

ДИПЛОМНЫЙ ПРОЕКТ

по специальности 09.02.07 Информационные системы и программирование
код и наименование специальности

по теме: Разработка программного продукта для проверки зрения

Разработал студент _____ Ю.Р. Асадуллина группы ИСП-433
подпись *И.О. Фамилия*
_____ 20 _____

Руководитель дипломного проекта _____
подпись *И.О. Фамилия*

Помощь с ВКР под ключ!
Задание принял к исполнению _____
подпись *И.О. Фамилия*
пишем-вкр-сами.рф _____ 20 _____

Нормоконтролер _____
подпись *И.О. Фамилия*
_____ 20 _____

Рецензент _____
подпись *И.О. Фамилия*
_____ 20 _____

Самара 2022 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
ГЛАВА 1 ОБЩАЯ ТЕОРЕТИЧЕСКАЯ ЧАСТЬ	6
1.1 Анализ предметной области как объекта информатизации	6
1.2 Анализ информационного обеспечения объекта информатизации	7
1.3 Организационные и функциональные схемы объектов информатизации	8
1.4 Описание категорий пользователей и потоков данных объекта информатизации	11
ГЛАВА 2 ПРЕДПРОЕКТНАЯ ЧАСТЬ	12
2.1 Анализ функциональных требований	12
2.2 Анализ входных и выходных данных	17
2.3 Разработка тестовых сценариев, программы и методики испытаний	18
2.4 Разработка технического задания программного продукта	21
2.5 Выбор программного обеспечения для реализации (разработки) программного продукта	28
2.6 Выбор программного обеспечения для разработки и эксплуатации удаленных базы данных	29
2.7 Разработка и оптимизация схемы данных программного продукта	30
ГЛАВА 3 ПРОЕКТНАЯ ЧАСТЬ	32
3.1 Разработка интерфейса программного продукта	32
3.2 Разработка удаленной базы данных и необходимых запросов	38
3.3 Реализация функциональных требований	39
3.4 Отладка и тестирование программного продукта	48
ГЛАВА 4 ДОКУМЕНТАЦИОННОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММНОГО ПРОДУКТА	50
4.1 Документирование программного кода	50
4.2 Руководство пользователя программы	52
4.3 Руководство администратора по ГОСТ 19.503-79	58

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

ГЛАВА 5 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ	63
5.1 Разработка мер защиты информации от несанкционированного доступа	
63	
5.2 Мероприятия по обеспечению техники безопасности и пожарной	
безопасности в документе	64
ЗАКЛЮЧЕНИЕ	68
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	70
ПРИЛОЖЕНИЕ А UML - диаграммы	75
ПРИЛОЖЕНИЕ Б Руководство пользователя (на английском)	77
ПРИЛОЖЕНИЕ В Руководство администратора (на английском)	83
ПРИЛОЖЕНИЕ Г Документированный код программного продукта	87
ПРИЛОЖЕНИЕ Д Диск с материалами	110

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

ВВЕДЕНИЕ

В наше время технологии играют огромную роль в жизни любого человека, ведь они помогают решить огромное количество проблем и задач. Однако регулярное использование гаджетов очень сильно влияет на зрение и к сожалению не в лучшую сторону, в связи с этим огромное количество людей регулярно посещает окулиста, и в первую очередь окулист проведет проверку на остроту зрения.

Самым популярным тестом на остроту зрения является таблица Сивцева. Она представляет собой таблицу из 12 строк с буквами, которые становятся меньше и меньше с каждой строкой. Помимо букв каждая строка имеет такие характеристики как острота зрения, и расстояние с которого эту букву должен видеть человек со 100% зрением. Тестирование происходит следующим образом – пациент, закрыв один глаз, должен назвать букву, на которую указывает врач.

Необходимость разработки программного продукта заключается в том, что, несмотря на все плюсы данного теста, из-за однообразности таблицы многие ее заучивают специально, или же машинально. Из за этого результаты являются в лучшем случае примерными, а в худшем совершенно неверными, вследствие чего дальнейшее лечение принесет больше вреда, чем пользы, поэтому программный продукт должен случайно генерировать буквы (или другие картинки), а также соблюдать стандарты основной таблицы.

Актуальность работы заключается в разработке программного продукта для программы проверки зрения по таблице Сивцева, на основе которой проверяется острота зрения, является однообразной и легко запоминаемой.

В связи с этими проблемами удалось выявить цель настоящего проекта: «Разработка программного продукта для проверки зрения».

Цель исследования: разработка программного продукта для проверки зрения ГБУЗ СОКОБ им Т.И. Ерошевского.

Объект: разработка информационной системы для программы «Проверка зрения».

Предмет исследования: средства и методы разработки системы для проверки зрения.

Гипотеза исследования: если показ картинок будет происходить в случайном порядке, то это позволит получить более точные результаты тестирования.

Программный продукт будет максимально эффективным, если:

- правильно описаны функциональные требования программного продукта;
- правильно подобрано программное обеспечение для разработки программного продукта;
- разработаны оптимальный, удобный и понятный интерфейс пользователя;
- реализованы и протестированы все функции программного продукта;
- техническая документация к программе будет разработана в соответствии с ГОСТ;
- уменьшится вероятность потери информации;
- будут соблюдены все нормы и правила безопасности при работе;
- будет возможность редактирования данных;
- генерирование тестов будет проходить без повторений;

Для достижения данных целей необходимо выполнить такие задачи как:

- провести анализ объекта, процесса, предметной области как объекта информатизации;
- провести анализ информационного обеспечения объекта информатизации;

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

- построить организационные и функциональные схемы объектов информатизации;
- описать категории пользователей и потоков данных объекта информатизации;
- провести анализ функциональных требований программного продукта;
- провести анализ входных и выходных данных;
- построить структуру данных;
- спроектировать функционал разрабатываемой системы;
- разработать тестовые сценарии, программы и методики испытаний;
- разработать техническое задание для программного продукта;
- разработать и оптимизировать схемы данных для программного продукта;
- выбрать программное обеспечение для реализации (разработки)

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

- выбрать программное обеспечение для реализации (разработки) программного продукта;
- выбрать программное обеспечение для реализации и эксплуатации удаленных базы данных;
- разработать графический интерфейс на русском и английском языках для программного продукта;
- разработать удаленную базу данных и необходимые запросы;
- реализовать функциональные требования;
- описать интеграцию программных модулей;
- выполнить отладку программного продукта;
- провести тестирование программного продукта;
- выполнить документирование программного кода;
- разработать руководство пользователя для программного продукта;
- разработать меры защиты информации от несанкционированного доступа;
- описать мероприятия по обеспечению техники безопасности и пожарной безопасности.

ГЛАВА 1 ОБЩАЯ ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1.1 Анализ предметной области как объекта информатизации

Процесс анализа предметной области начинается с определения концептуальных требований ряда пользователя [7].

Концептуальное проектирование включает описания объектов и их взаимосвязей, представляющих интерес в рассматриваемой предметной области и выявляемых в результате анализа данных, который может проводиться с помощью традиционных методик – анкетирования и интервьюирования. На этом этапе выявляются заинтересованные лица (все те, кто имеет прямое или косвенное отношение к процессу, автоматизация которого производится), их потребности и соответственно функции программного продукта.

ГБУЗ Сокоб им. Т.И. Ерошевского - самарская областная клиническая офтальмологическая больница, которая занимается оказанием специализированной медицинской офтальмологической помощи по всей самарской области и за ее пределами.

Данное медицинское учреждение работает с 1969 года. Инициатором создания больницы является Тихон Иванович Ерошевский, в честь него и названо это учреждение.

В состав больницы входит 4 корпуса и более 7 отделений для оказания помощи, благодаря этому больница имеет большую и разнообразную специализацию (от подбора очков до лечения глауком). Также каждому пациенту обеспечивается полноценный и оперативный доступ к новейшим достижениям медицины.

Ежегодно квалифицированные специалисты из больницы оказывают помощь более десяткам тысяч пациентов из Самары, самарской области и

других регионов.

Одним из главных отделений является отделение диагностики, которое работает со времен основания больницы и помогает пациенту улучшить зрение при помощи различных тренажеров, проверок и гимнастик.

Для оказания помощи используется только современные и высокотехнологичные методы исследования. Также постоянно проводится внедрение новых диагностических систем, которые позволяют увеличить эффективность работы отделения, скорость приема пациентов, а также пропускную способность.

Диагностическое отделение является одним из первых мест в данной больнице, куда приходит пациент, там проводят целенаправленные исследования для определения первичных причин проблем со зрением.

Данное отделение выполняет такие функции как:

- Выявление первичных проблем со зрением у пациента;
- Выдача доступных результатов исследования не позднее 2 дней;
- Оказание своевременной и качественной медицинской помощи;
- Ранняя диагностика и оценка здоровья пациента.

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

1.2 Анализ информационного обеспечения объекта информатизации

Во время своей работы сотрудники СОКОМ им Т.И. Ерошевского используют следующее аппаратное обеспечение:

- Видеокарта NVIDIA Quadro P1000;
- Процессор intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz 2.40 GHz;
- Материнская плата ASUS TUF Z390M-PRO GAMING;
- Охлаждение HYPERPC Water Cooling 120 RGB;

- Оперативная память 2 x 10GB;
- SSD накопитель 240GB Kingston A1000;
- Жесткий диск 1TB Seagate BarraCuda;
- Блок питания Cooler Master V750 750W.

Также используются такие программные продукты как:

- Операционная система Windows 10;
- Microsoft Office 2020;

1.3 Организационные и функциональные схемы объектов информатизации

Организационная схема показывает взаимодействие основных отделов и служб организации, представленная на рисунке 1

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

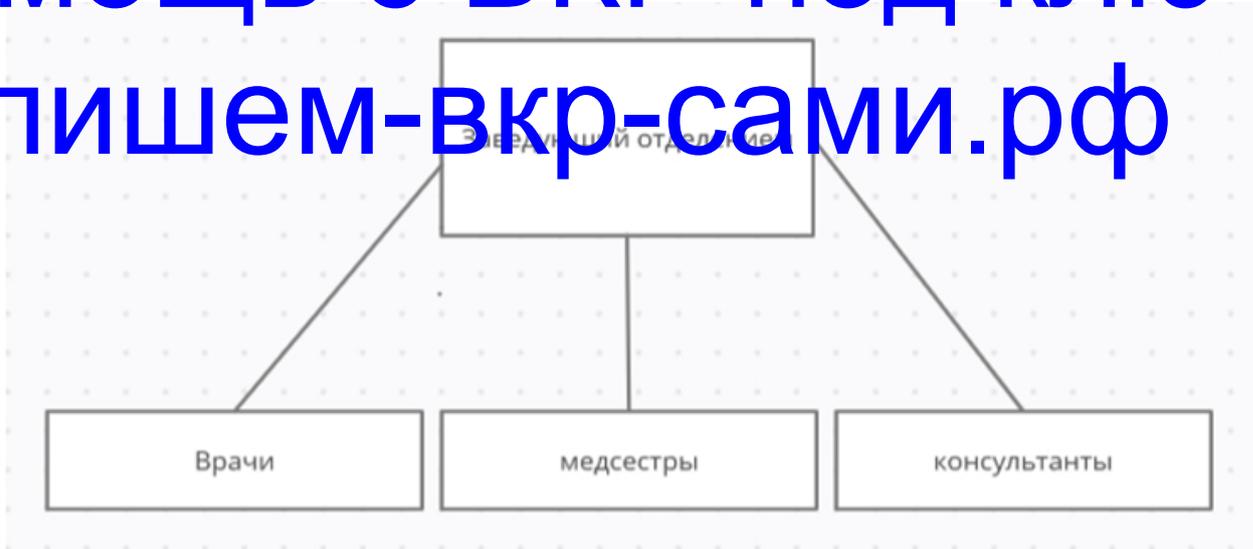


Рисунок 1 - Организационная схема

Для получения необходимой информации об объекте автоматизации, поиска процессов, которым необходима автоматизация и для определения потребностей непосредственно заказчика и пользователей будущей информационной системы были использованы такие способы как:

- Беседа с сотрудниками (интервью);
- Анкетирование сотрудников объекта информатизации;
- Наблюдение за выполнением рабочих процессов сотрудниками.

