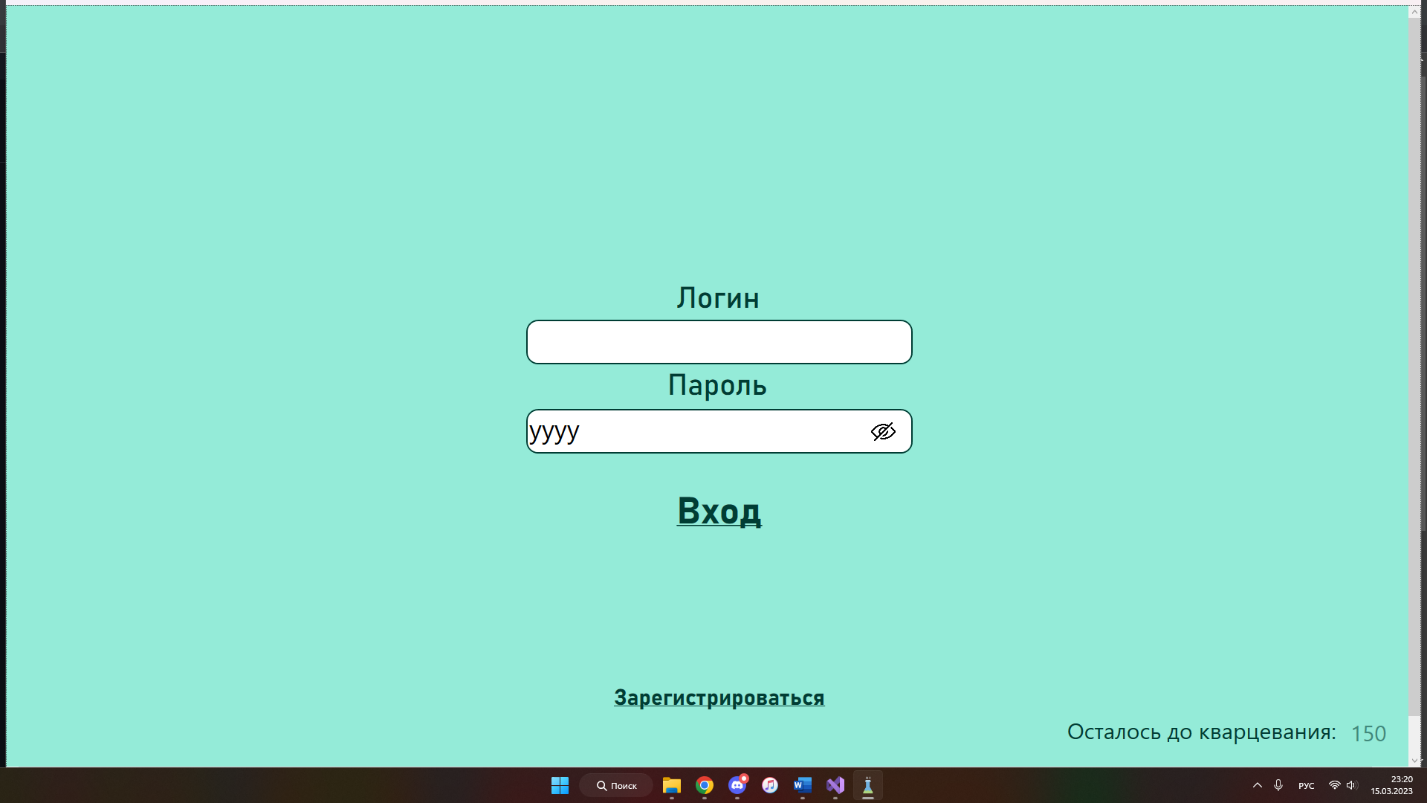
[](https://user-images.githubusercontent.com/100990884/225432817-56b4da99-3221-43dd-8044-a816a0e806b5.png)

Рисунок 1.2.3 – Форма авторизации с показывающимся паролем

При неверном вводе пароля, пользователю выводится сообщение о количестве попыток повторного ввода:

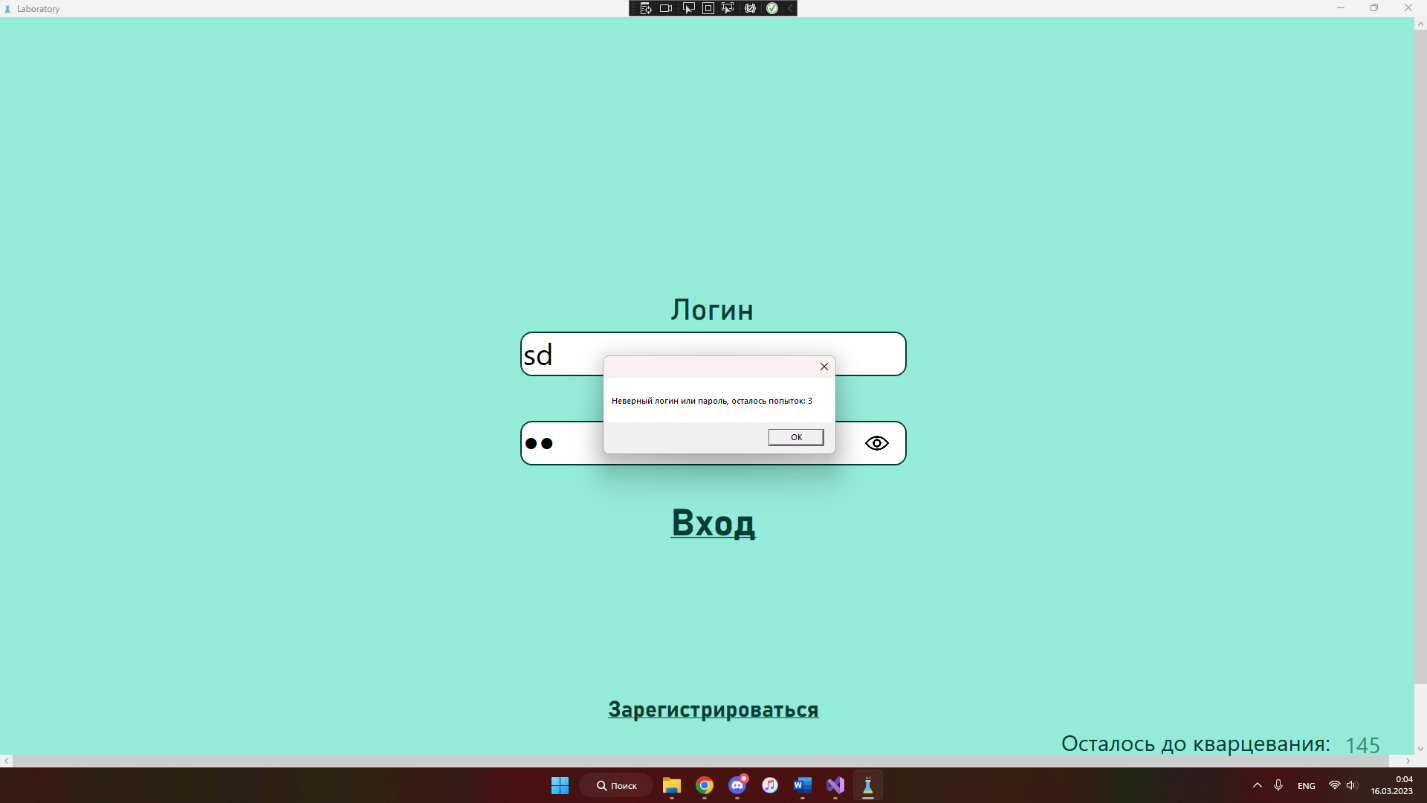
[](https://user-images.githubusercontent.com/100990884/225442140-db74cab9-86e7-41c7-9c14-cbbbdf988d65.png)

Рисунок 1.2.4 – Сообщение о неправильно введенном пароле

После окончания попыток, пользователя перекидывает на другую страницу где требуется ввести капчу для повторной попытки входа:

