

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОРЕНБУРГСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«ОРЕНБУРГСКИЙ КОЛЛЕДЖ ЭКОНОМИКИ И ИНФОРМАТИКИ»
(ГАПОУ «ОКЭИ»)

ОТЧЁТ

ОКЭИ. 09.02.07. 9023 44 П

(код документа)

По профилю специальности 09.02.07 Информационные системы и программирование

ПМ.01 Осуществление интеграции программных модулей

Производственная

(вид практики)

МНТК «Микрохирургия глаза»

(место прохождения практики)

Количество страниц 25

Дата готовности 17.04.2023

Разработал Морозюк А.О 4пк2

(Ф.И.О., группа)

Руководитель: Малышев А.О.

Согласовано Начальник отдела организационно-методической работы и информационного обеспечения, Щербанов А.В.

(Должность, Ф.И.О. руководителя практики от предприятия)

Соответствие отчета с заданием на практику

(Ф.И.О. рук. практики от колледжа)

Оренбург 2023

Содержание

Введение.....	3
Характеристика предприятия.....	5
1 Структура типичного приложения.....	6
2 Реализация внедрения базы данных.....	9
3 Требования к дизайну пользовательского интерфейса.....	11
4 Информационное обеспечение.....	12
5 Техническое обеспечение.....	13
6 Программное обеспечение.....	14
7 Функционирование программы.....	16
8 Логика работы программы.....	17
9 Рефакторинг программного кода.....	19
10 Тестирование программного обеспечения.....	20
11 Техника безопасности и пожарная безопасность.....	22
Заключение.....	23
Список используемых источников.....	25

Введение

Дальтониизм, цветовая слепота - наследственная, реже приобретённая особенность зрения человека и приматов, выражающаяся в неспособности различать один или несколько цветов.

Причина нарушений цветового зрения

У человека в центральной части сетчатки расположены цветовоспринимающие рецепторы - нервные клетки, которые называются колбочками. Каждый из трёх видов колбочек имеет свой тип цветовоспринимающего пигмента белкового происхождения. Один тип пигмента чувствителен к красному цвету с максимумом 552-557 нм, другой - к зелёному, третий - к синему. Люди с нормальным цветовым зрением имеют в колбочках все три пигмента (красный, зелёный и синий) в необходимом количестве. Их называют трихроматами.

Диагностика

Характер цветового восприятия определяется на специальных полихроматических картинках. Изображение на которых состоит из множества цветных кружков и точек, имеющих одинаковую яркость, но несколько различных по цвету. Человеку с частичной или полной цветовой слепотой, не различающему некоторые цвета на рисунке, таблица кажется однородной. Человек с нормальным цветовосприятием способен различить цифры или геометрические фигуры, составленные из кружков одного цвета.

Дихроматы бывают слепыми на красный цвет, у которых воспринимаемый спектр укорочен с красного конца, и слепыми на зелёный цвет. При протанопии красный цвет воспринимается более тёмным, смешивается с тёмно-зелёным, тёмно-коричневым, а зелёный - со светло-серым, светло-жёлтым, светло-коричневым. При дейтеранопии зелёный цвет смешивается со светло-оранжевым, светло-розовым, а красный - со светло-зелёным, светло-коричневым.

Актуальность работы следует из того, что аномальное цветовосприятие может привести к тяжелым последствиям для самого человека, а также для других, взаимодействующих с ним. Исходя из этого, необходима разработка удобного и полноценного программного обеспечения для диагностики данного заболевания, чтобы при выявлении патологии человеку были бы созданы определенные, специфичные его диагнозу ограничения. Данное Desktop WPF приложение написано на языке программирования C#.

Суть проекта заключается в том, что дает возможность получить знания и умения практической разработки клиент-серверных приложений в технологии .NET Framework, с применением современных паттернов и инструментов программирования, с целью разработки многофункциональных информационных систем. Изучение языка программирования C# позволяет приобрести фундаментальные знания, которые позволят освоить любые аспекты современного программирования.

Выбор интегрированной среды Microsoft Visual Studio обусловлен тем что, самый серьезный продукт производства Microsoft для профессиональной разработки программного обеспечения.

Характеристика предприятия

Оренбургский филиал «НМИЦ «МНТК «Микрохирургия глаза» - крупнейший центр офтальмологической помощи в Южно-Уральском регионе, оснащённый новейшим уникальным операционно-диагностическим оборудованием от ведущих мировых производителей (США, Германия, Япония, Англия).

Филиал, расположенный в крупном областном центре, обслуживает не только пациентов города и области, часто обращаются к нам жители других регионов, а также стран СНГ и дальнего зарубежья.

Врачи, владеющие передовыми высокотехнологичными методиками лечения, ежедневно оказывающие помощь более 200 пациентам на уровне мировых стандартов и проводящие более 150 операций в день.

За время работы филиала обследовано более 1 350 000 пациентов, пролечено более 420 тысяч пациентов, осуществлено более 5000 выездов к пациентам отдаленных районов. Научные разработки: сотрудниками филиала опубликовано более 1200 печатных работ, получено 126 патентов на изобретения и свидетельств на полезные модели, издано 28 монографий, 2 учебника, 2 руководства, более 45 учебно-методических пособий.

За время работы филиала всего защищены: 1 докторская и 26 кандидатских диссертаций.

Оренбургский филиал МНТК – неоднократный участник и победитель многих конкурсов и Лауреат премий Губернатора и Правительства Оренбургской области.

В Оренбургском филиале МНТК ведется образовательная деятельность: повышения курсов квалификации врачей-офтальмологов (сертификационные циклы, тематические циклы и стажировка на рабочем месте), а также для среднего медицинского персонала разработан WETLAB - недельный курс усовершенствования знаний и навыков с получением удостоверения установленного образца.

1 Структура типичного приложения

Разработка ПП должна быть проведена в 3 стадии:

- Разработка технического задания;
- Рабочее проектирование
- Внедрение

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего задания. На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

- Разработка программы;
- Разработка программной документации;
- Испытание программы;

На стадии внедрения должен быть выполнен этап разработки подготовка и передача программы. На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

- Постановка задачи;
- Определение и уточнение требований к техническим средствам;
- Определение требований к программе;
- Определение стадий, этапов и сроков разработки программы и документации на нее;

Данный продукт предназначен для выявления признаков дальтонизма путем прохождения тестовых заданий, он оперирует следующими данными:

- Данные о клиенте;
- Данные о его состоянии зрения (конечный результат);

Цель создания системы:

- Проверить наличие дальтонизма у клиентов;

Также в программе есть авторизация и регистрация которые представлены на рисунках 1 и 2.