В результате обработки результатов, полученных в ходе анализа, получены следующие данные:

- Список заинтересованных в информационной системе лиц;
- Функциональные обязанности сотрудников диагностического отделения;
- Список потребностей заинтересованных лиц в разрабатываемой информационной системе;
- Описание объекта информатизации и рабочих процессов;
- Модель объекта информатизации.

При изучении функциональных обязанностей сотрудника данного структурного подразделения было выяснено следующее: Устный, должностным инструкциям, положением о структурном подразделении, в котором работает сотрудник, положениями в компании и различными документированными процедурами.

Функциональные обязанности сотрудника диагностического отделения:

- Организует все необходимые диагностические процедуры или мероприятия;
- Проводит диагностические процедуры;
- Оказывает пациентам помощь консультативного характера;
- Систематически повышает свою квалификацию;
- Своевременно готовит и предоставляет необходимую документацию.

Было проведено анкетирование сотрудника отделения, на основании которого сделан вывод, что сотрудник заинтересован в разработке информационной системы для проверки зрения.



Рисунок 2 – Функциональная схема

Вначале пользователю необходимо авторизоваться в программе, если он вводит неверные данные, то появляется ошибка и сообщение о том, что пользователю нужно повторить попытку авторизации. Если пользователь всё же ее прошёл, то у него открывается главная форма программы.

1.4 Описание категорий пользователей и потоков данных объекта информатизации

На основе проведенного интервью были выявлены следующие категории пользователей и потоков данных объекта информатизации, расположенные в таблице 1.

Таблица 1

Категории пользователей

Категория пользователей	Функциональные возможности
Администратор	Управление списком результатов Управление списком врачей Управление списком пациентов Добавление пользователей Настройка подключения базы данных Проведение и включение проверки Изменение набора карточек Смена языка Смена цветовой гаммы
Врач	Управление списком результатов Управление списком врачей Управление списком пациентов Проведение и включение проверки Изменение набора карточек Смена языка Смена цветовой гаммы

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

ГЛАВА 2 ПРЕДПРОЕКТНАЯ ЧАСТЬ

2.1 Анализ функциональных требований

Для определения функциональных требований к программному продукту необходимо провести интервью с сотрудником, который заинтересован в автоматизировании процесса проверки на остроту зрения.

Интервьюирование - это метод получения необходимой информации во время устного общения между двумя собеседниками — интервьюером (тем, кто спрашивает) и респондентом (отвечающим). Одним из важных условий интервью является регистрация и анализ ответов на вопросы [21, с.94].

Интервьюирование является важнейшим и необходимым методом обследования, только с его помощью возможно разобраться во всех тонкостях применяемых на предприятии технологий. Проведение интервью представляет собой общение с предполагаемым заказчиком и заполнение определенной анкеты.

Анкета представляет собой специально разработанный опросный лист для получения определенных сведений о том, кто его заполняет. Анкеты позволяют составить грубое представление о деятельности предприятия, что позволит спланировать первоначальное распределение работ группы аналитиков. Шаблон анкеты для интервьюирования представлен в таблице 2.

Таблица 2

Карта тем для интервью

№п/п	Тема	Вопросы	Примечание
1	Информация о проекте	1)Какова основная предметная область? 2)Каковы основные функциональные обязанности сотрудника? 3)Какие основные проблемы существуют?	Оценить отношение к проекту

		<p>4) Как решаются данные проблемы в настоящее время?</p> <p>5) Как хотелось бы решать их в дальнейшем?</p>	
2	Понимание пользовательской среды	<p>1) Какие пользователи системы планируются?</p> <p>2) Какое образование и навыки работы с компьютером?</p> <p>3) Имеют ли пользователи опыт работы с таким типом приложений?</p> <p>4) Какие платформы используются?</p> <p>5) Каковы ожидания относительно будущего продукта?</p> <p>6) Каковы планы на будущее относительно платформ?</p> <p>7) Используется ли ПО для решения проблемы на текущий момент? Каков его функционал?</p> <p>8) Необходимо ли обучение пользователей?</p> <p>9) Необходимы ли справочные руководства, и в каком виде?</p>	Степень заинтересованности, вовлеченности
3	Описание бизнес-процесса	<p>1) Пожалуйста, опишите основной алгоритм проведения проверки.</p> <p>2) Опишите процесс внесения данных о полученных результатах. Какие данные вносятся?</p> <p>3) Каким образом следует осуществлять их хранение? Необходимо ли шифрование?</p> <p>4) Каким образом высчитывается острота зрения?</p> <p>5) Какой вид у списка с результатами и пациентами?</p>	
4	Проблемное поле	<p>1) Часто ли бывают ошибки в данных о пациентах и результатах? Если да, то какие?</p> <p>2) Сколько времени уходит на обработку данных одного пациента?</p> <p>3) Сколько времени уходит на внесение результатов и</p>	

Помощь с ВКР под ключ!
пишем-вкр.сами.рф

		оформление документации? 4) С какими еще проблемами вы сталкиваетесь?	
5	Готовность и предложения	1) Пользуетесь ли вы в данный момент каким-либо ПО для работы в данном направлении? Каким? Что вас в нем устраивает/не устраивает? 2) Хотите что-либо улучшить или изменить в процедурах работы с пациентами?	Готовность изменить привычки, переучиться
6	Подведение итогов	1) Подскажите, пожалуйста, с кем еще нам необходимо встретиться для получения дополнительной информации? 2) Можно ли будет обратиться к вам в будущем для получения дополнительной информации?	

Интервью проводилось с сотрудником диагностического отделения.
Результаты интервью приведены в таблице 3.

Помощь с ВКР под ключ!

пишем-вкр-сами.рф

Таблица 3

Ответы на интервью

№ п/п	Вопрос	Ответ	Примечание
1.	Какова основная предметная область?	Диагностическое отделение, по работе с пациентами, занимается первичной проверкой зрения, а также проводит различные тренировки с использованием тренажеров и тестов	Посмотреть работу диагностического отделения для понимания алгоритмов
2.	Каковы основные функциональные обязанности сотрудника?	Основные функциональные обязанности изложены в документированной процедуре.	Посмотреть документированную процедуру.
3.	Какие основные проблемы существуют?	Некоторые пациенты запоминают таблицу, что приносит много проблем в дальнейшем	
4.	Используется ли какое-нибудь ПО для решения данных проблем?	В настоящее время не используется никакое ПО	
5.	Как хотелось бы решать их в дальнейшем?	Хотелось бы удобную для работы программу.	
6.	Какие пользователи	Планируется врач	

	системы планируются?	диагностического отделения т.к. данной системой будут пользоваться только они. Также можно добавить кого-нибудь кто будет добавлять пользователей.	
7.	Какое образование и навыки работы с компьютером?	Сотрудники имеют высшее образование, и хороший опыт работы с компьютером.	
8.	Имеют ли пользователи опыт работы с таким типом приложений?	Да, у нас есть похожие приложения.	
9.	Какие платформы используются?	Используются такие платформы как: ОС Windows, браузер, почтовый клиент, офисный пакет	Посмотреть компьютер сотрудника
10.	Каковы планы на будущее относительно платформ?	Пока изменение платформ не планируется	
11.	Необходимо ли обучение пользователей?	Думаю, да, ознакомительное занятие не помешает.	
12.	Необходимы ли справочные руководства, и в каком виде?	Скорее всего, да, на всякий случай. В электронном виде.	
13.	Пожалуйста, опишите основной алгоритм проведения проверки	Пациент садится на расстоянии примерно 6 метров от таблицы и говорит ту букву на которую указывает проверяющий. Если буква названа верно, то показывают на следующей строке, если неверно, то на ту же или с более большими буквами	
14.	Опишите процесс внесения данных о полученных результатах. Какие данные вносятся?	В список пациентов вводится ФИО, адрес, полис, дата рождения и телефон. В результаты вводится острота зрения, она обозначается на таблице буквой V, также вводится ФИО пациента и дата проверки	
15.	Каким образом следует осуществлять их хранение, и нужно ли шифрование?	Хранение желательно осуществлять в табличном виде. Шифрование не мешает	
16.	Какие параметры при проверке являются самыми важными?	Наиболее важными являются верное соотношения размера картинок и расстояния.	Выявить соотношение размера и расстояния

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

17.	Какой вид у списка с результатами и пациентами?	Результаты записываются в карточку пациента, строкой V OD = 0.6	
18.	Что означает строка V OD = 0,6?	V- это номер строки, в приведенном примере это строка 0.6, OD- правый глаз OS- левый глаз	
19.	Часто ли бывают ошибки в данных о пациентах или результатах? Если да, то какие.	Иногда неверно вводятся данные о пациенте, например фамилия	
20.	Сколько времени уходит на внесение результатов и оформление документации?	На внесение всех данных уходит примерно 10 минут	
21.	С какими еще проблемами вы сталкиваетесь?	Иногда пациенты заранее заучивают таблицу	
22.	Подскажите, пожалуйста, с кем еще нам необходимо встретиться для получения дополнительной информации?	Можете еще встретиться с Еленой Сергеевной	
23.	Можно ли будет обратиться к Вам в будущем для получения дополнительной информации?	Да, конечно.	

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

В результате анкетирования и интервьюирования заинтересованных лиц были сформулированы потребности заказчика относительно разрабатываемого ПП. Далее необходимо провести аналогию между выявленными потребностями структурой и требованиями ТЗ в соответствии с ГОСТ. Таким образом, потребности заказчика в ТЗ могут быть описаны в разделе «Назначение и цели создания системы».