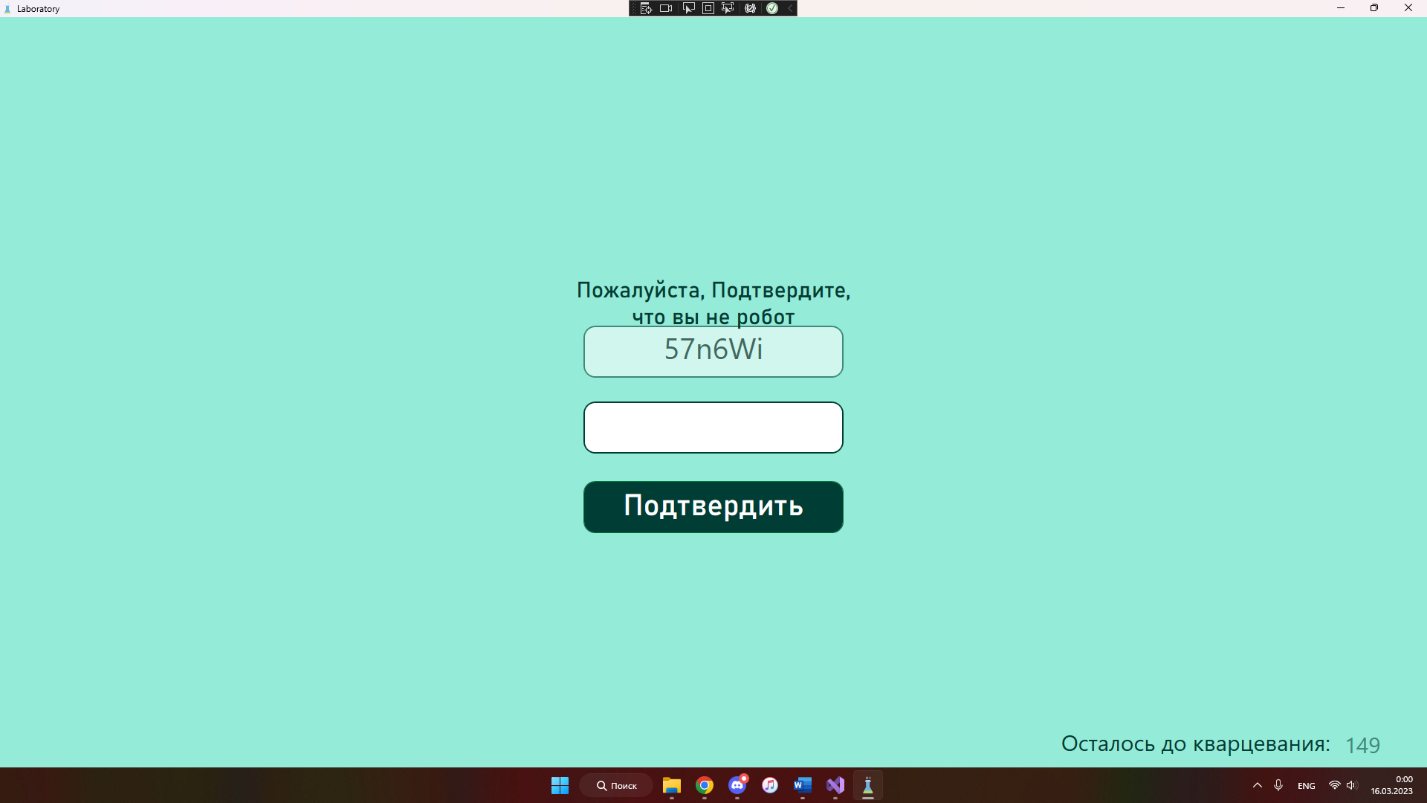
[](https://user-images.githubusercontent.com/100990884/225442046-1648e4f3-d495-447b-8b9e-2acfebb96640.png)

Рисунок 1.2.5 – Форма окна капчи

Когда таймер заканчивает отсчёт пользователю будет предложени либо прекратить действие программы, либо остаться на 5 минут:

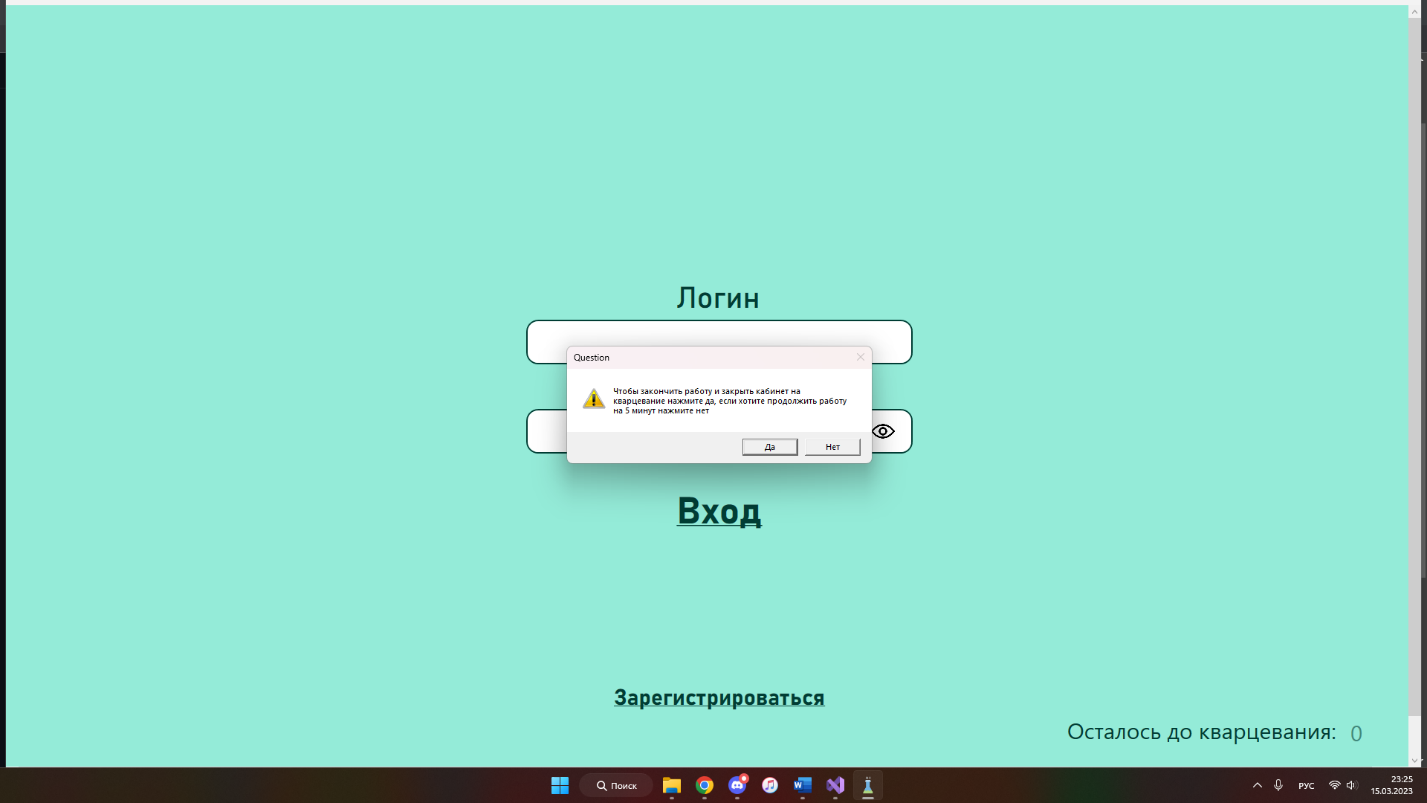
[](https://user-images.githubusercontent.com/100990884/225433992-13042279-744f-452a-b639-f9962d05ebe6.png)

Рисунок 1.2.6 – Окно окончания сеанса по истечению таймера

На главной странице так же присутствует таймер и выбор конкретной категории информации.

Для админа существует доступ к истории входа пользователей:

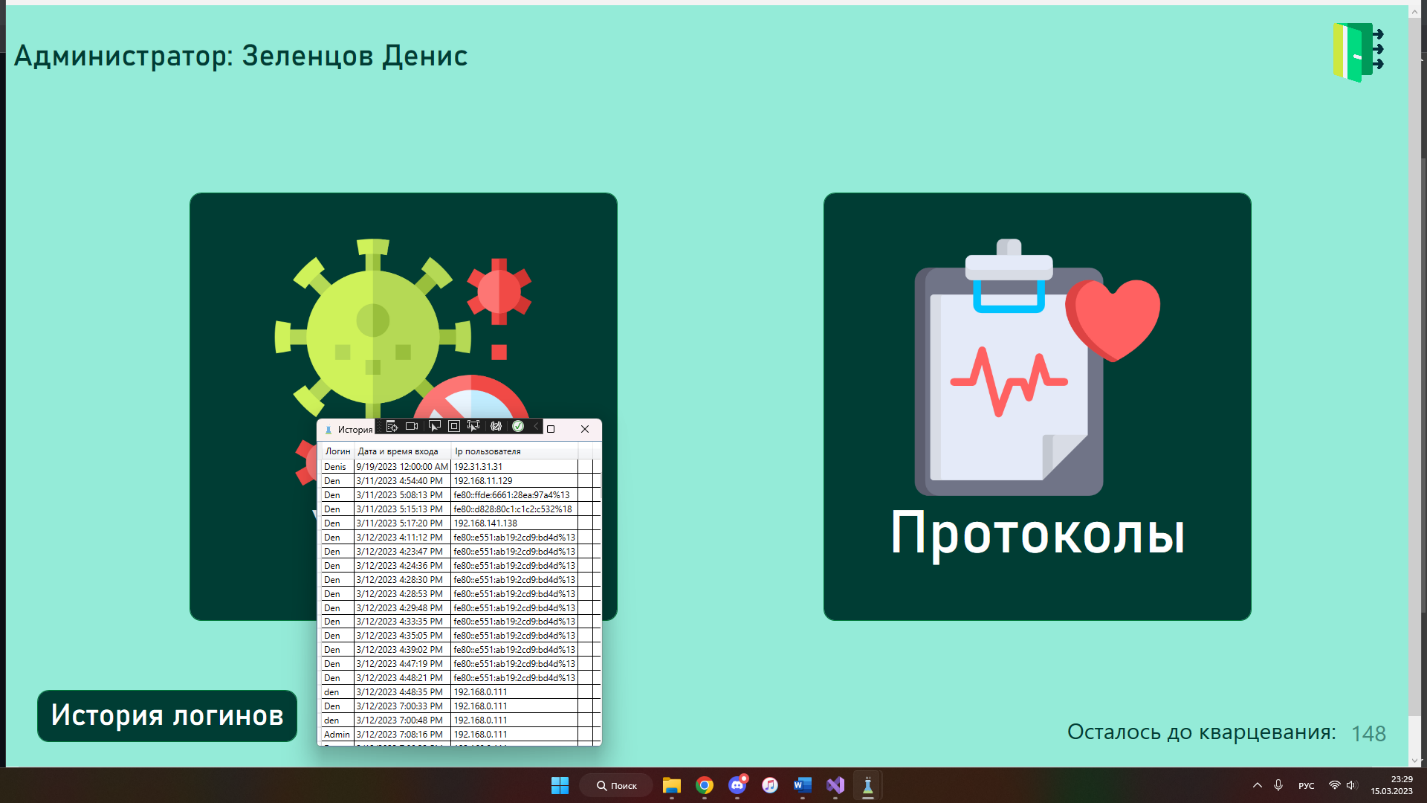
[](https://user-images.githubusercontent.com/100990884/225434679-4806d73b-a2a4-4114-b601-43cd7ac87b64.png)