Рисунок 1 – Авторизация

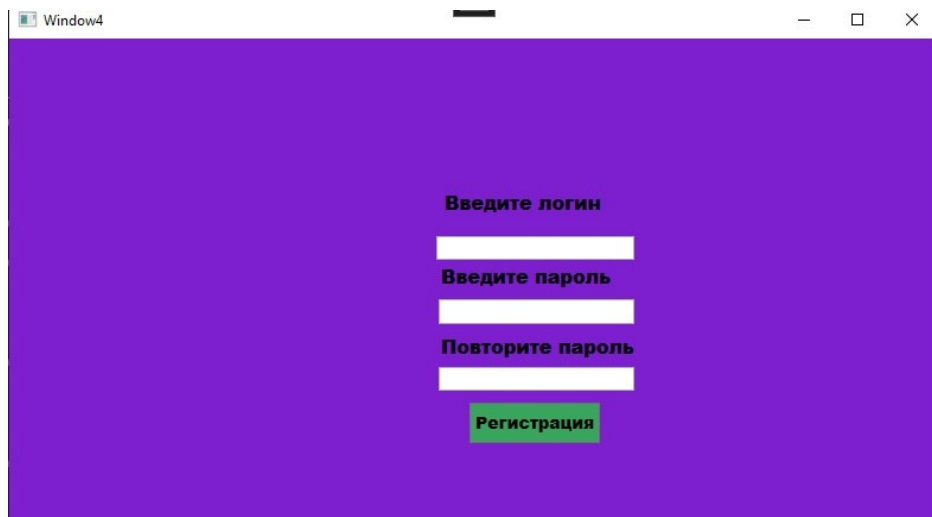


Рисунок 2 – Регистрация

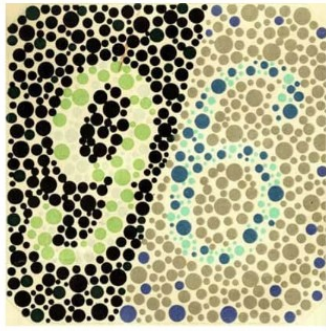
В данном приложении присутствует прохождения трех видов тестов. Это на обычный дальтонизм, на дейтеранопию и тританопию интерфейс выбора теста представлен на рисунке 3.



Рисунок 3 – Выбор тестов

После регистрации или авторизации клиент выбирает нужный ему тест как показано на рисунке 3. И начинает прохождение своего теста, рисунок 4.

MainWindow



Тест пройден, ваш результат 4 из 4, у вас все в порядке

OK

Начать заново

96

Что вы видите?

Вперед

Рисунок 4 – Прохождение теста

2 Реализация внедрения базы данных

В данном программном продукте также присутствует подключение и связь с базой данных MS SQL Server 2019. В ней отображаются данные пользователей и их результаты после тестирования, рисунок 5.

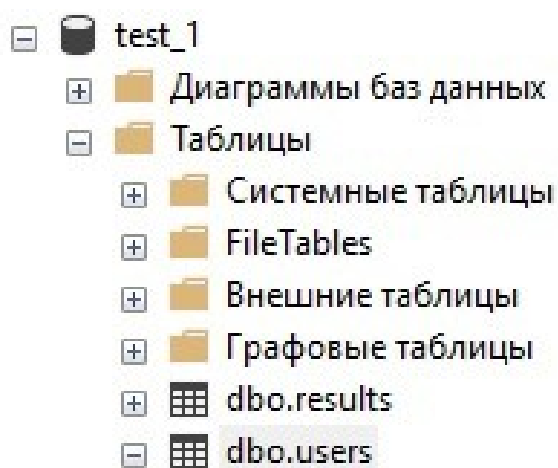


Рисунок 5 – База данных

Пользователи и результаты отображены в таблицах, рисунок 6 и 7.

	id	login	password
▶	1	Alex	1
	2	Dima	1
	3	Roma	1
	4	Max	1
	5	Den	1
	6	Dima	qwerty
	7	Alex	1
	8	Alex	1
	9		
*	NULL	NULL	NULL

Рисунок 6 - Таблица пользователей

	id	userId	diagnos
▶	1	1	тританопия
	2	1	нет дейторано...
	3	1	дальтонизм
	4	6	нет дальтонизма
	5	1	дейторанопия
	6	1	дальтонизм
	7	1	дальтонизм
	8	1	нет дальтонизма
	9	1	дальтонизм
	10	1	дейторанопия
	11	1	тританопия
*	NULL	NULL	NULL

Рисунок 7 – Таблица результатов

После авторизации либо регистрации пользователь проходит тест и его результат записывается в БД, рисунок 8 9.

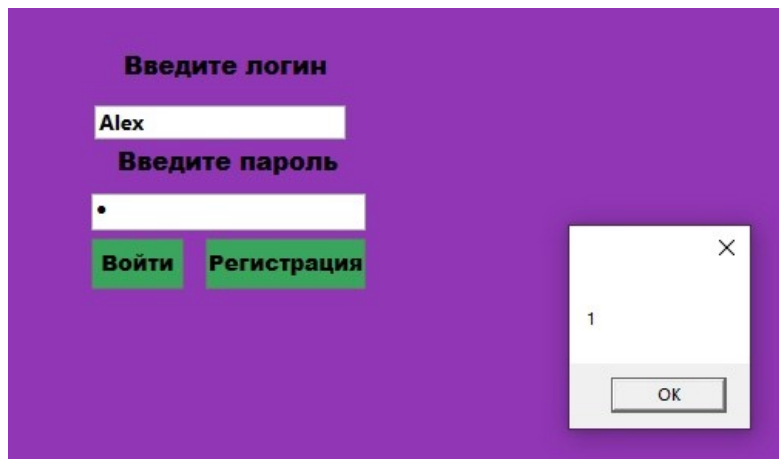


Рисунок 8 – Авторизация пользователя

12	12	11	нет дальтонизма
----	----	----	-----------------

Рисунок 9 – Вывод результат в БД

При правильной авторизации высвечивается сообщение с номером пользователя в базе как показано на рисунке 8.