В процессе анализа зоны проектирования и предметной области были выявлены следующие потребности:

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций:

- начальное формирование данных о пациентах в виде списка (ввод с

клавиатуры).

- начальное формирование данных о врачах в виде списка (ввод с клавиатуры).
- начальное формирование данных о результатах в виде списка (ввод с клавиатуры).

Автоматизация осуществляемых на предприятии процессов позволит хранить информацию в одной базе, информация в которую вводится с помощью удобного интерфейса.

Создание собственной автоматизированной системы позволит учесть все особенности учета на предприятии, то есть разработать только то, что нужно для данного предприятия.

Программный продукт должен выполнять следующие функции/соответствовать следующим функциональным требованиям:

- Разделение доступа пользователей к информации;
- Учет информации про пациентов (ФИО, дата рождения, адрес, полис, номер телефона);
- Возможность переключения языков (русский и английский);
- Учет информации про врачей (ФИО, номер телефона, время работы);
- Соблюдение стандартов таблицы для проверки остроты зрения (размеры картинок, острота V и расстояние D);
- Возможность переключать наборы картинок (буквы, круги, фигуры).

2.2 Анализ входных и выходных данных

Входными данными разрабатываемой информационной системы можно считать:

- информация о враче: ФИО, телефон, время работы;
- информация о пациенте: ФИО, дата рождения, адрес, полис, телефон;

Указанную информацию можно назвать основной входной информацией. В процессе работы программы, от пользователя, для работы некоторых функций, могут быть запрошены дополнительные данные: примечания и др.

К выходным данным информационной системы можно отнести составленные по запросу пользователя отчеты, а также представление имеющейся ранее входной информации в базе данных:

- результаты проверки;
- результаты внесенных изменений.

2.3 Разработка тестовых сценариев, программы и методики испытаний

Помощь с ВКР под ключ!

Цель тестирования

Целью тестирования проекта является проверка всех функциональных возможностей программного продукта и совместимости с различными операционными системами, проведение серии стресс-тестов для выявления уязвимостей и поиска багов решения [50].

План тестирования

Функциональное тестирование

Цель: Выявление функциональных ошибок, несоответствий ТЗ и ожиданиям пользователя путем имитации стандартных действий пользователя.

Тестирование в определенной среде

Цель: проверить работу всех модулей программного продукта и корректность отображения форм в различных операционных системах и при различных разрешениях монитора.

Стресс-тестирование

Цель: выявить уязвимости программного продукта посредством имитации чрезмерной нагрузки на информационную систему, ввода данных большой длины и некорректных данных.

Этапы:

- Авторизация пользователя;
- Добавление нового пользователя;
- Удаление пользователя;
- Поиск.

План тестирования для последующего процесса тестирования представлен в таблицах 4, 5, 6.

Таблица 4

Функциональное тестирование

Функция	Действие	Ожидаемый результат	Полученный результат	Фатальные ошибки
Авторизация пользователя	Ввод верных данных для входа в поля Логин и Пароль	Успешный переход на страницу главную страницу	Успешный переход на страницу главную страницу	Отсутствуют
Меню	Выбор страницы врачи	Успешный переход на страницу врачи	Успешный переход на страницу врачи	Отсутствуют
Добавление	Переход на страницу добавления, где вводятся необходимые данные	Добавление введенных данных в базу данных и	Добавление введенных данных в базу данных	Отсутствуют
Удаление	Нажатие кнопки «Удалить», около определенной записи на странице врачи	Удаление записи из базы данных	Удаление записи из базы данных	Отсутствуют
Редактирование	Нажатие кнопки «Редактировать», около определенной записи на странице врачи	Данные в БД заменяются на новые	Данные в БД заменяются на новые	Отсутствуют
Поиск	Вводится текст в строку поиска	Обновление данных в реальном	Обновление данных в реальном	Отсутствуют

		времени согласно поисковому запросу	времени согласно поисковому запросу	
Проверка	Нажатие кнопки «Проверка»	Открытие страницы с проверкой	Открытие страницы с проверкой	Отсутствуют
Увеличить	Нажатие кнопки странице с проверкой	Картинка сменяется случайной из набора и становятся больше	Картинка сменяется случайной из набора и становятся больше	Отсутствуют

Таблица 5

Тестирование в определенной среде

Система	Разрешение экрана	Ошибки/несоответствия
Windows 8	1366x768	Нет
Windows 7	1280x1024	Нет
Windows 10	1280x1024	Нет
Windows 8	1440x900	Нет

Помощь с ВКР под ключ!

Таблица 6

Стресс-тестирование:

Функция	Действие	Ожидаемый результат	Полученный результат	Фатальные ошибки
1 Авторизация пользователя	Ввод некорректных данных для входа в поле «Логин»	Появляется сообщение об ошибке и пользователь остается на странице авторизации	Появляется сообщение об ошибке и пользователь остается на странице авторизации	Отсутствуют
2 Авторизация пользователя	Ввод некорректных данных для входа в поле «Пароль»	Появляется сообщение об ошибке и пользователь остается на странице авторизации	Появляется сообщение об ошибке и пользователь остается на странице авторизации	Отсутствуют
3 Подключение к базе данных.	Ввод некорректных данных в поле «пользователь»	Появляется сообщение об ошибке и пользователь остается на странице проверки	Появляется сообщение об ошибке и пользователь остается на странице проверки	Отсутствуют

		подключения	подключения	
3 Подключение к базе данных.	Ввод некорректных данных в поле «База-данных»	Появляется сообщение об ошибке и пользователь остается на странице проверки подключения	Появляется сообщение об ошибке и пользователь остается на странице проверки подключения	Отсутствуют

2.4 Разработка технического задания программного продукта

Введение

Данное техническое задание предназначено на разработку автоматизированной информационной системы для проверки зрения ГБУЗ СОКОБ им Т.И. Ерошевского, предназначенной для использования сотрудниками диагностического отделения [4].

Наименование

Разработка информационной системы для проверки зрения ГБУЗ СОКОБ им Т.И. Ерошевского.

Краткая характеристика области применения программы

Область применения данной системы – диагностический кабинет. Данная разработка программного продукта предназначена для автоматизации проверки на остроту зрения. Программа будет работать с базой данных пациент, считывая оттуда различные данные, необходимые для работы автоматизированной информационной системы.

Основание для разработки

Основанием для проведения разработки является установленная тема ВКР разработка программного продукта для проверки зрения ГБУЗ СОКОБ им Т.И. Ерошевского.

Общие положения

1. Назначение документа

- В настоящем документе приводится набор требований к ИС, необходимых для ее правильной реализации и работы.
- При реализации необходимо выполнить работы в объеме, настоящем Техническом задании.
- Все неоднозначности, выявленные в настоящем Техническом задании после его согласования и подписания, подлежат Сторонами.

2. Цели создания

- Сформировать базу пациентов и врачей для учета результатов проверки зрения.
- Создать программный продукт, реализующие следующие функции:
 - настройка удобной работы сотрудников диагностического отделения, а также ее упрощение;
 - упрощение процесса проведения проверки;
 - упрощение и ускорение процесса внесения результатов проверки;
 - повышение комфортности просмотра списка пациентов для пользователей системы;

3. Основные функциональные возможности

- Ведение списка пациентов:
 - внесение пациентов в базу данных;
 - редактирование данных пациентов;
 - удаление данных;
- Ведение списка врачей:
 - внесение врачей в базу данных;
 - редактирование данных врачей;
 - удаление данных;

4. Просмотр данных:

- просмотр и сортировка списка пациентов;

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

- просмотр и сортировка списка врачей
- просмотр и сортировка результатов;
- Проведение проверки:
 - смена картинок из набора представленных;
 - верное соотношение параметров проверки;

5. Пользователи системы

Основными пользователями системы являются:

- Врач – имеет полный доступ к системе, с возможностью просмотра, добавления, удаления и редактирования врачей, пациентов и результатов, а также имеет возможность проводить проверку;
- Администратор – Имеет доступ к тем же функциям но при этом добавляет новых пользователей и подключение

Требования к составу выполняемых функций

Разрабатываемый программный продукт должен обеспечивать возможность выполнения перечисленных ниже функций:

1. Ведение списка пациентов:

- внесение пациентов в базу данных;
- редактирование данных пациентов;
- удаление данных.

2. Ведение списка врачей:

- внесение врачей в базу данных;
- редактирование данных врачей;
- удаление данных.

3. Просмотр данных:

- просмотр и поиск списка пациентов;
- просмотр и поиск списка врачей;
- просмотр и поиск результатов.

4. Проведение проверки:

- смена картинок из набора представленных;

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

- верное соотношение параметров проверки (размера картинок и надписей).

Требования к организации входных данных

Входные данные программы должны быть организованы в виде вводимого в специальную форму текста или файла, соответствующего определенному шаблону.

Файлы указанного формата должны размещаться (храниться) на локальных или съемных носителях, отформатированных согласно требованиям операционной системы.

Требования к организации выходных данных

Выходные данные программы должны быть организованы в виде таблиц.

Файлы указанного формата должны храниться на локальных или съемных носителях, отформатированных операционной системы.

Отчеты формируются в режиме реального времени и передаются пользователю в формате «.docx». При желании любой отчет можно сохранить отдельно.

Описание требований к системе

1 Требования к системе в целом

Система включает в себя 2 уровня доступа. Каждый пользователь должен функционировать в рамках своего информационного подпространства.

Для получения доступа в пространство персонифицированного доступа, пользователь должен пройти процедуру аутентификации. В случае успешной аутентификации пользователя все его действия в системе будут однозначно ассоциироваться с его учетной записью.

Процедура аутентификации пользователя системы должна выполняться при помощи логина и пароля пользователя.

2 Требования к информационному обеспечению системы

Состав, структура и способы организации данных в системе должны быть определены на этапе технического проектирования.

Уровень хранения данных в системе должен быть построен на основе современных реляционных или объектно-реляционных СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

Средства СУБД, а также средства используемых операционных систем должны обеспечивать документирование и протоколирование обрабатываемой в системе информации.

Структура базы данных должна поддерживать кодирование хранимой и обрабатываемой информации в соответствии с общероссийскими классификаторами (там, где они применимы).

Доступ к данным должен быть предоставлен только авторизованным пользователям с учетом их служебных полномочий, а также с учетом категории защищаемой информации.

Структура базы данных должна быть организована рациональным способом, обеспечивающим эффективную полную выгрузку информации, содержащейся в базе данных системы.

Технические средства, обеспечивающие хранение информации, должны использовать современные технологии, позволяющие обеспечить повышенную надежность хранения данных и оперативную замену оборудования (распределенная избыточная запись/считывание данных; зеркалирование; независимые дисковые массивы; кластеризация).