Рисунок 1.2.7 – Демонстрация истории входов

Организован вывод данных для таблиц результатов анализов, а также услуг, предоставляемых лабораторией

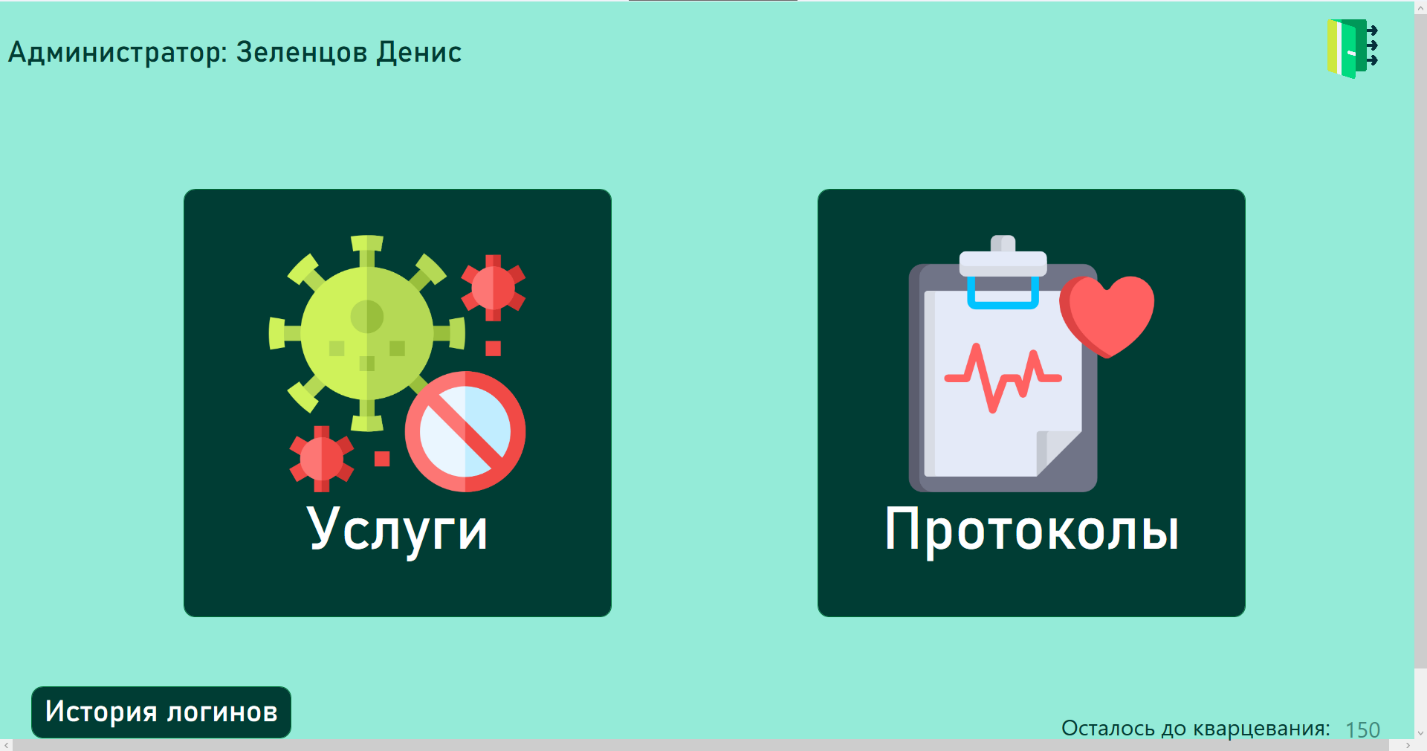
[](https://user-images.githubusercontent.com/100830195/225432600-036878bf-f595-4b3b-940a-d17e69fa70be.png)

Рисунок 1.2.8 – Главное меню программы

Вывод услуг:

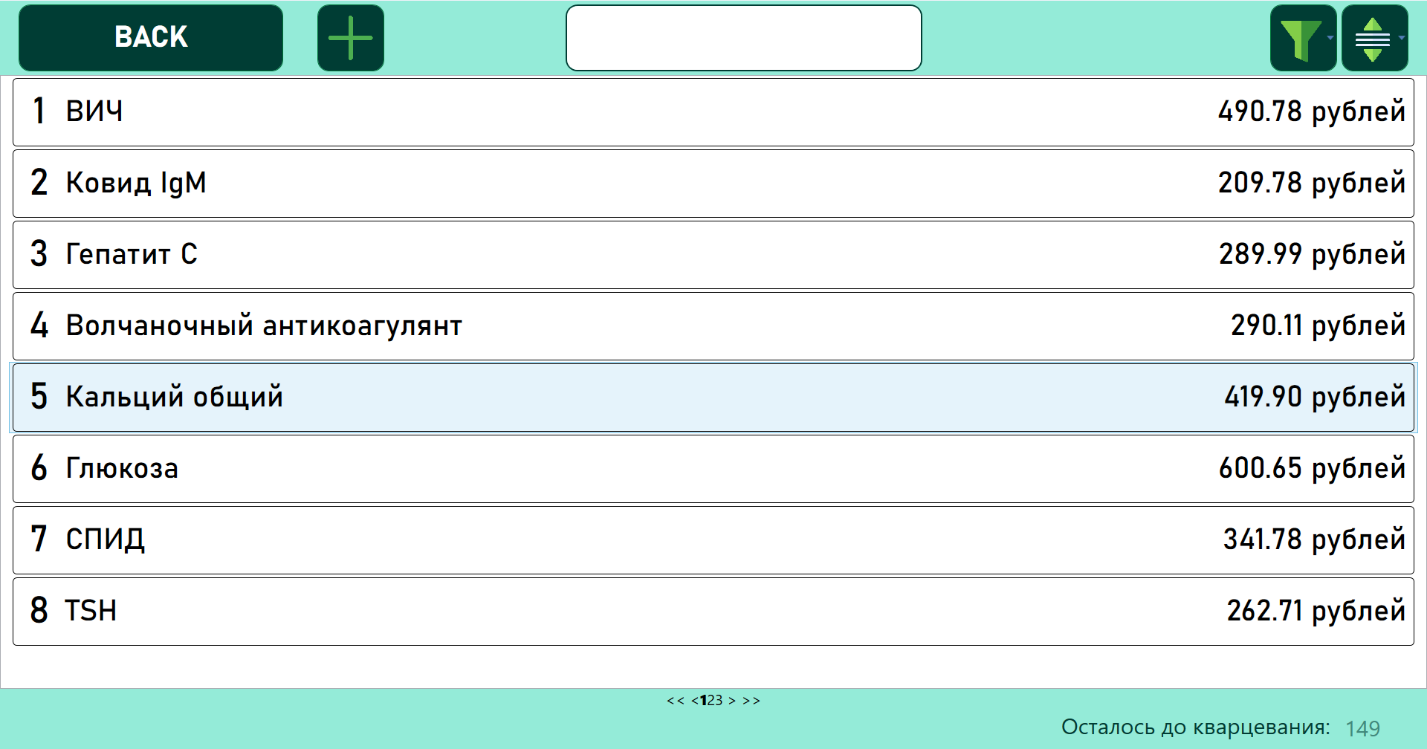
[](https://user-images.githubusercontent.com/100830195/225432701-f2b323dc-93d3-45cf-8c62-2dc36e04c255.png)

Рисунок 1.2.9 – Форма вывода сервисов

Вывод результатов:

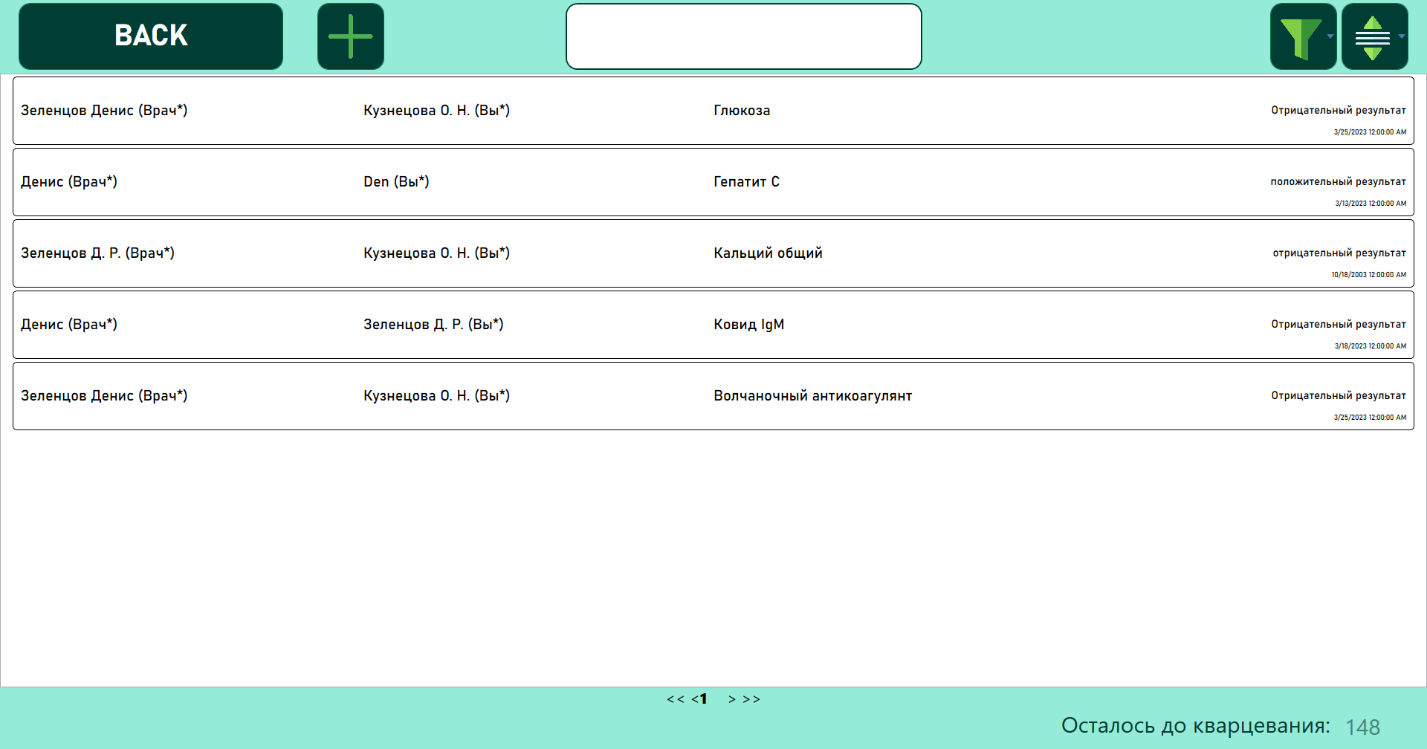
[](https://user-images.githubusercontent.com/100830195/225433013-811ac6fe-8c9b-4cd5-908a-9730c007a55c.png)

Рисунок 1.2.10 – Форма вывода результатов анализов

Разработаны формы добавления редактирования и удаления данных

Добавление услуг:

[](https://user-images.githubusercontent.com/100830195/225433280-33769a1c-b5a8-4e7d-932a-c8ce07a2fec8.png)

Рисунок 1.2.11 – Форма добавления услуг

Редактирование и удаление услуг:

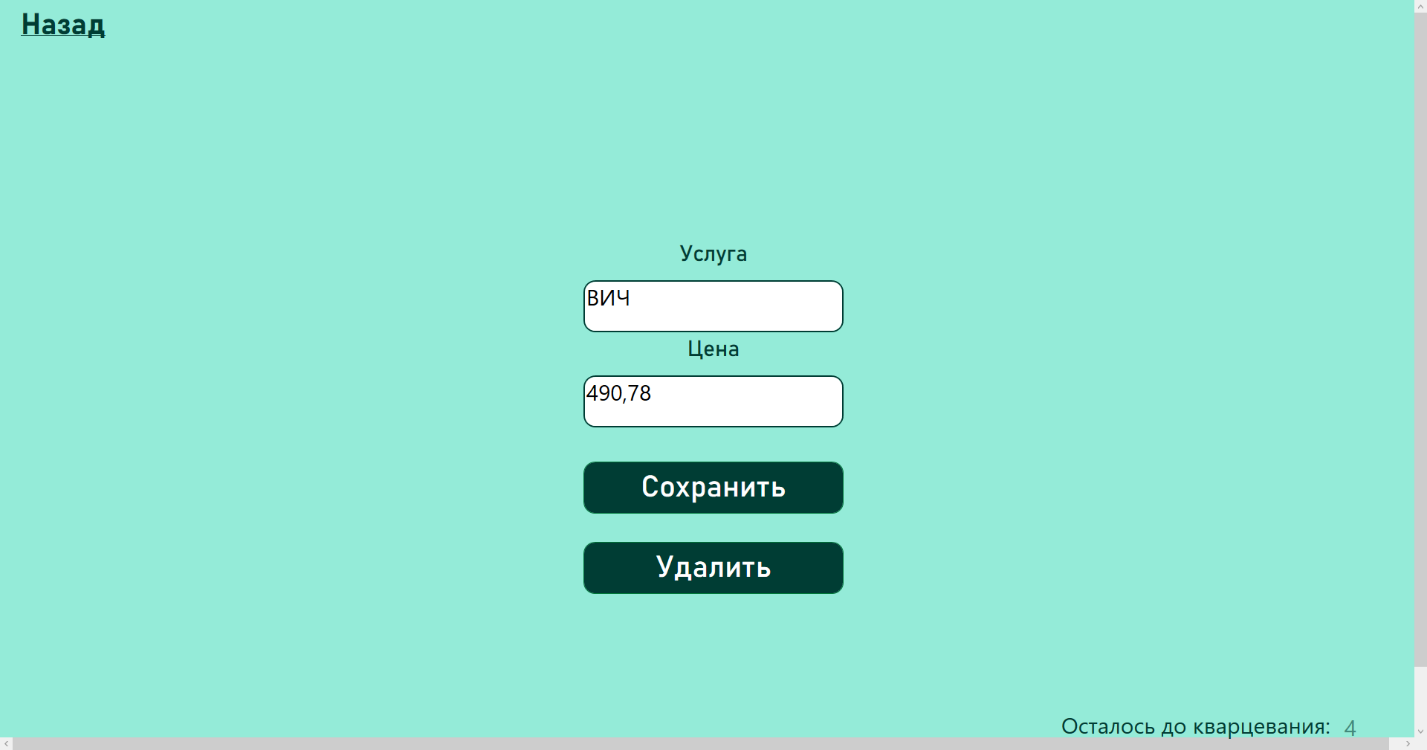
[](https://user-images.githubusercontent.com/100830195/225436637-76381fac-6277-49a9-a841-5b3102baead9.png)

Рисунок 1.2.12 – Форма редактирования сервисов

Добавление результатов (выпадающие списки берут поля из соответствующих таблиц):

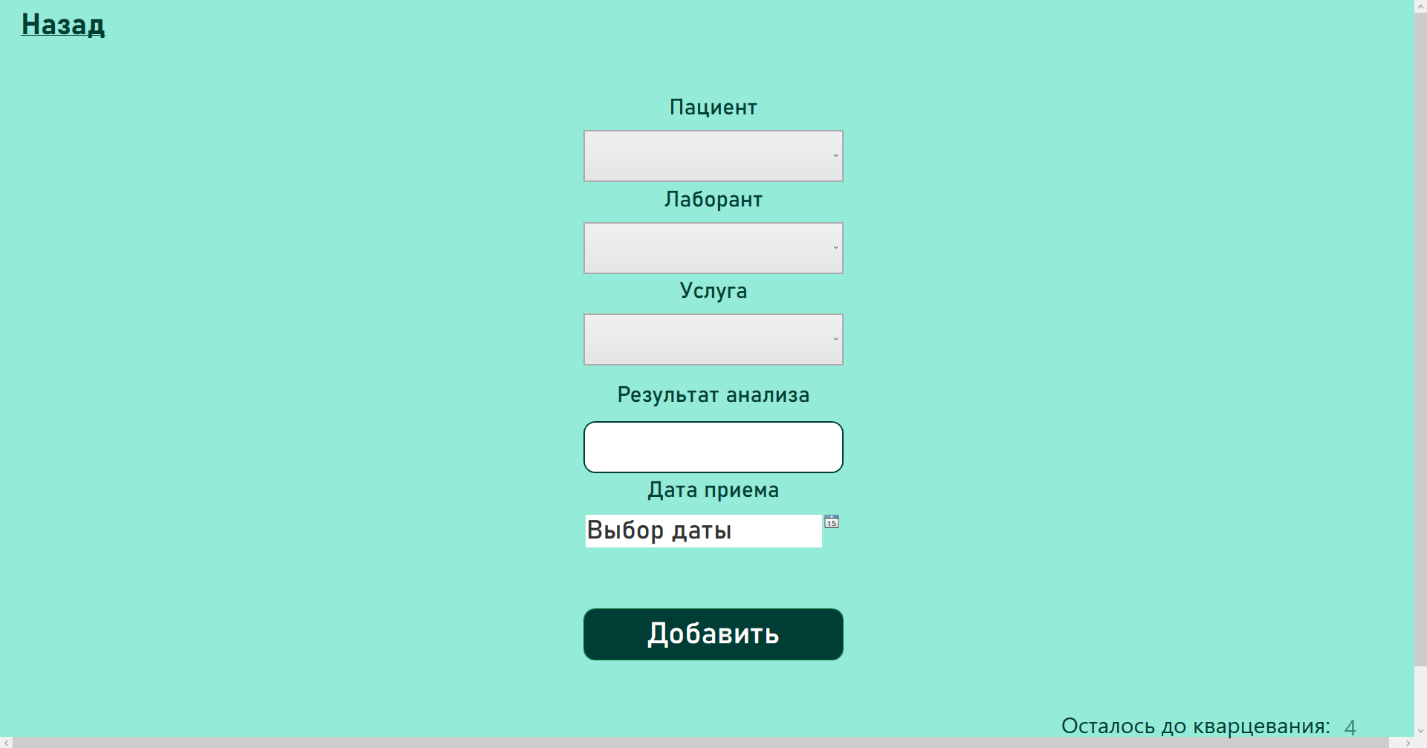
[](https://user-images.githubusercontent.com/100830195/225436892-2ca36da2-61ed-4574-a463-86a5221f380a.png)