3 Требования к дизайну пользовательского интерфейса

Платформа Windows Presentation Foundation (WPF) позволяет создавать клиентские приложения для настольных систем Windows с привлекательным пользовательским интерфейсом.

В основе WPF лежит независимый от разрешения векторный модуль визуализации, использующий возможности современного графического оборудования. Возможности этого модуля расширяются с помощью комплексного набора функций разработки приложений, которые включают в себя язык XAML, элементы управления, привязку к данным, макет, двумерную и трехмерную графику, анимацию, стили, шаблоны, документы, мультимедиа, текст и типографические функции. WPF является частью .NET, поэтому вы можете создавать приложения, включающие другие элементы .NET API.

При создании пользовательского интерфейса вы компоуете элементы управления, настраивая их расположение и размер. Основным требованием любого макета является адаптация к изменениям размеров окна и параметров экрана. Платформа WPF избавляет вас от необходимости писать код для адаптации макета к таким условиям, предоставляя первоклассную расширяемую систему макета.

Ключевым элементом системы макета является относительное позиционирование, которое упрощает адаптацию к меняющимся характеристикам окна и экрана. Кроме того, система макета управляет взаимодействием между элементами управления для определения макета. Взаимодействие протекает в два этапа: сначала элемент управления сообщает родительскому объекту о требуемом расположении и размере, а затем родительский объект сообщает родительскому элементу, какое пространство он может занять.

Система макета доступна дочерним элементам управления посредством базовых классов WPF. Для стандартных макетов, таких как сетка, наложение и закрепление, в WPF имеется несколько элементов управления макетом:

- Canvas: дочерние элементы управления предоставляют собственный макет.
- DockPanel: дочерние элементы управления выравниваются по краям панели.
- Grid: дочерние элементы управления упорядочиваются по строкам и столбцам.
- StackPanel: дочерние элементы управления располагаются с наложением по вертикали или по горизонтали.
- VirtualizingStackPanel: дочерние элементы управления виртуализируются и располагаются в линию по горизонтали или по вертикали.

- WrapPanel: дочерние элементы управления располагаются в порядке слева направо и переносятся на следующую строку, если не помещаются в текущей.

4 Информационное обеспечение

На основании анализа функциональных требований к разрабатываемому программному модулю проведен анализ полей класса и результат представлен в таблице 1.

Таблица 1 – Таблица пользователей

Название поля	Тип поля	Ключ
Id	Int	PK
Login	Varchar(50)	
password	Varchar	

Информация о результате пользователей после прохождения теста отображается в таблице 2.

Таблица 2 – Таблица результатов

Название поля	Тип поля	Ключ
Id	Int	
userId	Int	PK
diagnos	Varchar(50)	

5 Техническое обеспечение

Для эксплуатации данной автоматизированной системы необходимо системный блок удовлетворяющие характеристикам, представленным в таблице 3.

Таблица 3 – Характеристики системного блока

Наименование	Значение
Операционная система	Windows 10, Debian 10, MacOS 12
Процессор	Intel Core i3-10100f, AMD Ryzen 5500
Жесткий диск	SDD 250 Гб
Оперативная память	8 ГбDDR 4 2400 МГц
Видеокарта	Intel HD Graphics, AMD Vega 7

Монитор, характеристики представлены в таблице 4.

Таблица 4 – Характеристики устройства

Наименование	Значение
Диагональ	24
Максимальное разрешение	1920x1080
Соотношение сторон	16:9
Яркость	250
Время отклика пикселя	5
Угол обзора по горизонтали/вертикали	178/178
Размер пикселя	248

Также необходимы клавиатура и мышь.

6 Программное обеспечение

Программное обеспечение – это совокупность программ, позволяющих осуществить на устройстве автоматизированную обработку информации. Основополагающим элементом среди программного обеспечения, является операционная система. В логической структуре типичной вычислительной системы операционная система занимает положение между устройствами с их микроархитектурой, машинным языком и, возможно, собственными (встроенными) микропрограммами (драйверами) с одной стороны и прикладными программами - с другой.

Для запуска системы необходимо использовать операционную систему Windows, Linux, MacOS.

Windows - группа семейств коммерческих проприетарных операционных систем корпорации Microsoft, ориентированных на управление с помощью графического интерфейса, пришедших на смену MS-DOS. В отличие от текстовой однозадачной MS-DOS, операционные системы Windows являются графическими и многозадачными.

Linux - семейство Unix-подобных операционных систем на базе ядра Linux, включающих тот или иной набор утилит и программ проекта GNU, и, возможно, другие компоненты. Как и ядро Linux, системы на его основе, как правило, создаются и распространяются в соответствии с моделью разработки свободного и открытого программного обеспечения.

Операционная система образует автономную среду, не связанную ни с одним из языков программирования. Любая же прикладная программа связана с операционной системой и может эксплуатироваться только на тех компьютерах, где имеется аналогичная системная среда. Прикладные программные средства, разработанные в среде одной операционной системы, не могут быть использованы для работы в среде другой операционной системы, если нет специального комплекса программ (конверта), позволяющего это сделать. В таком случае говорят о программной несовместимости компьютеров/

Для функционирования системы, должны быть установлены следующие программы:

- DOTNETCORE 7 - это модульная платформа для разработки программного обеспечения с открытым исходным кодом, основана на .NETFramework, отличается от неё модульностью, кроссплатформенностью, возможностью применения облачных технологий, и тем, что в ней произошло разделение между библиотекой CoreFX и средой выполнения CoreCLR;

- Node.js — программная платформа, основанная на движке V8 (компилирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения. Node.js добавляет

возможность JavaScript взаимодействовать с устройствами ввода-вывода через свой API, написанный на C++, подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript-кода. Node.js применяется преимущественно на сервере;

7 Функционирование программы

В ходе разработки ПП «Тест на дальтонизм» были разработаны следующие функции:

- Приложение после прохождения теста должно вносить результаты пользователей в базу данных;
- Приложение должно хранить в базе данных имена пользователей при авторизации в систему;
- Приложение должно конкретно выводить результат клиенту после прохождения;
- Все компоненты программы должны работать без ошибок;
- У окон авторизации и регистрации пользователей должен быть дизайн;
- Каринки тестов для прохождения должны выгружаться с json файлов и конкретно отображаться в окнах;
- В данном проекте должна присутствовать связь и подключение с базой данных;
- После авторизации пользователя должно выводиться сообщение с его номеров в базе данных;
- После регистрации нового пользователя должна быть обратная связь с ним и выводиться сообщение, что регистрация прошла успешно;
- Также приложения должно добавлять нового пользователя в базу данных после регистрации;

8 Логика работы программы

Схематически архитектуру WPF можно представить следующим образом:

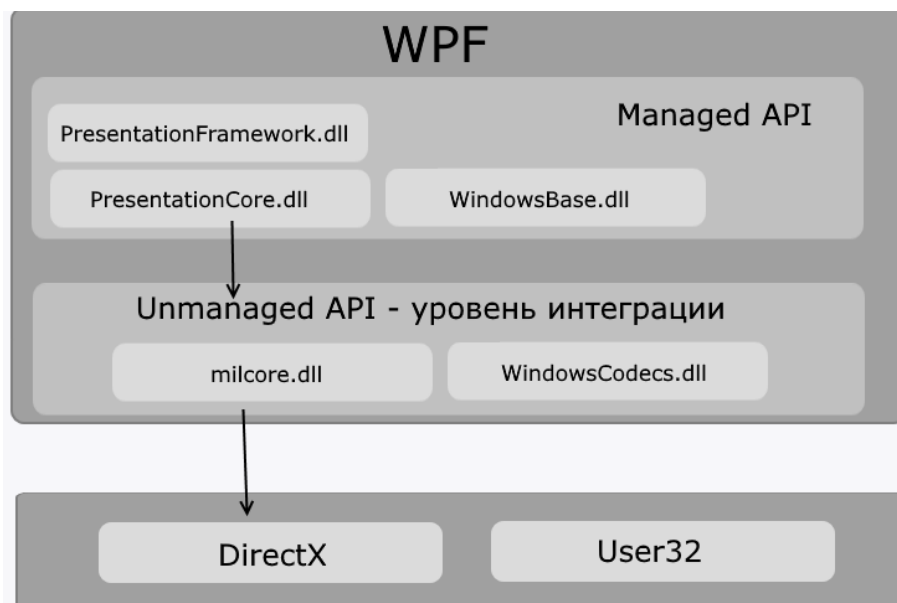


Рисунок 13 – Архитектура WPF приложения

Как видно на схеме, WPF разбивается на два уровня: managed API и unmanaged API (уровень интеграции с DirectX). Managed API (управляемый API-интерфейс) содержит код, исполняемый под управлением общезыковой среды выполнения .NET - Common Language Runtime. Этот API описывает основной функционал платформы WPF и состоит из следующих компонентов:

- PresentationFramework.dll: содержит все основные реализации компонентов и элементов управления, которые можно использовать при построении графического интерфейса
- PresentationCore.dll: содержит все базовые типы для большинства классов из PresentationFramework.dll
- WindowsBase.dll: содержит ряд вспомогательных классов, которые применяются в WPF, но могут также использоваться и вне данной платформы

Unmanaged API используется для интеграции вышележащего уровня с DirectX:

- milcore.dll: собственно обеспечивает интеграцию компонентов WPF с DirectX. Данный компонент написан на неуправляемом коде (C/C++) для взаимодействия с DirectX.
- WindowsCodecs.dll: библиотека, которая предоставляет низкоуровневую поддержку для изображений в WPF.

Еще ниже собственно находятся компоненты операционной системы и DirectX, которые производят визуализацию компонентов приложения, либо выполняют прочую низкоуровневую обработку. В частности, с помощью низкоуровневого интерфейса Direct3D, который входит в состав DirectX, происходит трансляция.

Здесь также на одном уровне находится библиотека user32.dll. И хотя выше говорилось, что WPF не использует эту библиотеку для рендеринга и визуализации, однако для ряда вычислительных задач (не включающих визуализацию) данная библиотека продолжает использоваться.

Таблица 5 – Управляющие элементы фрагмента запись

Тип элемента управления	Имя элемента	Какие действия выполняет
Button	logButton	Авторизует пользователя
Button	regButton	Регистрирует пользователя
Button	Button	Выбирает прохождение теста на дальтонизм
Button	Button2	Выбирает прохождение теста на дейтеранопию
Button	Button3	Выбирает прохождение теста на тританопию
Button	NextButton	Переключает на следующую картинку

9 Рефакторинг программного кода

Рефакторинг (англ. refactoring), или перепроектирование кода, переработка кода, равносильное преобразование алгоритмов — процесс изменения внутренней структуры программы, не затрагивающий её внешнего поведения и имеющий целью облегчить понимание её работы. В основе рефакторинга лежит последовательность небольших эквивалентных (то есть сохраняющих поведение) преобразований. Поскольку каждое преобразование маленькое, программисту легче проследить за его правильностью, и в то же время вся последовательность может привести к существенной перестройке программы и улучшению её согласованности и чёткости.