В состав системы должна входить специализированная подсистема резервного копирования и восстановления данных.

Данные должны быть защищены от разрушений при авариях и сбоях в электропитании системы путем создания резервных копий.

При проектировании и развертывании системы необходимо рассмотреть возможность использования накопленной информации из уже

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

функционирующих информационных систем.

3 Требования к лингвистическому обеспечению системы

Все обозначения, названия элементов управления Системы, тексты должны быть изложены на русском языке без применения терминов, непонятных пользователю.

Разработка системы должна вестись на языке программирования высокого уровня PHP.

4 Требования к программному обеспечению системы

Системные программные средства свободно распространяемая операционная система Windows.

Программное обеспечение, распространяемое свободно:

- База данных (СУБД) MySQL Workbench;
- Apache HTTP Server версии 2.2.16 (или выше);
- PHP версии 5.1 (или выше);
- Система управления информацией с открытым кодом;
- ОС Windows 10.

5 Требования к техническому обеспечению системы

Система должна функционировать на ПК, удовлетворяющих следующим требованиям:

1. Тип процессора: процессор типа Intel Core i3 (или эквивалент);
2. Базовая тактовая частота процессора: минимум: 3,6 ГГц;
3. Оперативная память: минимум: 3ГБ;
4. Дисковое пространство: минимум: 40 ГБ;
5. Источник бесперебойного питания - Обеспечивает работоспособность в условиях кратковременного сбоя электроснабжения;
6. Принтер лазерный сетевой - Формат А4 Быстродействие не ниже 25 стр./мин, разрешение не ниже 600×600 dpi;
7. Сканер - Оптическое разрешение не менее 1200×2400 dpi;
8. Цифровая фотокамера;

9. Расходные материалы – бумага А4, картриджи и тонер для лазерного принтера;
10. Операционная система: Windows 7 или более поздних версий;
11. Файловый менеджер (в составе операционной системы или др.);
12. Браузеры, один из предложенных ниже:
 - Internet Explorer версии 10 или более поздней;
 - Mozilla Firefox версии 5 или более поздней;
 - Google Chrome версии 13 или более поздней;
 - Opera версии 11 или более поздней;
13. Почтовый клиент (входит в состав операционных систем или др.);
14. Антивирусная программа (клиентская лицензия);
15. Комплект офисных программ, включающий:
 - текстовый редактор,
 - программу разработки презентаций,
 - электронные таблицы;

Помощь с ВКР под ключ! пишем-вкр-сами.рф

16. Программа для просмотра статических изображений;
Внутренняя сеть и средства коммуникации должны обладать как минимум следующими характеристиками:

1. скорость передачи данных подключаемого канала к публичным сетям не менее 2 Мб/с;
2. оборудование узла должно оставаться работоспособным при кратковременных отключениях электропитания (на время не менее 10 минут);
3. оборудование узла должно обеспечивать коммутируемое подключение всех устройств со скоростью до 100 Мбит/с.

6 Требования к организационному обеспечению системы

Организационное обеспечение системы должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей при осуществлении автоматизированных и связанных с ними

неавтоматизированных функций системы.

Заказчиком должны быть определены структура и функция подразделений, участвующих в функционировании системы, должностные лица, ответственные за:

- обработку информации АС;
- администрирование АС;
- обеспечение безопасности информации АС;
- управление работой персонала по обслуживанию АС.

К работе с системой должны допускаться сотрудники, имеющие навыки работы на персональном компьютере, ознакомленные с правилами эксплуатации и прошедшие обучение работе с системой.

2.5 Выбор программного обеспечения для реализации (разработки) программного продукта

Помощь с ВКР под ключ! пишем-вкр-сами.рф

Выбирая интегрированную среду разработки для реализации информационной системы, сравнивались три системы Visual Studio, Project Rider, Eclipse.

- Project Rider - не самый лучший вариант, тк. часть функций все еще находится в разработке и среда разработки содержит много багов, в добавок к этому является платным приложением.
- Eclipse - хоть это и бесплатная среда разработки, но она работает очень нестабильно, из-за чего часто вылетает и имеет много глюков.
- Visual Studio- является наиболее подходящей тк она является бесплатной и содержит большое количество различных инструментов и дополнений которые могут помочь в разработке.

Результаты сравнения языков программирования представлены ниже:

- Python - хоть он и является кроссплатформенным языком, и легким для

понятия языком, Python долго загружает и запускает программу, а также выдает ошибки при работе с русскими буквами что является огромным минусом.

- C++ - также как и стальные языки является кроссплатформенным, но имеет сложности при поиске ошибок, ведь их придется искать вручную, также данный язык имеет громоздкий синтаксис и запутанный синтаксис, из-за этого может увеличиться время разработки
- C# - кроссплатформенный язык постоянно совершенствуется, имеет огромное количество дополнительных библиотек и плагинов, также быстро загружает проект.

На основании вышесказанного был выбран язык программирования C# и система Visual studio для разработки информационной системы [37].

Помощь с ВКР под ключ!

2.3 Выбор программного обеспечения для разработки и эксплуатации удаленных базы данных

пишем-вкр-сами.рф

Выбирая систему управления базами данных (далее – СУБД) для разработки базы данных сравнивались три системы: SQLite, MySQL, SQL. Результаты представлены ниже:

- SQL - имеет возможность отмены запроса, но не является кроссплатформенной СУБД, является платной и может использовать только один механизм хранения.
- SQLite - является кроссплатформенной, бесплатной и маловесящей СУБД, но не подходит для создания многопользовательских приложений, поддерживает ограниченное количество типов данных, также имеет ограничение на количество записанных данных из-за чего при создании больших СУБД будут ошибки.

- MySQL - тоже является кроссплатформенной и бесплатной СУБД , в отличии от SQLite подходит для создания многопользовательской базы данных, поддерживает большое количество типов данных, может обрабатывать большой объем данных в кратчайшие сроки, также имеет большой функционал.

В результате сравнения, наиболее подходящим вариантом является СУБД MySQL [31].

Также одним из наиболее важных факторов была возможность СУБД поддерживать Entity Framework. Entity ускоряет и упрощает разработку программного продукта, тк база создается и редактируется при помощи кода без использования sql запросов, что позволяет избежать лишних проблем.

2.7 Разработка и оптимизация схемы данных программного продукта

Помощь с ВКР под ключ! пишем-вкр-сами.рф

Для хранения данных автоматизированной информационной системы были выбраны реляционные базы данных. Для создания базы данных и правильной работы ее в последующем необходимо разработать схему данных.

В процессе разработки схемы данных необходимо произвести нормализацию, структуры базы данных. Этот процесс подразумевает под собой оптимизацию таблиц базы данных таким образом, чтобы в таблицах отсутствовали, как минимум, повторяющиеся данные.

По результатам анализа предметной области необходимо создать ER-диаграмму по которой создаются таблицы в базе данных а также их взаимодействие.

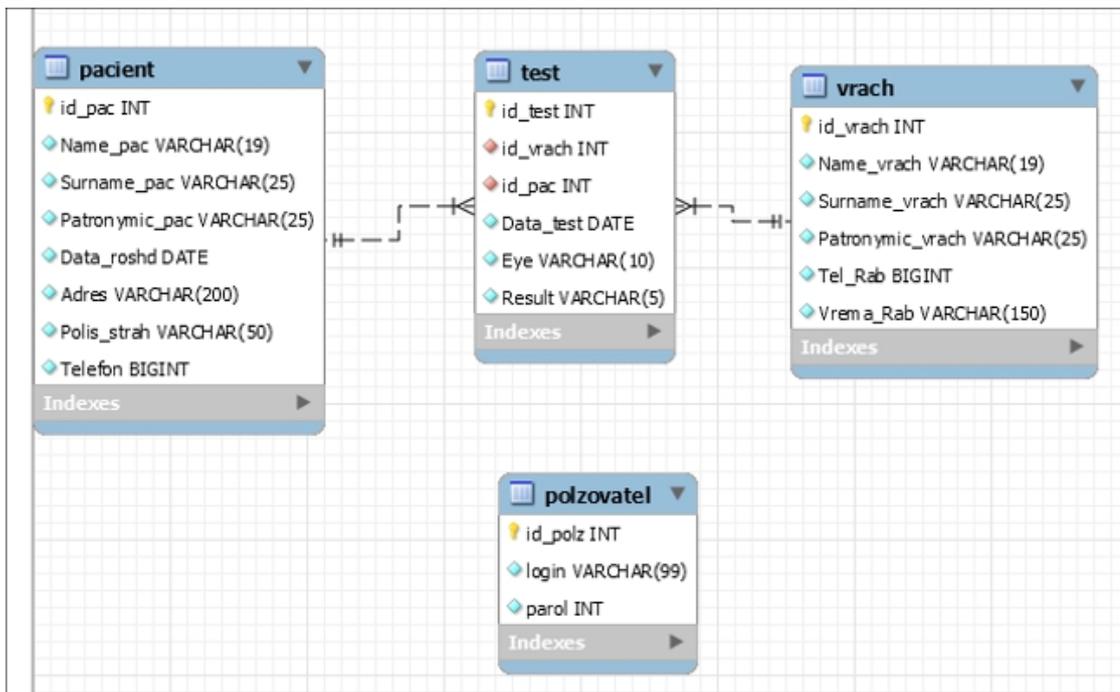


Рисунок 3 - ER диаграмма

Также ключевые таблицы должны иметь ключевое поле (первичный ключ), со свойством автозаполнения. А таблицы, которые позволяют построить связь, должны иметь несколько ключевых полей. Это необходимо для успешного построения связей между таблицами для их дальнейшего взаимодействия.

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

ГЛАВА 3 ПРОЕКТНАЯ ЧАСТЬ

3.1 Разработка интерфейса программного продукта

Программная часть АИС работает на базе WPF – часть экосистемы платформы .NET и представляет собой подсистему для построения графических интерфейсов. Все страницы программы взаимосвязаны между собой: это позволяет пользователю переходить с одной рабочей страницы на другую, а в самой программе появляется возможность работы с данными, созданными на одной странице, в другой странице. Также к особенностям можно отнести использования языка декларативной разметки интерфейса XAML, основанного на XML, благодаря чему появляется возможность создать оригинальный, удобный и понятный графический интерфейс.

При загрузке программы пользователю доступна страница авторизации, которая изображена на рисунке 4.



Авторизация

Логин:	<input type="text"/>
Пароль:	<input type="password"/>
	<input type="checkbox"/> Показать пароль

Вход

© ГБУЗ СОКОБ им.Т.И. Ерошевского.