Рисунок 1.2.13 – Форма добавления результатов

Редактирование и удаление результатов:

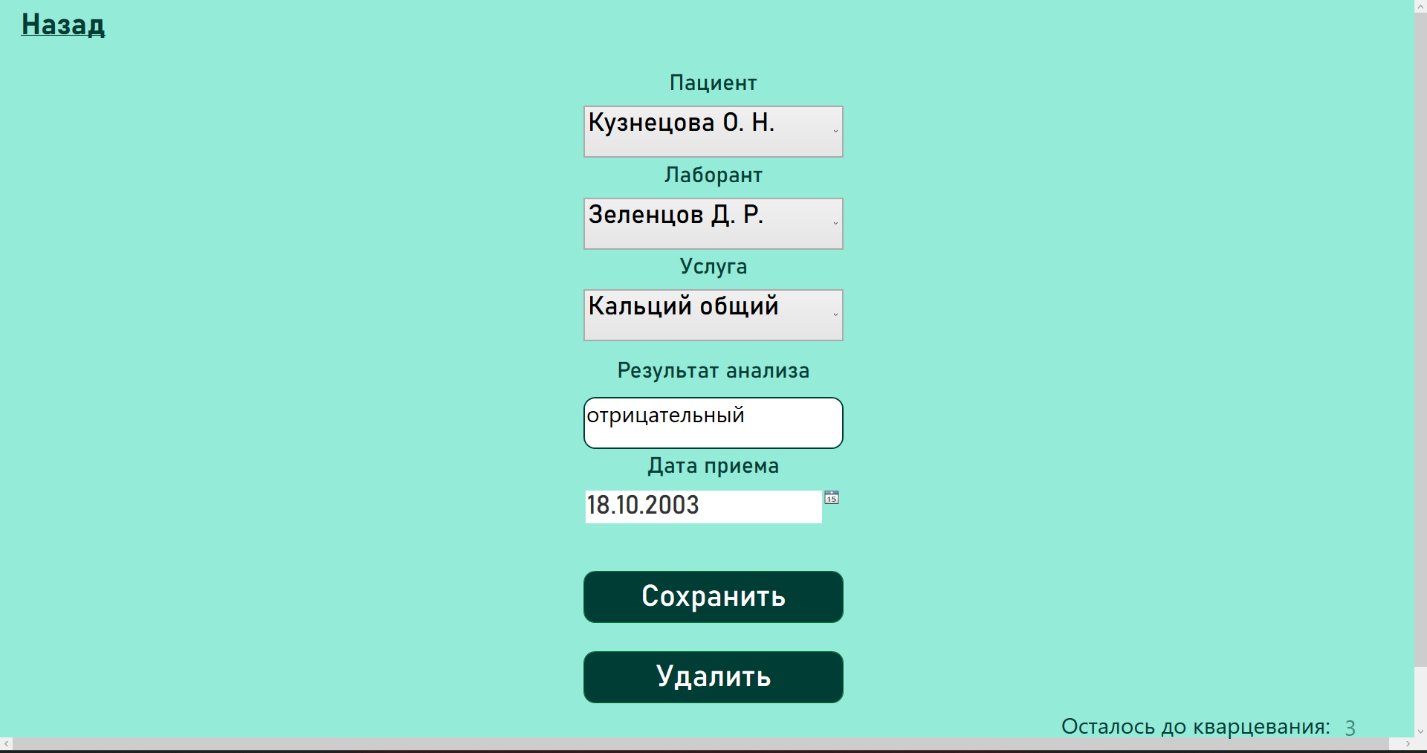
[](https://user-images.githubusercontent.com/100830195/225437115-b667a5c1-0f4d-4f91-9943-5c010d3ae16d.png)

Рисунок 1.2.14 – Форма окна капчи

Поиск по услугам:

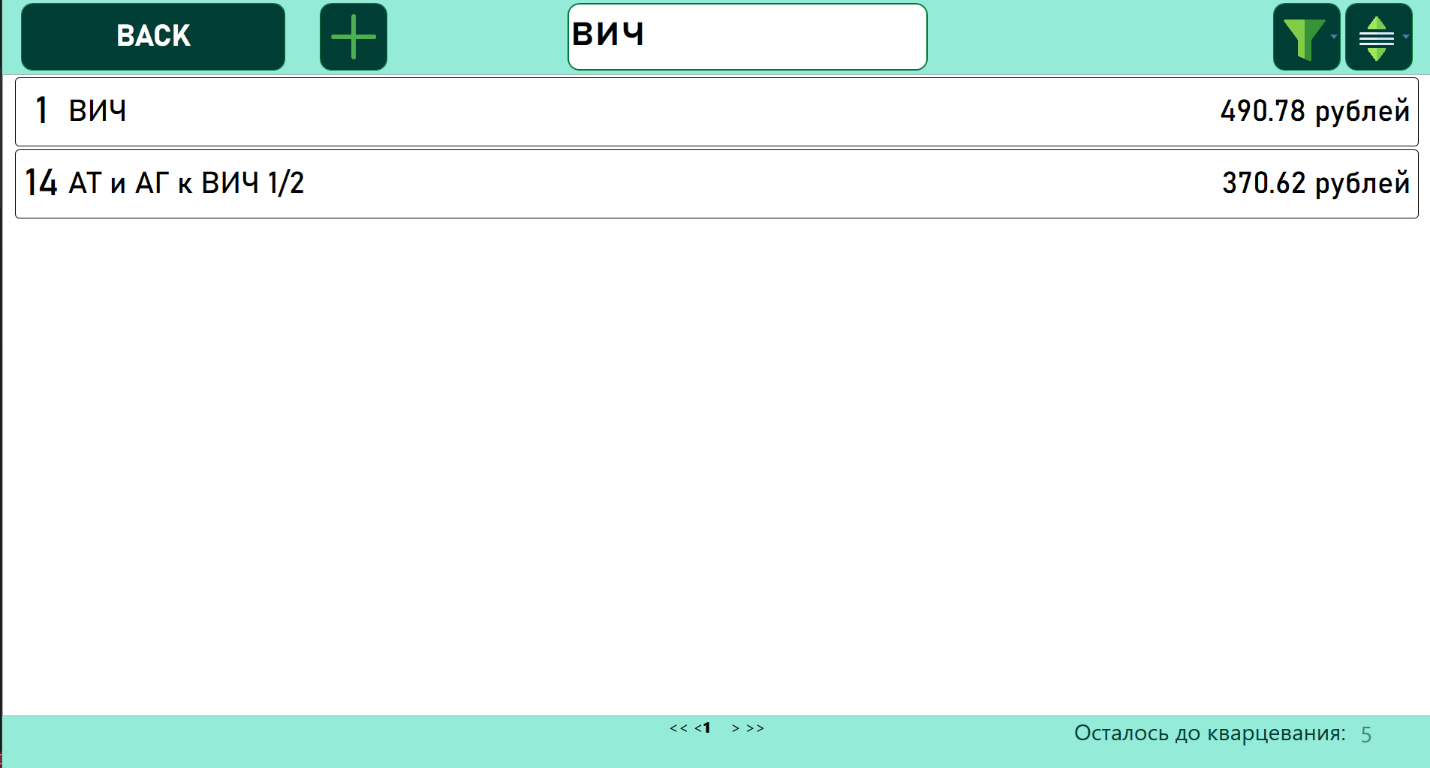
[](https://user-images.githubusercontent.com/100830195/225438720-38979df7-7b0a-4633-922c-85c3e82a6735.png)

Рисунок 1.2.15 – Демонстрация поиска по услугам

Поиск по анализам (можно выбрать по какому столбцу искать при помощи выпадающего списка)