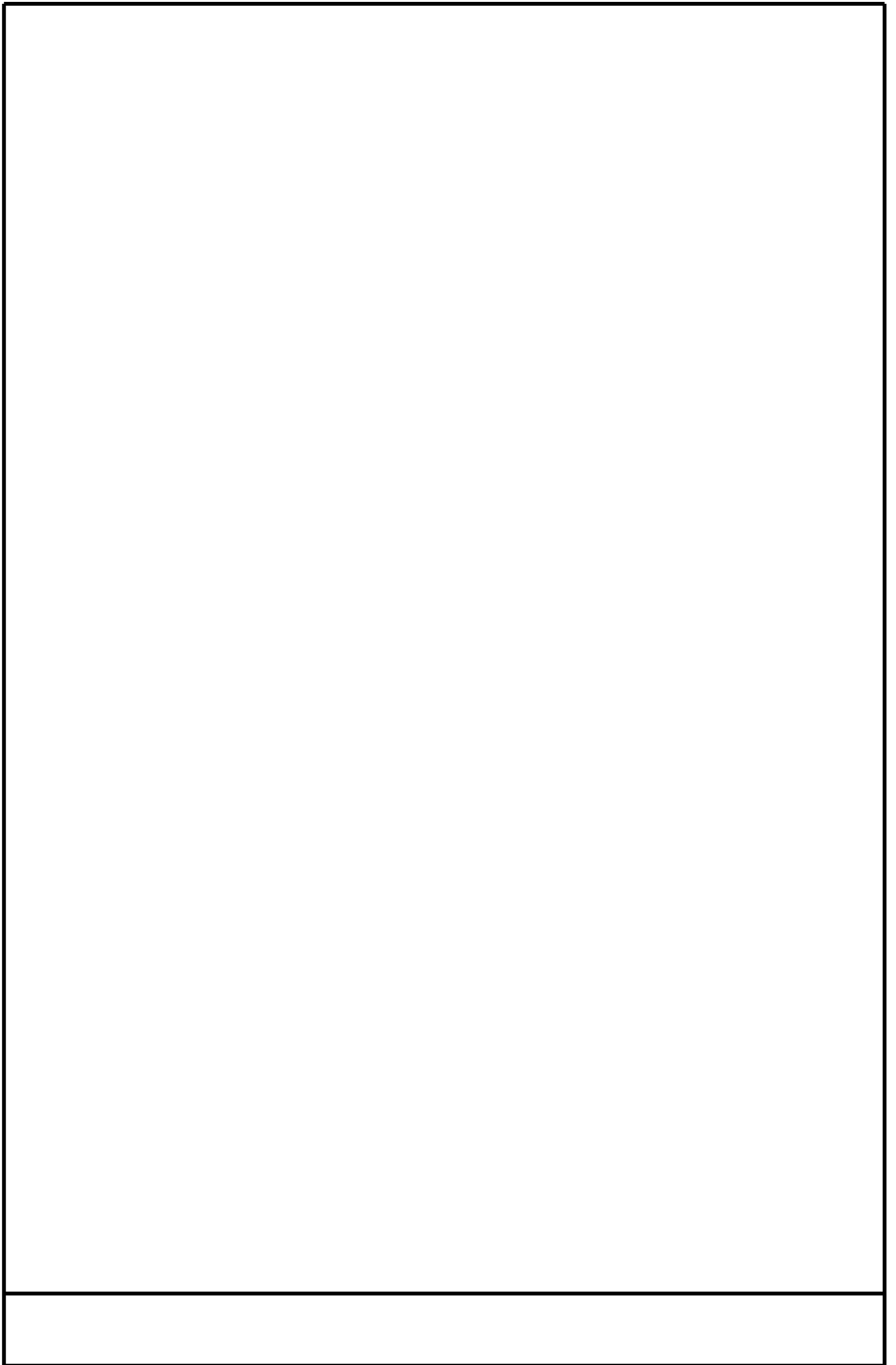
Цель рефакторинга — сделать код программы более легким для понимания, без этого рефакторинг нельзя считать успешным.

Рефакторинг нужно применять постоянно при разработке кода, основными стимулами его проведения являются следующие задачи:

- необходимо добавить новую функцию, которая недостаточно укладывается в принятое архитектурное решение;
- необходимо исправить ошибку, причины возникновения которой сразу не ясны;
- преодоление трудностей в командной разработке, которые обусловлены сложной логикой программы.

Наиболее употребимые методы рефакторинга:

- изменение сигнатуры метода (change method signature);
- инкапсуляция поля (encapsulate field);
- выделение класса (extract class);
- выделение интерфейса (extract interface);
- выделение локальной переменной (extract local variable);
- выделение метода (extract method);
- генерализация типа (generalize type);
- встраивание (inline);
- введение фабрики (introduce factory);
- введение параметра (introduce parameter);
- подъём метода (pull up method);
- спуск метода (push down method);
- переименование метода (rename method);
- перемещение метода (move method);
- замена условного оператора полиморфизмом (replace conditional with polymorphism);
- замена наследования делегированием (replace inheritance with delegation);



10 Тестирование программного обеспечения

Начало работы приложения это авторизация и регистрация пользователя

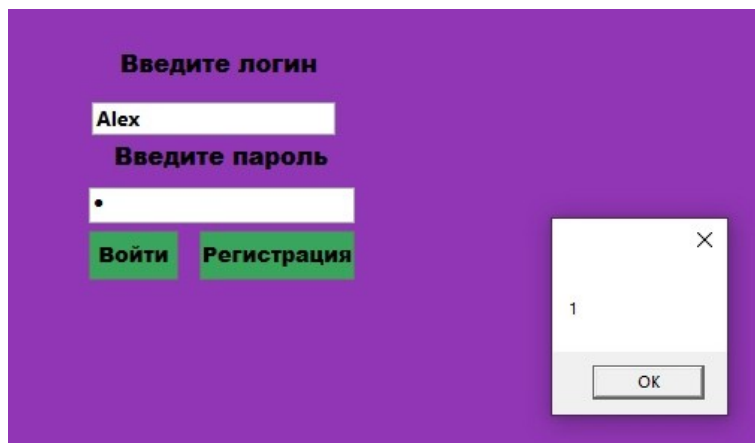


Рисунок 14 – Авторизация

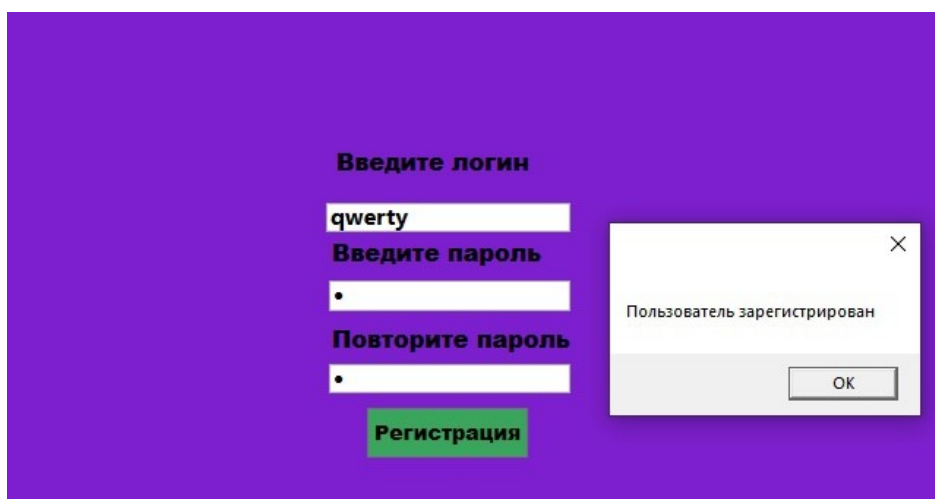


Рисунок 15 – Регистрация

Далее в приложении происходит выбор тестирования. Это на обычный дальтонизм, на дейтеранопию и тританопию.