Рисунок 4 - Окно авторизации

После прохождения авторизации доступна возможность проверки и добавления подключения (рисунок 5). Данная функция доступна только для администратора

База данных:	<input type="text" value="ISPr21-33_AsadullinaUR_Diplom"/>
Сервер:	<input type="text" value="cfif31.ru"/>
Пользователь:	<input type="text" value="ISPr21-33_AsadullinaUR"/>
Пароль:	<input type="text" value="ISPr21-33_AsadullinaUR"/>

© ГБУЗ СОКОБ им.Т.И. Ерошевского

Рисунок 5 - Окно проверки подключения

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

Также после авторизации и настройки появятся кнопки навигации, при помощи которых можно выбрать необходимую страницу (по умолчанию включается страница проверки). Интерфейс страницы изображен на рисунке 6.

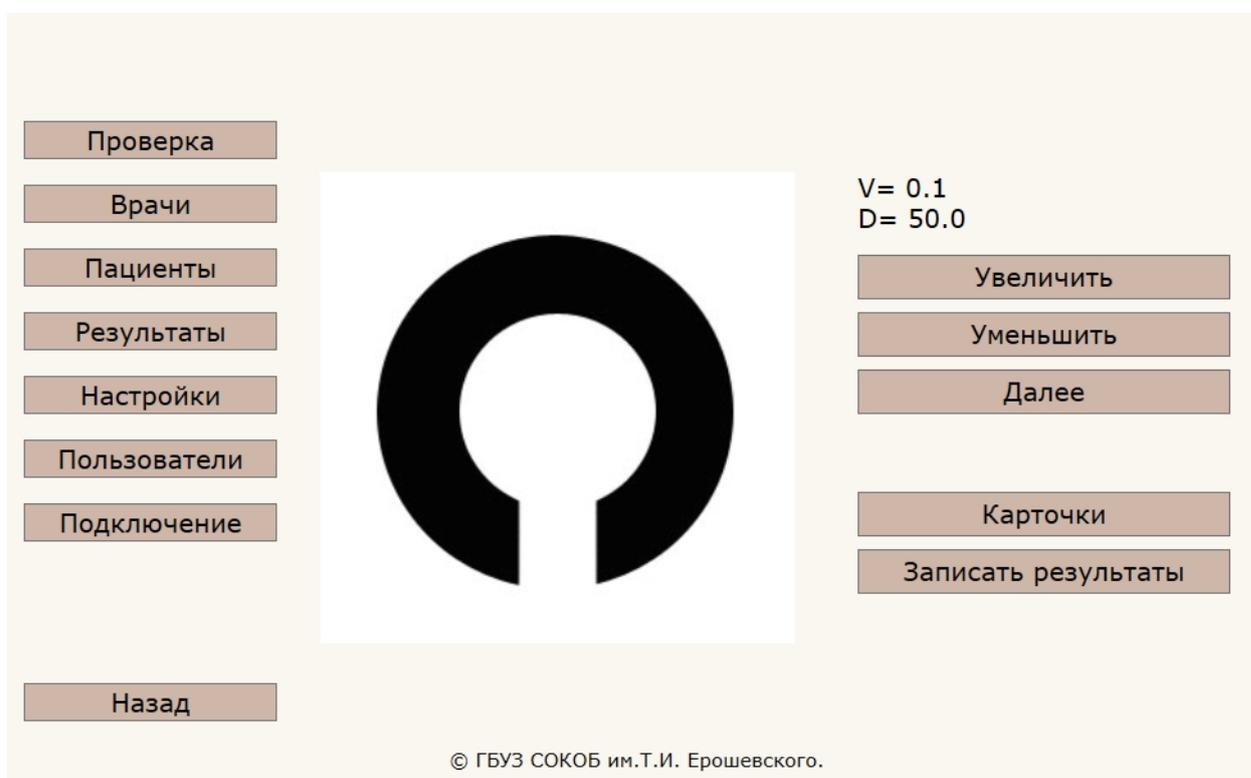


Рисунок 6 - Главное меню

Помощь с ВКР под ключ!

При нажатии кнопок “Врачи”, “Пациенты”, “Результаты” открываются страницы с данными и возможностью работы с ними (удаление, редактирование, добавление и поиск).

пишем-вкр-сами.рф

Данные формы имеют похожий интерфейс который отличается только различными полями.

Например, интерфейс формы врачи представлен на рисунке 7.

Поиск:

<p>Лимонов Виктор Николаевич Телефон: 8956382757 График работы: Будни 9-17</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>
<p>Венков Дмитрий Викторович Телефон: 8954654664 График работы: Будни 8- 16</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>
<p>Пирогов Николай Олегович Телефон: 89572666385 График работы: Будни 8-17</p> <p><input type="button" value="Добавить"/></p>

Рисунок 7 - Страница врачи

При нажатии кнопки “Добавить” и “Редактировать” появится

Помощь с ВКР под ключ!
 специальное окно для ввода данных.
 Оно также имеет похожий интерфейс, который отличается
 количеством некоторых элементов.
пишем-вкр-сами.рф
 Интерфейс формы представлен на рисунке 8.

Имя:	<input type="text"/>
Фамилия:	<input type="text"/>
Отчество:	<input type="text"/>
Телефон:	<input type="text" value="0"/>
График работы:	<input type="text"/>
<input type="button" value="Сохранить"/>	

Рисунок 8 - Страница добавления (редактирования)

При нажатии кнопки “проверка” в главном меню (или по умолчанию),

откроется форма проверки, с возможностью увеличить, уменьшить картинку или просто поменять ее, не меняя размер (кнопка далее).

Примерный интерфейс представлен на рисунке 9.



Помощь с ВКР под ключ!

Если же необходимо сменить набор картинок, например если пациент еще маленький или не знает буквы, то при нажатии кнопки «Карточки» откроется страница, где можно выбрать нужный вариант из представленных.

Интерфейс данной формы представлен на рисунке 10.

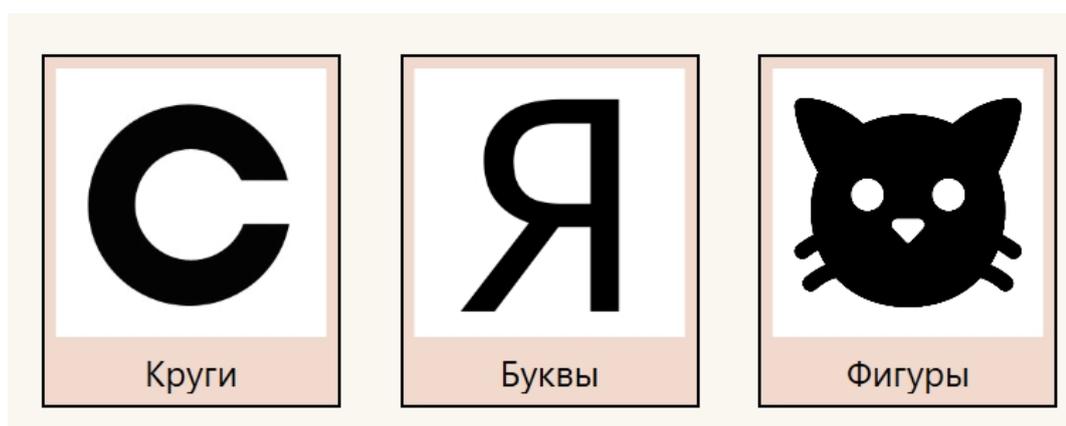


Рисунок 10 - Страница смены карточек

Также имеется страница с разнообразными настройками (смена языка,

смена цветовой гаммы).

Примерный интерфейс представлен на рисунке 11.



Рисунок 11 - Страница настроек

Также в программе имеется возможность включить режим для слабовидящих, в этом режиме цветовая гамма меняется на более контрастную и увеличивается шрифт и размер некоторых кнопок. Сравнение представлено на рисунке 12 -13.