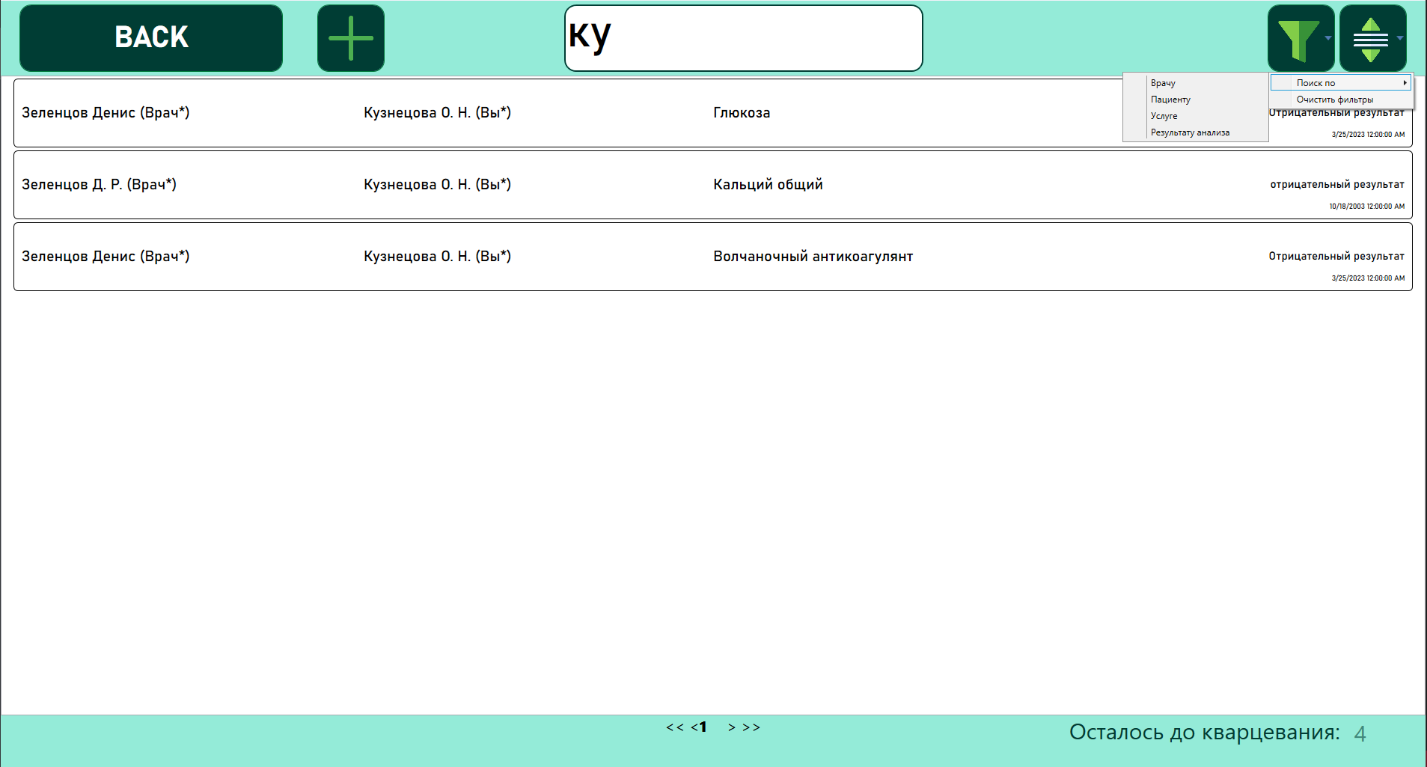
[](https://user-images.githubusercontent.com/100830195/225438985-962cc726-b403-4ef3-a74c-e83e2fa718e4.png)

Рисунок 1.2.16 – Форма окна капчи

В окне услуг присутствует фильтрация по цене, а также сортирока по алфавиту(а-я, я-а) и по цене(по возрастанию и убыванию)

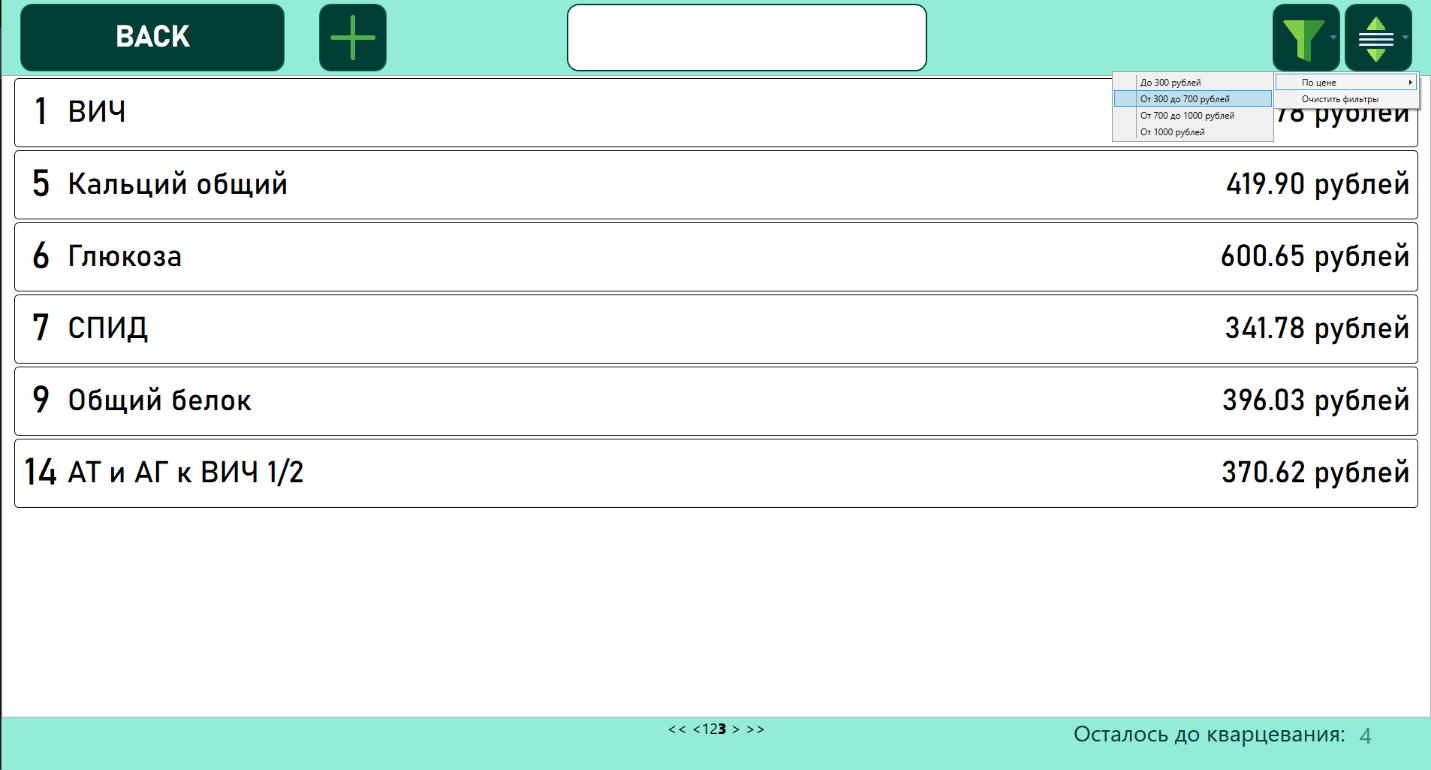
[](https://user-images.githubusercontent.com/100830195/225439609-3a72c701-e92d-4e55-a090-d54c4df37d11.png)

Рисунок 1.2.17 – Демонстрация фильтров

В окне анализов присутствует сортировка по услугам (а-я, я-а)

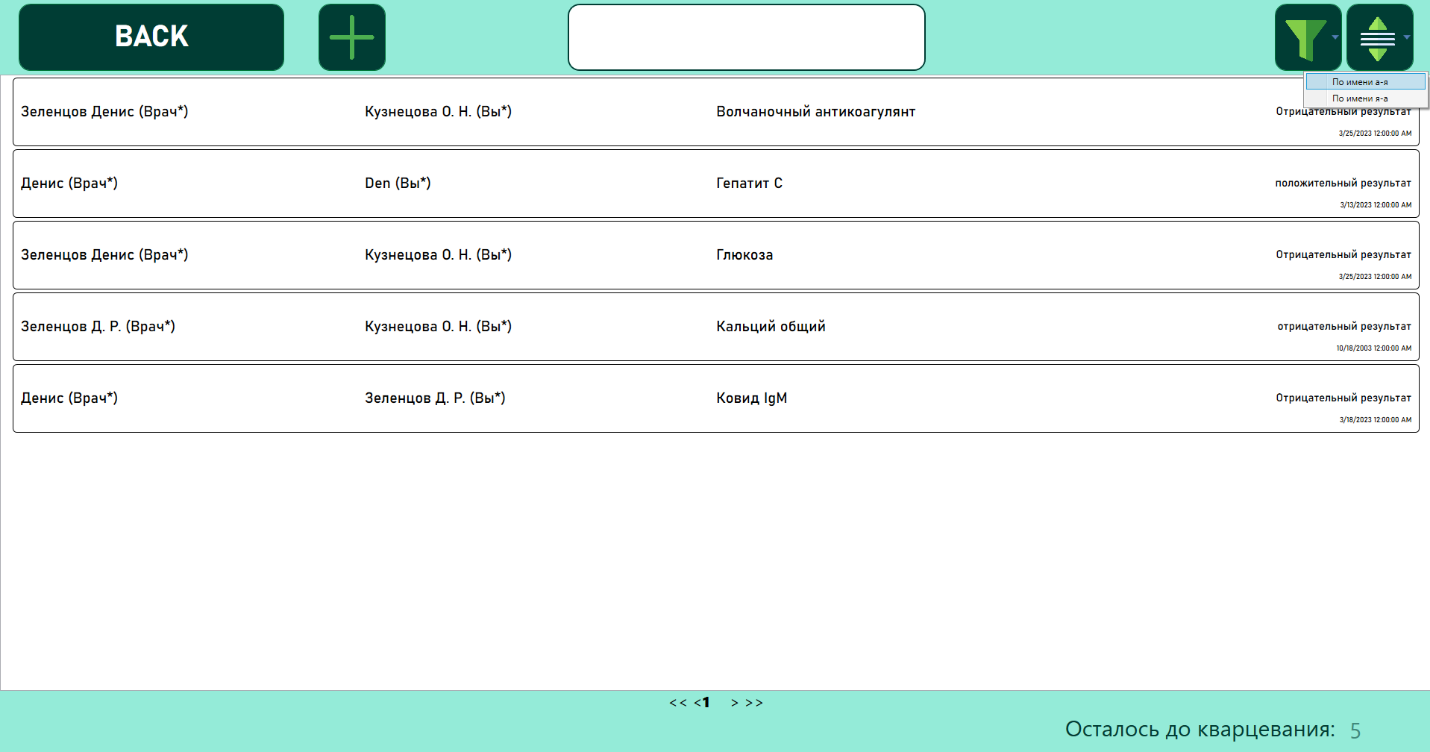
[](https://user-images.githubusercontent.com/100830195/225439805-73afedff-ca84-40cd-9990-579d86b483b1.png)

Рисунок 1.2.18 – Демонстрация сортировки

Для очистки фильтров присутствует отдельная кнопка

Также была разработана форма печати, доступная только работникам лаборатории.