Рисунок 16 – Выбор тестов

После регистрации или авторизации клиент выбирает нужный ему тест, и начинает прохождение своего теста.



Рисунок 17 – Прохождение теста

Все остальные тестовые требования приведены в таблице 6.

Таблица 6 – Тестовые требования

№	Тест требования	Результат
1	Между кнопками, стоящими рядом, должно быть пустое пространство, щелчок по которому не обрабатывается.	Выполняется
2	Все элементы интерфейса выполнены в едином стиле	Выполняется
3	Все элементы раскрашены в цвета из функциональных требований	Выполняется
4	При нажатии на кнопку сохранить поля проверяются на пустые значения	Выполняется
5	При неправильном вводе выводит всплывающее сообщение	Выполняется

11 Техника безопасности и пожарная безопасность

При внедрении, эксплуатации и обслуживании технических средств системы должны выполняться меры электробезопасности в соответствии с «Правилами устройства электроустановок» и «Правилами техники безопасности при эксплуатации электроустановок потребителей».

Аппаратное обеспечение системы должно соответствовать требованиям пожарной безопасности в производственных помещениях по ГОСТ 12.1.004-91. «ССБТ. Пожарная безопасность. Общие требования».

Должно быть обеспечено соблюдение общих требований безопасности в соответствии с ГОСТ 12.2.003-91. «ССБТ. Оборудование производственное. Общие требования безопасности» при обслуживании системы в процессе эксплуатации.

Аппаратная часть системы должна быть заземлена в соответствии с требованиями ГОСТ Р 50571.22-2000. «Электроустановки зданий. Часть 7. Требования к специальным электроустановкам. Раздел 707. Заземление оборудования обработки информации».

Значения эквивалентного уровня акустического шума, создаваемого аппаратурой системы, должно соответствовать ГОСТ 21552-84 «Средства вычислительной техники. Общие технические требования, приемка, методы испытаний, маркировка, упаковка, транспортирование и хранение», но не превышать следующих величин:

- 50 дБ - при работе технологического оборудования и средств вычислительной техники без печатающего устройства;
- 60 дБ - при работе технологического оборудования и средств вычислительной техники с печатающим устройством.

При использовании вычислительной техники необходимо соблюдать требования по обеспечению пожарной безопасности.

На рабочем месте запрещается иметь огнеопасные вещества.

В помещениях запрещается:

- зажигать огонь;
- включать электрооборудование, если в помещении пахнет газом;
- курить;
- закрывать вентиляционные отверстия в электроаппаратуре.

При возникновении пожароопасной ситуации или пожара персонал должен немедленно принять необходимые меры для его ликвидации, одновременно оповестить о пожаре администрацию.

Заключение

В ходе прохождения производственной практики были изучены цели, задачи, общие принципы, нормативная документация, виды и направления деятельности предприятия. В частности, произошло ознакомление с организационной структурой организации, ее ролью в социальной и экономической деятельности региона и была изучена система документооборота, в том числе и автоматизированная внутри этого предприятия. Были выполнены данные задачи:

- изучение деятельности предприятия, входной и выходной информации;
- определение предметной области для своей задачи;
- проведение предпроектного исследования: создание информационной и функциональной модели;
- разработка технического задания;
- разработка требований к программному обеспечению;
- выполнение проектирования ПО для решения прикладной задачи;
- создание базы данных;
- заполнение таблицы файла БД данными;
- создание формы для корректировки оперативной информации системы с использованием справочных данных;
- Присоединение базы данных к проекту;
- Создание desktop WPF приложения;
- формирование отчета, согласно требованиям ГОСТ;

По распоряжению руководителя проекта от образовательной организации были изучены основные принципы ведения документов и закреплена, изученная до этого на занятиях теория, необходимая для построения баз данных, что позволило в дальнейшем ускорить и оптимизировать работу.

Во время практики были рассмотрены результаты работ предприятий, что позволило получить представление о характере и качестве предоставляемых ей услуг. Во время выполнения проекта были выполнены порученные руководителем практики задания, связанные со сбором, обработкой информации и использованием этой информации в целях для описания и постановки задачи разрабатываемого приложения что позволило узнать о её необходимости и необходимых характеристиках подробнее.

В разрабатываемом приложении «Тест на дальтонизм» были реализованы функции:

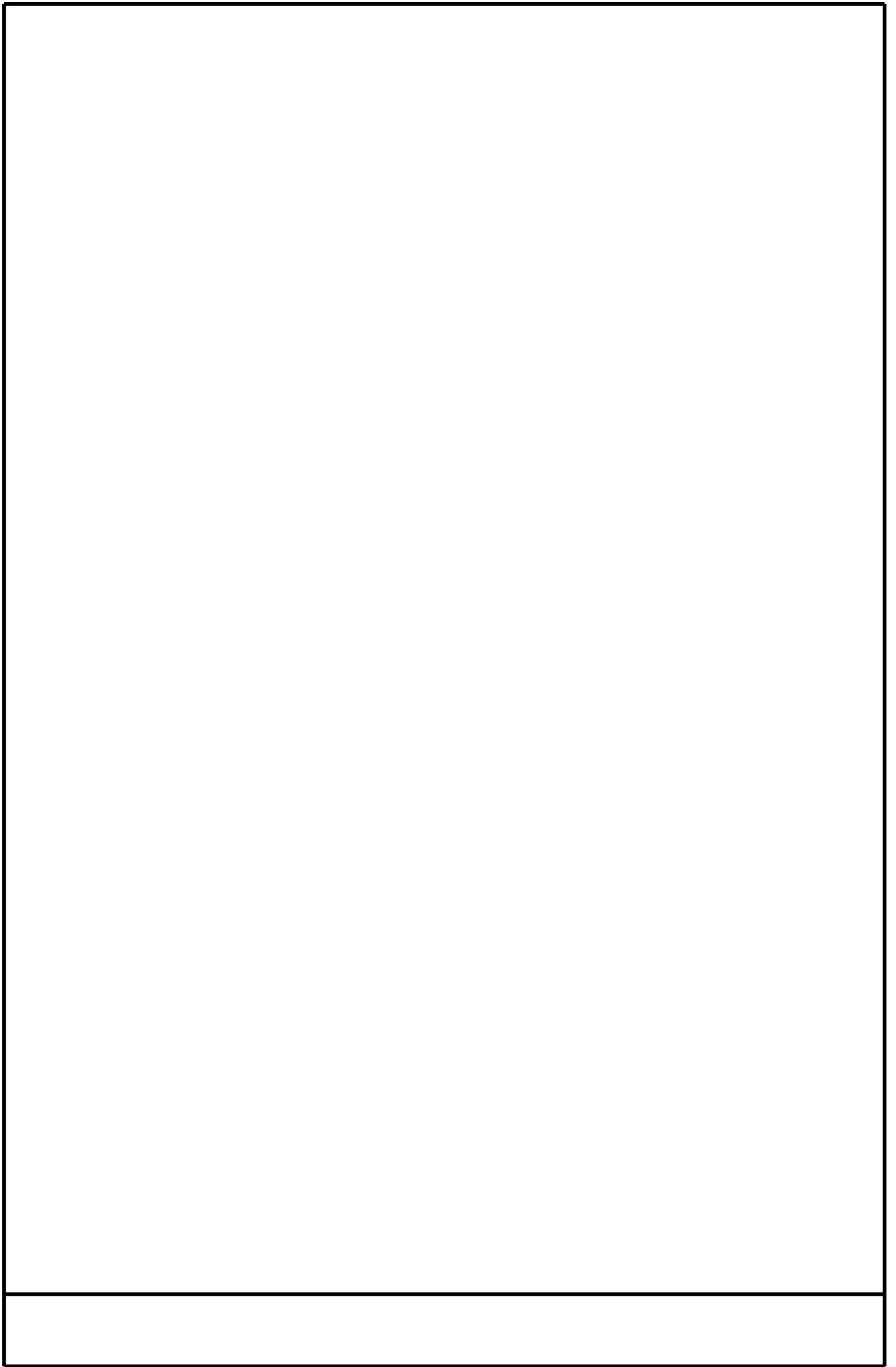
- авторизация;
- регистрация;
- вывод результатов;
- отображение информации в БД.

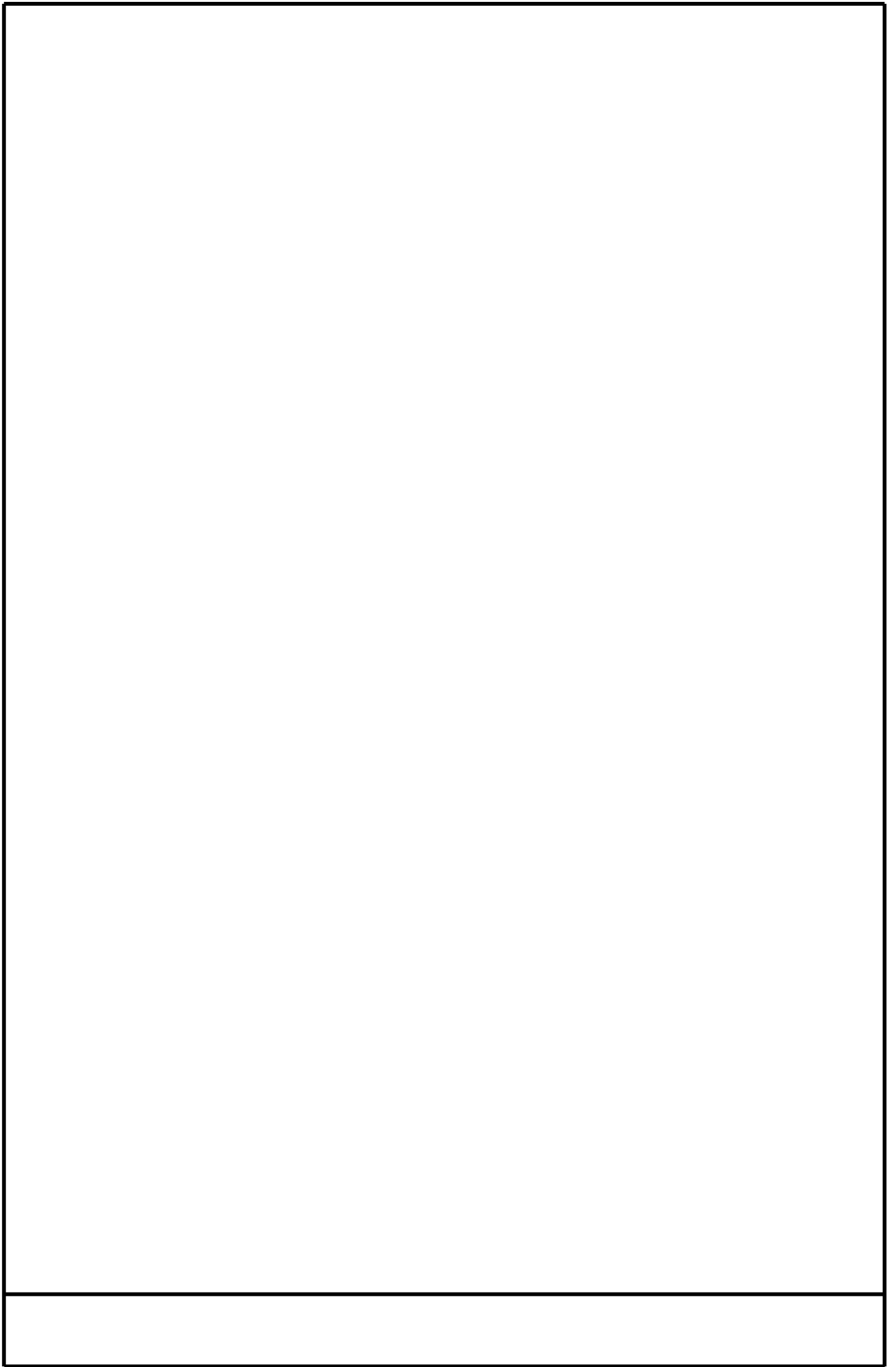
Функционал был полностью реализован в соответствии с поставленными задачами и соответствуют требованиям к разрабатываемой информационной базе.