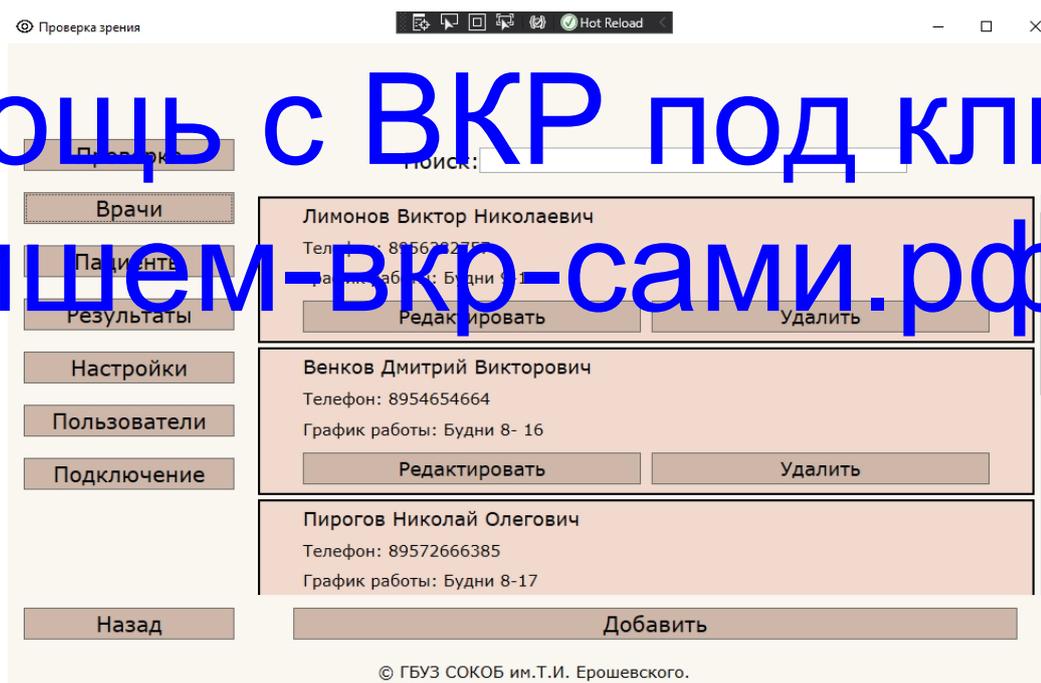


Рисунок 12 — Страница врачи в обычном режиме

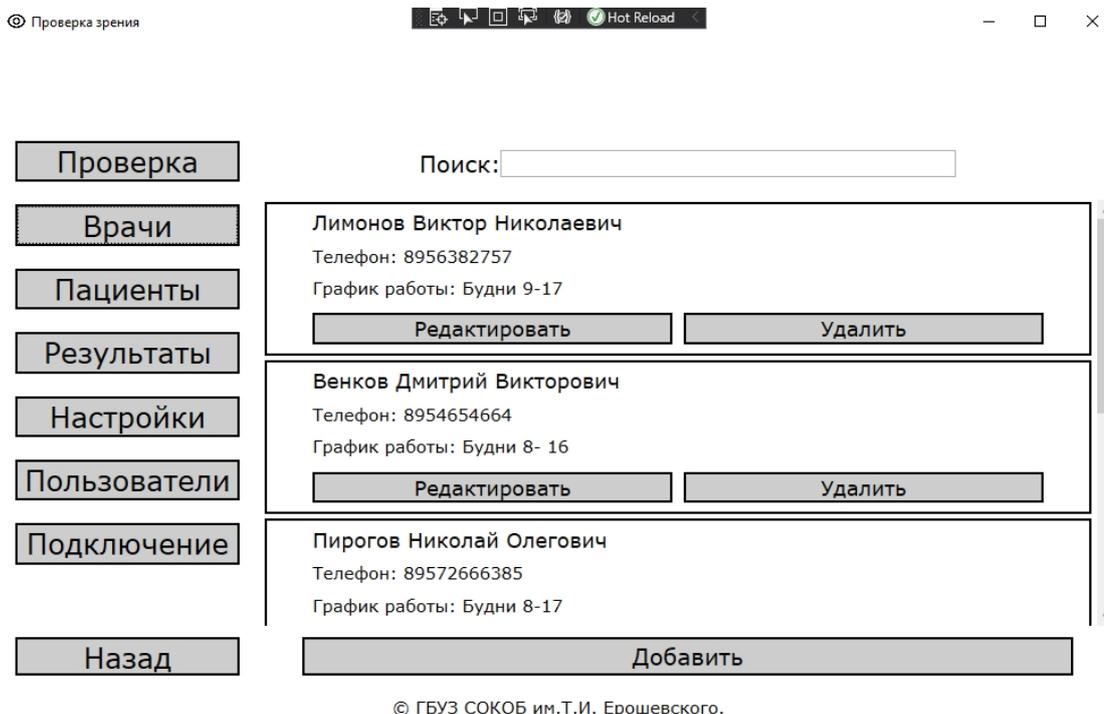


Рисунок 13 — Страница врачи в специальном режиме

Помощь с ВКР под ключ!

пишем-вкр-сами.рф

Для начала необходимо запустить MySQL Workbench, нажав пуск щелчком мыши по папке «MySQL». После чего нужно добавить подключение заполнив Hostname, Port, Username, Password.

Также для корректной работы с программой необходимо подключить библиотеку MySql.Data.EntityFramework,

В появившейся странице, в левом меню нажав правую кнопку мыши «Create Schema», введите имя создаваемой базы – «Врач».

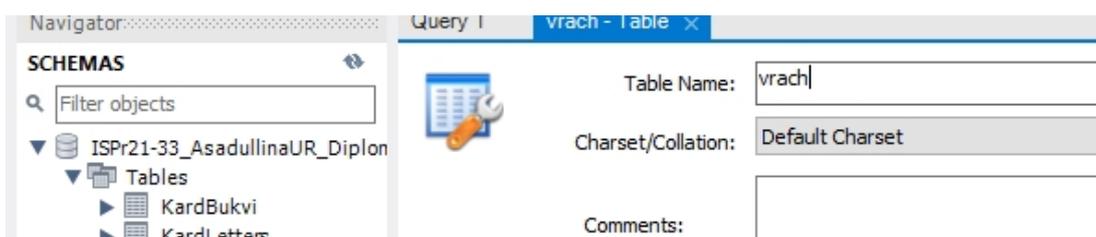


Рисунок 14 – Создание базы данных «Врач»

После чего нужно создать сущности/таблицы, базы данных, описывая ее название, и сколько выделяете характеристик, для этой сущности – «users», 6 – столбцов.

Следующим действием, будет описание характеристик/свойств, и указание их типов в соответствующих диапазонах.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
id_vrach	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
Name_vrach	VARCHAR(19)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Surname_vrach	VARCHAR(25)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Patronymic_vrach	VARCHAR(25)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Tel_Rab	BIGINT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Vrema_Rab	VARCHAR(150)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				

Рисунок 15 – Создание таблиц и описание свойств

Теперь в базе данных имеется пустая таблица «vrach», которую нужно заполнить, для этого в левом меню выберите таблицу «vrach» и перед вами

Помощь с ВКР под ключ!
 пишем-вкр-сами.рф

появится страница для заполнения данных полей

id_vrach	Name_vrach	Surname_vrach	Patronymic_vrach	Tel_Rab	Vrema_Rab
1	Виктор	Пименов	Николаевич	89538075	Будни 9-17
2	Дмитрий	Венков	Викторович	8954654664	Будни 8-16
3	Николай	Пирогов	Олегович	89572666385	Будни 8-17
4	Александр	Полянский	Сергеевич	89645645644	Будни 17-9
5	Ольга	Поваляева	Николаевна	89634524554	Будни 17-9
+	NULL	NULL	NULL	NULL	NULL

Рисунок 16 – Заполнение таблицы

3.3 Реализация функциональных требований

При включении программы пользователя встречает окно авторизации, при успешно пройденной авторизации пользователь попадает на главное меню, в зависимости от роли функционал будет немного изменен.

При попадании на главное меню пользователь может просмотреть,

добавить, удалить или изменить любые данные на страницах «Врачи», «Пациенты», «Результаты», а также провести проверку. Кроме данных функций админ имеет доступ к добавлению пользователей и проверки подключения



Помощь с ВКР под ключ! пишем-вкр-сами.рф

Изначально администратору выводится страница с пользователями, где можно просмотреть логин, пароль и роль.

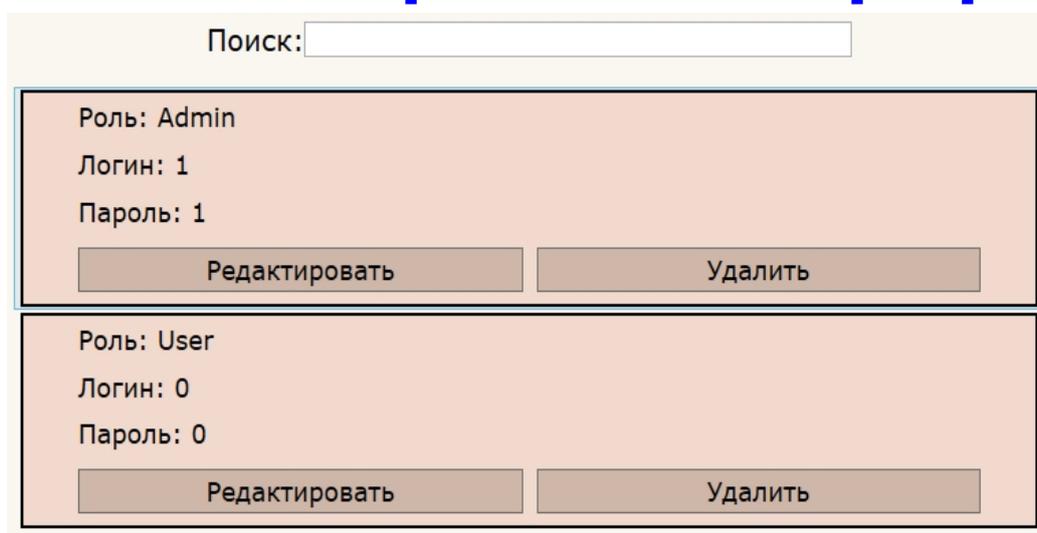
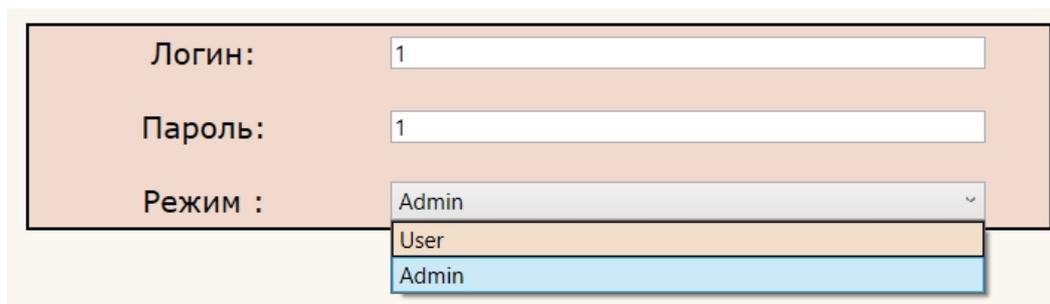


Рисунок 18 – Список пользователей

Также имеется возможность редактирования и добавления записей со

всеми необходимыми полями



Логин: 1

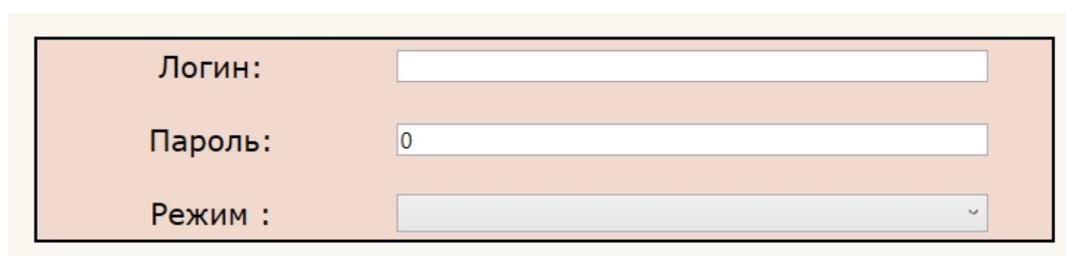
Пароль: 1

Режим : Admin

User

Admin

Рисунок 19 – Редактирование пользователей



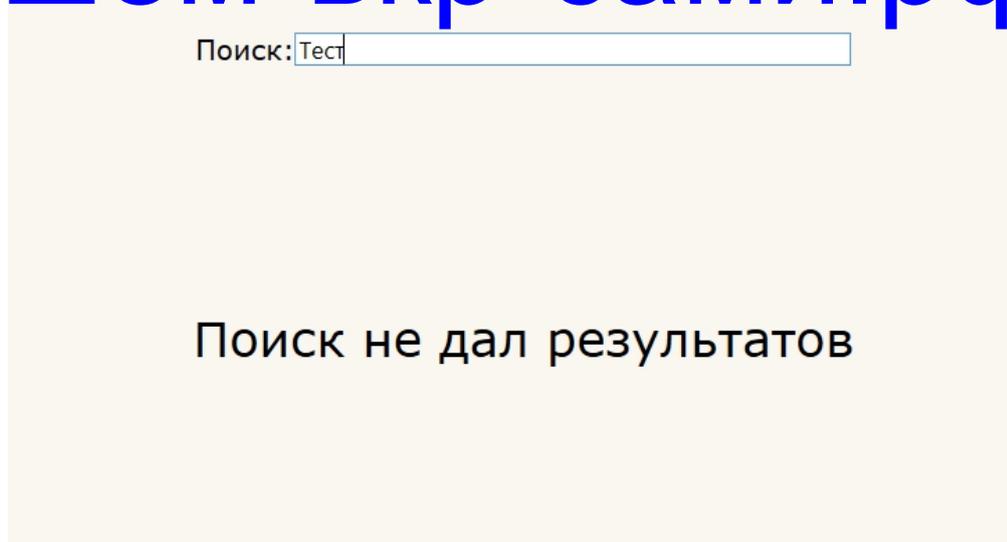
Логин:

Пароль: 0

Режим :

Рисунок 20 – Добавление пользователей

Помощь с ВКР под ключ!
На случай, если необходимо найти конкретного пользователя, присутствует поиск в реальном времени, если запись не будет найдена, появится соответствующее сообщение.
пишем-вкр-сами.рф



Поиск: Тест

Поиск не дал результатов

Рисунок 21 – Поиск

При попытке удалить запись будет запрашиваться подтверждение, если пользователь нажмет ДА, запись удалиться, если же будет нажато НЕТ, то

сообщение просто пропадет, а запись не удалится

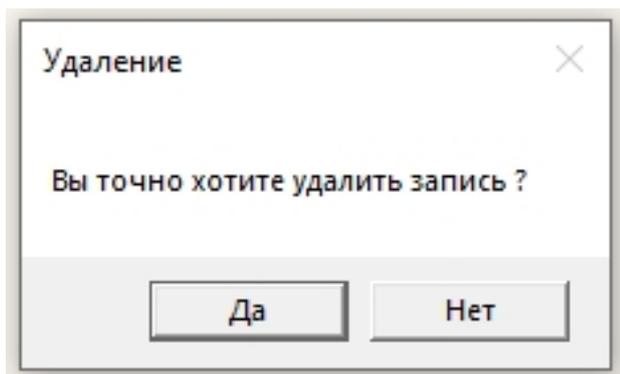


Рисунок 22 – Удаление

Список врачей выводит такие данные как ФИО, номер телефона и график работы

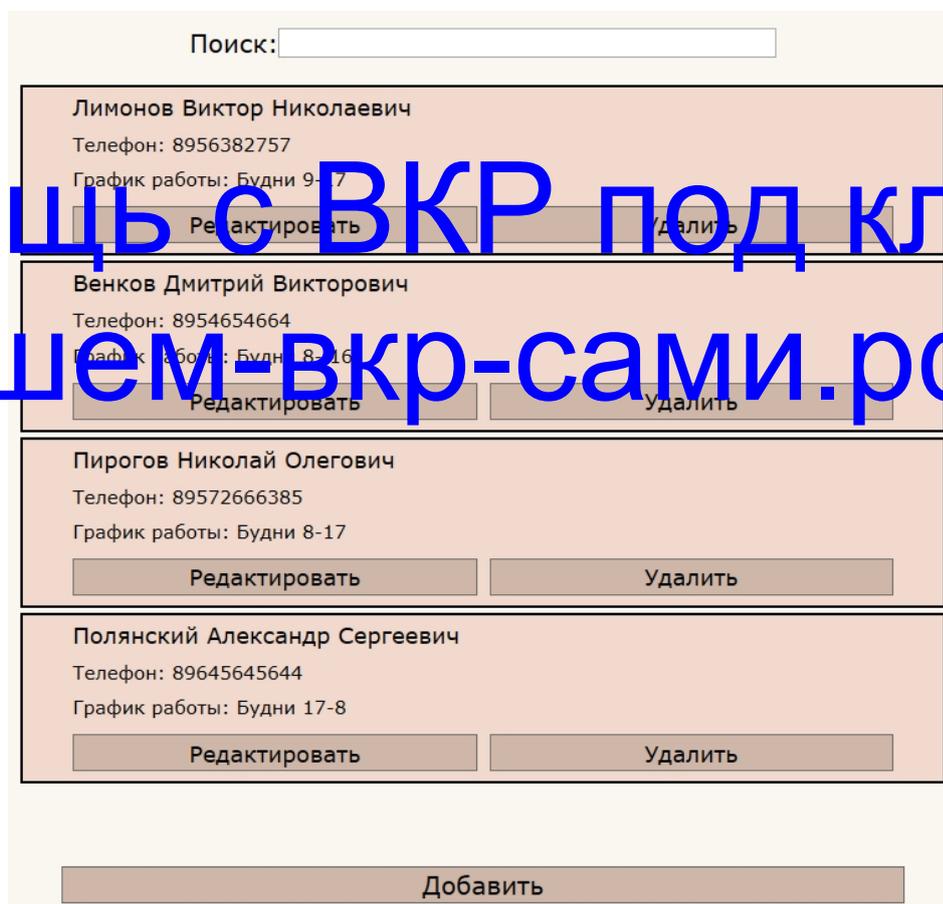


Рисунок 23 – Список врачей

При добавлении записи о врачах необходимо заполнить поля: имя, фамилия, отчество, телефон и график работы

Имя:	<input type="text"/>
Фамилия:	<input type="text"/>
Отчество:	<input type="text"/>
Телефон:	<input type="text" value="0"/>
График работы:	<input type="text"/>

Рисунок 24 – Добавление врачей

В списке пациентов хранятся данные о ФИО, телефоне, дате рождения, адресе и полисе

Поиск:

<p>Говердовская Ольга Александровна Телефон: 89456456567 Дата рождения: 27-05-1989 Адрес: ул.Кирова д 12 кв 2 Полис: 456-657-324 45</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>
<p>Полянский Андрей Васильев Телефон: 89456737854 Дата рождения: 01-10-1980 Адрес: ул. Челюскинцев д 17 кв 3 Полис: 688-079-385 77</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>
<p>jhi kjkgk fytf Телефон: 0 Дата рождения: 04-05-2022 Адрес: uug Полис: jhgjf</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>

Рисунок 25 – Список пациентов

При добавлении записи о пациентах необходимо заполнить поля: имя, фамилия, отчество, телефон дата рождения, адрес и полис. Дату рождения можно записать вручную или же просто выбрать ее при помощи календаря

Имя:

Фамилия:

Отчество:

Телефон:

Дата рождения: 15

Адрес:

Полис:

Июнь 2022

Пн	Вт	Ср	Чт	Пт	Сб	Вс
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

Рисунок 26 – Добавление пациентов

В списке результатов можно увидеть данные о враче который провел проверку, пациенте которого проверили, дате проведения проверки и результате (OS, OD).

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Поиск:

<p>Врач: Венков Дмитрий Викторович Пациент: Говердовская Ольга Александровна Дата проверки: 19-08-2019 OD: 0.4 OS: 0.8</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>
<p>Врач: Пирогов Николай Олегович Пациент: Говердовская Ольга Александровна Дата проверки: 25-12-2010 OD: 0.9 OS: 0.8</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>
<p>Врач: Венков Дмитрий Викторович Пациент: Говердовская Ольга Александровна Дата проверки: 04-04-2022 OD: 0.7 OS: 0.6</p> <p><input type="button" value="Редактировать"/> <input type="button" value="Удалить"/></p>
<p><input type="button" value="Добавить"/></p>

Рисунок 27 – Список результатов

При добавлении или редактировании необходимо выбрать врача, пациента и результатах проверки на правый и левый глаз при помощи выпадающего списка. Дату также можно ввести вручную или при помощи календаря

Врач: [dropdown]
Пациент: [dropdown]
Дата проверки: 14.06.2022 [calendar icon]
Правый глаз: [dropdown]
Левый глаз: [dropdown]

0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.5
2.0

Рисунок 28 – Добавление результата

Помощь с ВКР под ключ! пишем-вкр-сами.рф

При нажатии кнопки «Настройки» появится возможность сменить язык и включить режим для слабовидящих

Язык Русский Английский
Для слабовидящих Вкл

Рисунок 29 – Окно настроек

В программе имеется возможность выбрать русский и английский язык. Локализация работает в реальном времени.

Language: Rus Eng
For the visually impaired: On

Рисунок 30 – Переключение языка

Также как и локализация, режим для слабовидящих включается автоматически при включении соответствующей настройки.

Поиск:	<input type="text"/>
Лимонов Виктор Николаевич Телефон: 8956382757 График работы: Будни 9-17	<input type="button" value="Редактировать"/> <input type="button" value="Удалить"/>
Венков Дмитрий Викторович Телефон: 8954654664 График работы: Будни 8- 16	<input type="button" value="Редактировать"/> <input type="button" value="Удалить"/>
Пирогов Николай Олегович Телефон: 89572666385 График работы: Будни 8-17	<input type="button" value="Редактировать"/> <input type="button" value="Удалить"/>
Полянский Александр Сергеевич Телефон: 89645645644 График работы: Будни 17-8	<input type="button" value="Редактировать"/> <input type="button" value="Удалить"/>
<input type="button" value="Назад"/>	<input type="button" value="Добавить"/>

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Рисунок 31 – Режим для слабовидящих

Если в программу зашел администратор, то он также может проверить или редактировать подключение к базе данных. Для этого потребуется изменить данные в полях база данных, сервер, пользователь или пароль.

База данных:	<input type="text" value="ISPr21-33_AsadullinaUR_Diplom"/>
Сервер:	<input type="text" value="cfif31.ru"/>
Пользователь:	<input type="text" value="ISPr21-33_AsadullinaUR"/>
Пароль:	<input type="text" value="ISPr21-33_AsadullinaUR"/>
<input type="button" value="Сохранить"/>	<input type="button" value="Проверить"/>

Рисунок 32 – Проверка подключения

Затем необходимо нажать на кнопку проверить, если подключение прошло успешно появится соответствующее сообщение. Также сообщение выводится если что-то введено неверно

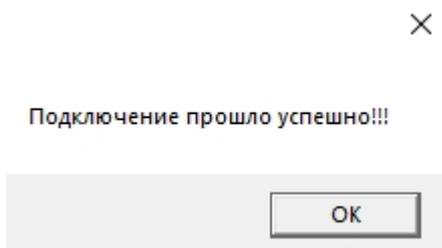


Рисунок 33 – Успешная проверка

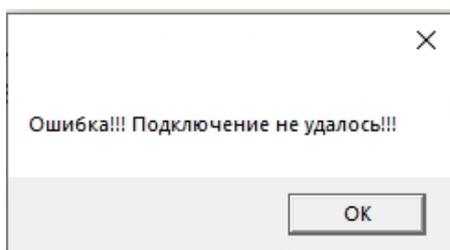


Рисунок 34 – Ошибка при подключении

Помощь с ВКР под ключ! пишем-вкр-сами.рф

На странице проверка можно сменить размер картинки (увеличить или уменьшить), сменить саму картинку (далее), записать результаты или сменить набор с картинками



Рисунок 35 – Страница проверки

При смене картинок пользователю предоставляется выбор из трех вариантов круга, буквы или фигурки.