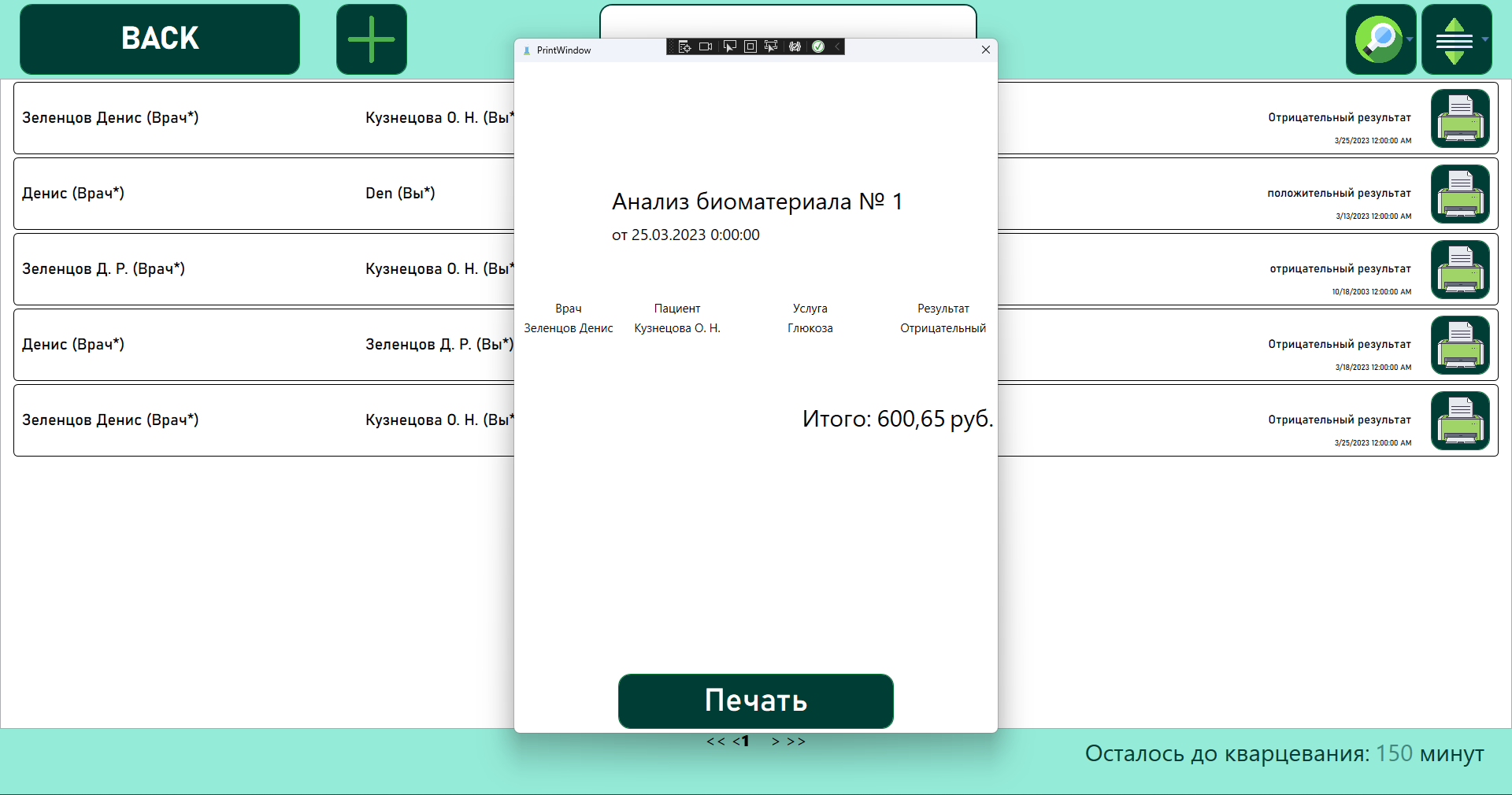


Рисунок 1.2.19 – Форма печати результатов анализа

После нажатия на кнопку печати, открывается диалоговое окно Windows для печати, и информацию можно получить прямо на лист а4.

## **1.3 Тестирование**

**Аннотация теста**

|  |  |
| --- | --- |
| Название проекта | Проект Лаборатории «MedicalLaboratory» |
| Рабочая версия | 1.0 |
| Имя тестирующего | Кузнецова О.Н. |
| Дата теста | 19.03.2022 |

**Тестовый пример #1:**

|  |  |
| --- | --- |
| Тестовый пример # | TC\_UI\_1: Тестовый случай пользовательского интерфейса |
| Приоритет тестирования | Средний |
| Заголовок | Воспользуйтесь фильтрацией списка по стоимости два раза |
| Краткое изложение теста | Проверить работу фильтра при повторном выборе другого фильтра |
| Этапы теста | 1. Авторизироваться в системе. 2. Перейти на страницу «Услуги». 3. Воспользоваться фильтрацией два раза, при этом выбрав разные позиции. |
| Тестовые данные | - |
| Ожидаемый результат | Исправная работа фильтра |
| Фактический результат | Данные не выводятся. Фильтр не работает |
| Статус | Неудачный |
| Предварительное условие | - |
| Постусловие | Состояние системы после выполнения теста должно быть стабильным. |
| Примечания | Ошибка связана с тем, что лист, который фильтруется по одному параметру снова принимает условие уже другого фильтра. Соответственно, если в первом случае был выбран фильтр по стоимости до 300 рублей, то при повторной фильтрации от 300 до 700 рублей алгоритм просто ничего не найдет. |

После теста, используя отладку, было найдено место ошибки, а именно необходимо было исправить работу с листами

**Тестовый пример #2:**

|  |  |
| --- | --- |
| Тестовый пример # | TC\_UI\_2: Тестовый случай пользовательского интерфейса |
| Приоритет тестирования | Высокий |
| Заголовок | Подтвердите страницу авторизации с действительным именем пользователя и паролем. |
| Краткое изложение теста | Проверить работу формы авторизации, проверить исключения, вызванные неправильным логином или паролем. |
| Этапы теста | 1. Ввести логин. 2. Ввести пароль. 3. Нажать кнопку «Войти». |
| Тестовые данные | Из базы данных. |
| Ожидаемый результат | При правильных введённых данных – переход на форму пользователя. При ошибке ввода появляется сообщение, которое предупреждает пользователя. |
| Фактический результат | Совпадает с ожидаемым. |
| Статус | Удачный |
| Предварительное условие | - |
| Постусловие | Состояние системы после выполнения теста должно быть стабильным. А при трех неправильных попытках входа пользователь должен ввести капчу для повторной авторизации |
| Примечания | - |

# **2 «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»**

## **2.1 Анализ аспектов среды разработки и выбор платформы**

В ходе разработки программного обеспечения необходимо внимательно отнестись к выбору среды разработки (IDE – Integrated Development Environment). Основная причина выбора той или иной IDE зависит от языка программирования и целевой платформы. Поскольку для разработки android приложений чаще всего используется язык программирования Java, то ниже будут представлены наиболее популярные IDE предназначенные для разработки приложений на Java для платформы Android.

На сегодняшний день существует множество Android IDE, используемых разработчиками. Одними из наиболее популярных IDE для разработки приложений для платформы Android являются: Android Studio, Eclipse, NetBeans IDE, Xamarin.Android, AIDE, Komodo, IntelliJ IDEA.

Eclipse – свободная интегрированная среда разработки модульных кроссплатформенных приложений. Однако, чтобы начать программировать под Android, необходимо установить плагин ADT (Android Development Tools). Eclipse является платформо-независимым продуктом, за исключением библиотеки SWT.

Komodo IDE – это коммерческая среда разработки, построенная на Komodo Edit, а также имеющая ряд дополнительных функций, таких как отладка кода (включая удаленную), рефакторинг, работа с VCS, HTTP Inspector, Rx Toolkit, интегативный шелл, юнит-тесты, интеграцию с другими продуктами компании (Stackato, Perl Dev Kit и TCL Dev Kit) и т.д. Данная среда разработки обеспечивает подсветку синтаксиса и сворачивание блоков для разнообразных языков.

IntelliJ IDEA представляет собой высокотехнологичный комплекс тесно интегрированных инструментов программирования, включающий интеллектуальный редактор исходных текстов с развитыми средствами автоматизации, мощные инструменты рефакторинга кода, встроенную поддержку технологий J2EE, механизмы интеграции со средой тестирования Ant/JUnit и системами управления версиями, уникальный инструмент оптимизации и проверки кода Code Inspection, а также инновационный визуальный конструктор графических интерфейсов.