Производственная практика является одним из основных условий закрепления полученных знаний, приобретения практических навыков по их применению, а также выявлению пробелов в знаниях теории; дает возможность получить начальную практическую подготовку, оценить свои возможности и определиться в своей будущей специальности.

Список используемых источников

- 1 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.wpf.com/guide/navigation/navigation-principles>
- 2 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/guide/fragments> - 30.06.2022
- 3 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/topic/libraries/architecture/viewmodel> - 30.06.2022
- 4 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/topic/libraries/architecture/viewmodel/viewmodel-apis> - 30.06.2022
- 5 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/topic/libraries/architecture/livedata> - 30.06.2022
- 6 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/develop/ui/views/device-control> - 30.06.2022
- 7 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/develop/ui/views/home-channels/read> - 30.06.2022
- 8 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/develop/ui/views/text-and-emoji/downloadable-fonts> - 30.06.2022
- 9 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/develop/ui/views/graphics> - 30.06.2022
- 10 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/develop/ui/views/components/settings/handle-other-form-factors> - 30.06.2022
- 11 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/develop/ui/views/layout/linear> - 30.06.2022
- 12 WPF Developers [Электронный ресурс]: учеб. пособие. – Режим доступа – <https://developer.android.com/develop/ui/views/layout/recyclerview-custom> - 30.06.2022
- 13 ГОСТ 2.104-2006 ЕСКД. Единая система конструкторской документации. Основные надписи. - Введ. 2006-09-01. - Москва: Изд-во стандартов, 2001. - 27 с.
- 14 ГОСТ 2.105-95 ЕСКД. Единая система конструкторской документации. Общие требования к текстовым документам. - Введ. 1996-07-01. - Москва: Изд-во стандартов, 2001. - 23 с.
- 15 ГОСТ 7.32-2017 Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления. - Введ. 2018-07-01. - Москва: Изд-во стандартов, 2020. – 1





Приложение А (обязательное)

Входные данные

id	login	password
1	Alex	1

Таблица А.1 – Регистрация пользователя Alex

2	Dima	1
---	------	---

Таблица А.2 – Регистрация пользователя Dima

3	Roma	1
---	------	---

Таблица А.3 – Регистрация пользователя Roma

Приложение Б (обязательное)

Выходные данные

id	userid	diagnos
1	1	тристанопия

Таблица Б.1 – Результат пользователя Alex

2	1	нет дейторано...
---	---	------------------

Таблица Б.2 – Результат пользователя Dima

3	1	дальтонизм
---	---	------------

Таблица Б.3 – Результат пользователя Roma

Приложение В (обязательное)

Диаграмма «Use case»

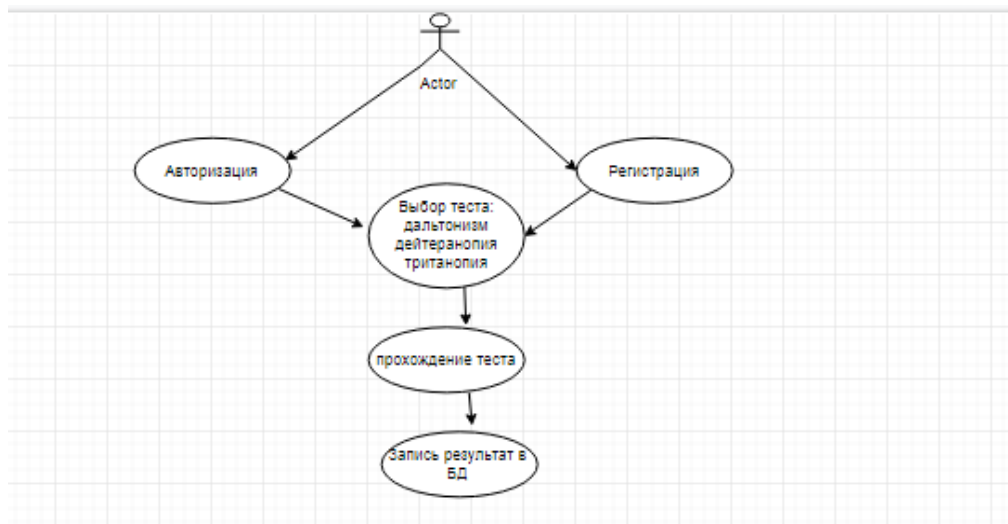


Рисунок В.1 – Диаграмма «Use case»

Приложение Г (обязательное)

Диаграмма деятельности

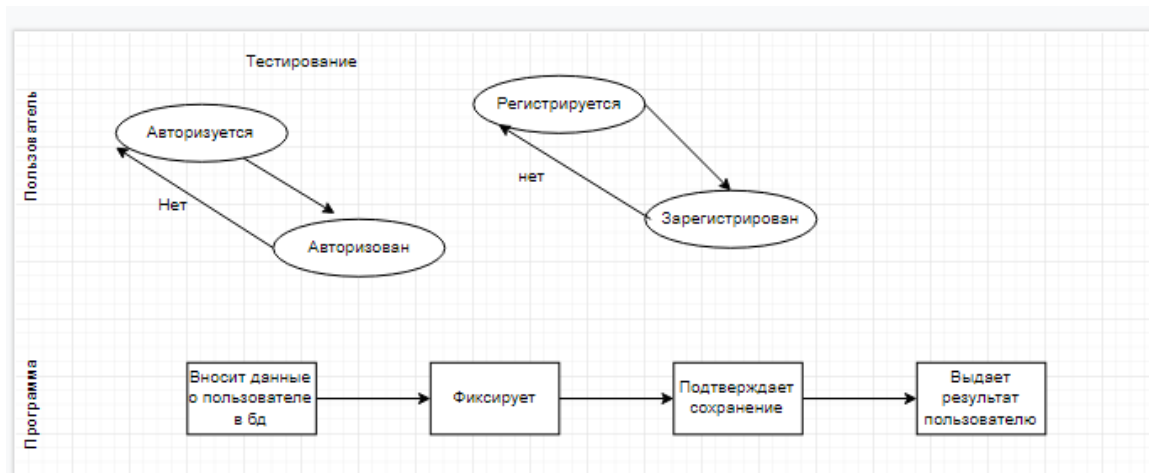


Рисунок Г.1 – Диаграмма деятельности