Xamarin – это платформа с открытым исходным кодом, предназначенная для построения современных производительных приложений для iOS, Android и Windows с .NET. Платформа Xamarin представляет собой уровень абстракции, который обеспечивает управление взаимодействием между общим кодом и кодом базовой платформы. Приложения Xamarin.Android компилируются из языка C# в промежуточный язык (IL), который при запуске приложения претерпевает Just-in-Time-компиляцию (JIT) в машинную сборку. Приложения Xamarin.Android работают в среде выполнения Mono параллельно с виртуальной машиной среды выполнения Android (ART). Xamarin предоставляет привязки .NET к пространствам имен Android.\* и Java.\*. Среда выполнения Mono обращается к этим пространствам имен с использованием управляемых вызываемых оболочек (MCW) и предоставляет среде выполнения ART вызываемые программы-оболочки Android (ACW), благодаря чему обе среды могут вызывать код друг друга.

NetBeans IDE – бесплатная интегрированная среда разработки с открытым исходным кодом для разработчиков программного обеспечения. Среда предоставляет все средства, необходимые для создания профессиональных десктоп приложений, корпоративных, мобильных и веб-приложений на платформе Java, а также C/C++, PHP, JavaScript, Groovy и Ruby.

Android Studio – интегрированная среда разработки, основанная на программном обеспечении IntelliJ IDEA от компании JetBrains, предназначенная для работы с платформой Android. Android Studio поддерживает различные виды сборок и генерацию нескольких .apk файлов, а также разработку приложений для Android Wear и Android TV. Кроме того, статический анализатор кода (Lint) данной среды разработки позволяет находить проблемы производительности, а также несовместимости версий. Android Studio доступна для Windows, OS X и Linux.

Проведя анализ некоторых наиболее используемых IDE, предназначенных для работы с платформой Android для разработки приложений рекомендуется использовать среду разработки Android Studio. На текущий момент данная IDE является наиболее используемой при разработке мобильных приложений, кроме того, Android Studio считается официальным средством разработки Android приложений.

## **2.2 Проектирование и разработка мобильного приложения**

Предметная область мобильного приложения заключается в отслеживании своего внутреннего состояния и другой информации. Пользователь должен авторизовываться, и видеть информацию своего аккаунта.

Был разработан загрузочный экран, чтобы пользователя встречала красивая картинка с хвойными лесами и логотипом программы. После чего уже высвечивается приветственное окно.



Рисунок 2.1.1 – Загрузочное окно программы

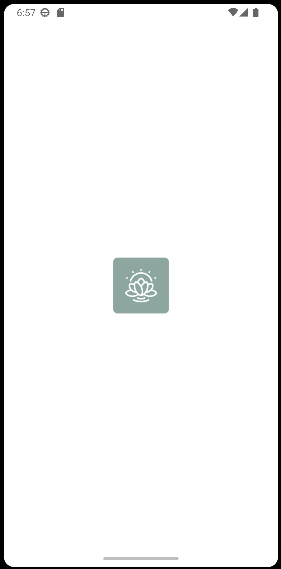


Рисунок 2.1.2 – Загрузочное окно программы

При загрузке программы, пользователя встречает приветственное окно, где он может перейти на окно авторизации или зарегистрироваться.

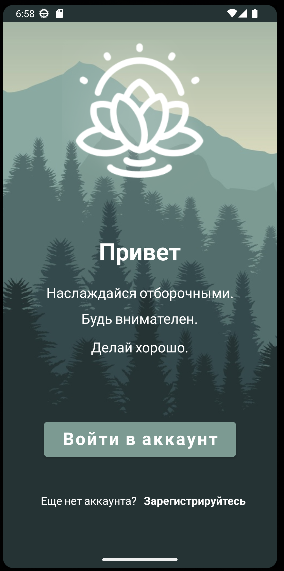


Рисунок 2.1.3 – Приветственное окно программы

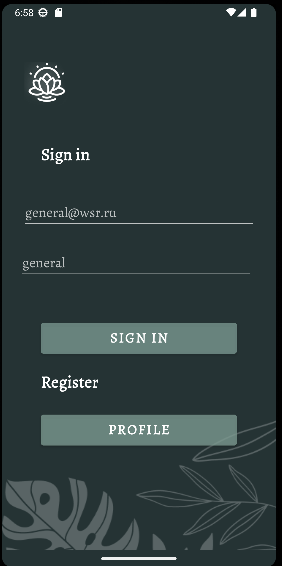


Рисунок 2.1.4 – Окно авторизации

При вводе некорректных данных, программа уведомляет об этом выдавая ошибку.

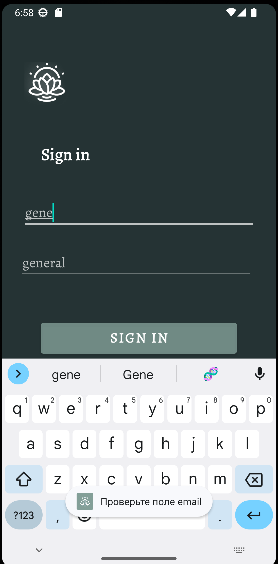


Рисунок 2.1.5 – Демонстрация валидации данных

После успешной авторизации, пользователь попадает в главное меню программы, здесь находится вся основная информация его аккаунта.

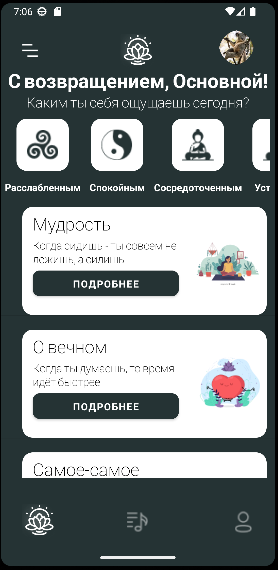


Рисунок 2.1.6 – Приветственное окно программы

В приложении есть страницы заглушки, которые открываются при переходе на соответствующие вкладки из главного меню.

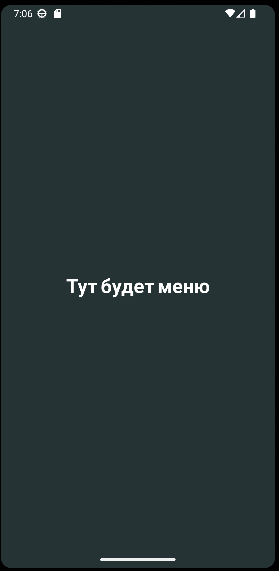


Рисунок 2.1.7 – Заглушка меню

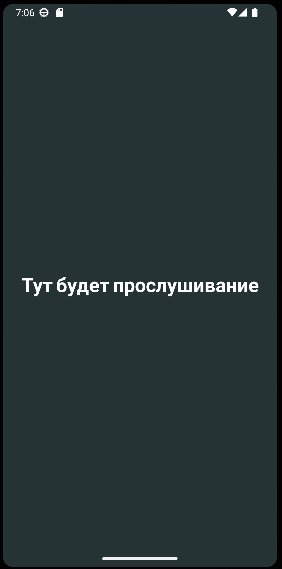


Рисунок 2.1.8 – Заглушка музыки

В будущем есть перспектива разработать окно профиля пользователя и его фотографий, но пока что открывается пустое окно.

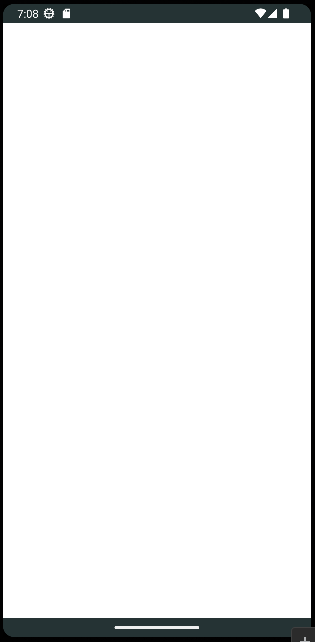


Рисунок 2.1.9 – Пустое окно с заделом на будущее

# **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ**

Монографическая литература:

1. Влацкая И.В. Проектирование и реализация прикладного программного обеспечения [Электронный ресурс] учебное пособие / И.В. Влацкая, Н.А. Заельская, Н.С. Надточий. — Электрон. текстовые данные. — Оренбург: Оренбургский государственный университет, ЭБС АСВ, 2020. — 119 c. — 978-5-7410-1238-3. — Режим доступа: http://www.iprbookshop.ru/54145.html (дата обращения 28.08.2020)
2. Федорова Г.Н. Разработка программных модулей программного обеспечения для компьютерных систем: учебник. Среднее профессиональное образование, профессиональная подготовка / Г. Н. Федорова. – М.: Академия, 2020.–336 с.
3. Журавлёва И.А. Системное и прикладное программное обеспечение [Электронный ресурс] лабораторный практикум / И.А. Журавлёва, П.К. Корнеев. — Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2017. — 132 c. — 2227-8397. —
4. Подбельский В. Язык C#. Базовый курс. Издание второе, переработанное и дополненное. Издательство: Финансы и статистика, 2020.–408 с.

Интернет-ресурсы:

1. Выбор среды разработки для создания android приложений – Режим доступа <https://infourok.ru/vibor-sredi-razrabotki-dlya-sozdaniya-android-prilozheniy-3028586.html>.
2. SQLite и Android – Режим доступа <http://developer.alexanderklimov.ru/android/sqlite/cathouse2.php>.