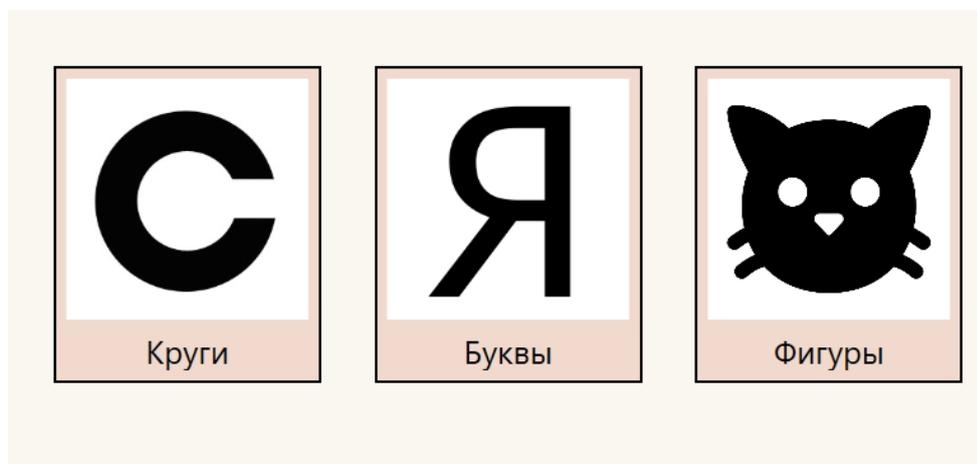


Рисунок 36 – Выбор картинок

3.4 Отладка и тестирование программного продукта

Помощь с ВКР под ключ!

Тестирование необходимо для проверки основного функционала программного продукта, например: удаление, добавление, редактирование, поиск, переключение языка или цветовой гаммы. В случае если будут найдены какие-либо ошибки, их необходимо будет исправить [50].

пишем-вкр-сами.рф

Таблица 7

Функциональное тестирование

Функция	Действие	Ожидаемый результат	Полученный результат	Фатальные ошибки
Авторизация пользователя	Ввод верных данных для входа в поля Логин и Пароль	Успешный переход на страницу главную страницу	Успешный переход на страницу главную страницу	Если у 2 пользователей одинаковый логин, программа вылетает
Авторизация пользователя	Ввод неверных данных для входа в поля Логин или Пароль	Появляется сообщение об ошибке и пользователь остается на странице авторизации	Появляется сообщение об ошибке и пользователь остается на странице авторизации	Если у 2 пользователей одинаковый логин, программа вылетает

Меню	Выбор страницы врачи	Успешный переход на страницу врачи	Успешный переход на страницу врачи	Отсутствуют
Добавлен ие	Переход на страницу добавления, где вводятся необходимые данные	Добавление введенных данных в базу данных и возвращение на страницу для просмотра данных	Добавление введенных данных в базу данных и возвращение на страницу для просмотра данных	Отсутствуют
Удаление	Нажатие кнопки «Удалить», около определенной записи на странице врачи	Появление сообщения о подтверждении удаления записи, и дальнейшее удаление записи из базы данных если пользователь нажал да	Появление сообщения о подтверждении удаления записи, и дальнейшее удаление записи из базы данных если пользователь нажал да	Отсутствуют
Редактир ование	Нажатие кнопки «Редактировать», около определенной записи на странице врачи	Данные в БД заменяются на новые	Данные в БД заменяются на новые	Отсутствуют
Поиск	Вводится текст в строку поиска	Обновление данных в реальном времени согласно поисковому запросу	Обновление данных в реальном времени согласно поисковому запросу	Отсутствуют
Проверка	Нажатие кнопки «Проверка»	Открытие страницы с проверкой	Открытие страницы с проверкой	Отсутствуют
Карточки	Нажатие кнопки «Карточки» и выбор одного из вариантов	Пользователя возвращает на форму с проверкой, где набор картинок заменяется на выбранный	Пользователя возвращает на форму с проверкой, где набор картинок заменяется на выбранный	Отсутствуют
Увеличит ь	Нажатие кнопки «Увеличить» странице с проверкой	Картинка сменяется случайной из набора и становятся больше	Картинка сменяется случайной из набора и становятся больше	Отсутствуют

ГЛАВА 4 ДОКУМЕНТАЦИОННОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММНОГО ПРОДУКТА

4.1 Документирование программного кода

Документирование кода является неотъемлемой частью при создании документационного сопровождения, но довольно часто ей уделяется недостаточно внимания.

Документирование кода – это вставка в код определенных комментариев, которые позволяют в дальнейшем упростить работу с кодом, как автору, так и другим программистам. Чтобы было понятно, как пользователю, так и программистам, что есть что, то комментировать код нужно правильно: то есть не повторять код, а описать что делает то, или иное действие, что означает эта переменная и как она используется в дальнейшем, с чем связывается. Если код имеет ссылку на библиотеку или класс, то желательно это тоже указывать. При документировании кода необходимо руководствоваться ГОСТ 19.402-78 ЕСПД «Описание программизм [2], с.233].

Полный листинг документированного кода программного продукта, представлен в приложении I.

Пример документированного кода программного продукта приведен ниже.

//Код для главного меню

```
private void Button_page_Proverka_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new
Proverka_Circl_page()); //Выводим страницу проверки
}

private void Button_page_Vrach_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new Vrach_page());
//Выводим страницу врачей
}

private void Button_page_Patient_Click(object sender,
```

```

RoutedEventArgs e)
    {
        FrameNavigations.Navigate(new
Pacient_page()); //Выводим страницу пациентов
    }

    private void Button_page_Result_Click(object sender,
RoutedEventArgs e)
    {
        FrameNavigations.Navigate(new
Result_page()); //Выводим страницу результаты
    }

    private void Button_page_Settings_Click(object sender,
RoutedEventArgs e)
    {
        FrameNavigations.Navigate(new
Setting_page()); //Выводим страницу настроек
    }

    private void Button_page_User_Click(object sender,
RoutedEventArgs e)
    {
        FrameNavigations.Navigate(new
Users_page()); //Выводим страницу пользователей
    }

    private void Button_page_Connection_Click(object sender,
RoutedEventArgs e)
    {
        FrameNavigations.Navigate(new
Connection_page()); //Выводим страницу проверки подключения
    }

    private void Button_page_Back_Click(object sender,
RoutedEventArgs e)
    {
        if (FrameNavigations.CanGoBack) //Проверка можно ли
вернуться на предыдущую страницу
        {
            FrameNavigations.GoBack(); //Если можно то
возвращаемся
        }
    }

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

4.2 Руководство пользователя программы

Руководство пользователя — документ, назначение которого — предоставить людям помощь в использовании некоторой системы. Документ входит в состав технической документации на систему и, как правило, подготавливается техническим писателем [3].

Введение

Область применения

Требования настоящего документа применяются при:

- предварительных комплексных испытаниях;
- опытной эксплуатации;
- приемочных испытаниях;
- промышленной эксплуатации.

Краткое описание возможностей

Программный продукт обеспечивает реализацию следующих

функциональных возможностей:

1. Ведение списка пациентов:

- внесение пациентов в базу данных;
- редактирование данных пациентов;
- удаление данных;

2. Ведение списка врачей:

- внесение врачей в базу данных;
- редактирование данных врачей;
- удаление данных;

3. Просмотр данных:

- просмотр и сортировка списка пациентов;
- просмотр и сортировка списка врачей;
- просмотр и сортировка результатов;

4. Проведение проверки:

- смена картинок из набора представленных;

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

- верное соотношение параметров проверки.

Уровень подготовки пользователя

Пользователи АРМ ОТ должны обладать квалификацией, обеспечивающей, как минимум:

- базовые навыки работы на персональном компьютере с графическим пользовательским интерфейсом;
- базовые навыки использования стандартной клиентской программы (Microsoft Internet Explorer 6.0 и выше или Mozilla FireFox 3.0 и выше) в среде Интернета;
- базовые навыками использования стандартной почтовой программы (создание, отправка и получение e-mail);
- знание основ информационной безопасности;

Помощь с ВКР под ключ!
знаете принципы организации процесса подготовки и публикации материалов.

Перечень эксплуатационной документации, с которой необходимо ознакомиться пользователю
пишем-вкр-сами.рф

- Перед началом работы с АРМ ОТ пользователю рекомендуется ознакомиться с настоящим документом.

Подготовка к работе

Состав и содержание дистрибутивного носителя данных

Для работы с ПО необходимо следующее программное обеспечение:

- операционная система Windows;
- набор библиотек .NET Framework для запуска программы, написанной на данной программной платформе;
- Google Chrome;
- драйвер MySQL Connector ODBC для подключения удаленной базы данных к программе.

Порядок загрузки данных и программ

Перед началом работы с приложением «Информационная система для проверки зрения» на рабочем месте пользователя необходимо выполнить следующие действия:

Выбрать запускаемый файл с программой и дважды кликнуть мышкой на значке.

Порядок проверки работоспособности

Для проверки доступности приложением «Информационная система для проверки зрения» с рабочего места пользователя необходимо выполнить следующие действия:

- Открыть приложение, для этого необходимо дважды кликнуть мышкой на значке программы;
- Убедиться, что в окне открылось приложение «Информационная система для проверки зрения»;
- В случае если приложение не запускается, то следует обратиться в службу поддержки.

Описание операций

Выполняемые функции и задачи

Приложение «Информационная система для проверки зрения» выполняет функции и задачи, приведённые в таблице ниже (таблица 8).

Таблица 8

Функции и задачи программного продукта

Функции	Задачи	Описание
Авторизация	Защита данных от посторонних и распределение функциональных возможностей по ролям	Необходимо ввести пароль и логин для того, чтобы зайти в систему.
Главная страница	Предоставление списка данных и всех имеющихся функций программы	На данной странице представлены все данные и все функции, от которых уже и идет дальнейшая

		работа пользователя
Добавление и редактирование данных	Добавление и редактирование данных в удаленную базу данных	Добавление/редактирование данных в базу/е данных из приложения, через страницу Добавления/редактирование
Удаление данных	Удаление данных в удаленной базе данных	Удаление данных из базы данных через приложение
Поиск в реальном времени	Вывод данных по запросу	Выводит данные по необходимым критериям в реальном времени
Проверка зрения	Вывод картинок с определенным размером в случайном порядке для проверки зрения	Показ картинок с возможностью сменить размер, или картинку из определенного набора
Карточки	Смена набора картинок	Дает пользователю выбор из 3 наборов: буквы, круги, фигуры

Помощь с ВКР под ключ!

Описание операций технологического процесса обработки данных, необходимых для выполнения задач

пишем-вкр-сами.рф

Ниже приведено описание пользовательских операций для выполнения каждой из задач.

Задача: «Аутентификация пользователя»

Задача: «Авторизация»

Операция 1: запуск программы

1. Компьютер пользователя загружен.
2. Приложение установлено.

Подготовительные действия:

Запустить программу

Основные действия в требуемой последовательности:

После запуска программы, появляется окно Авторизация

Заключительные действия:

Не требуются.

Ресурсы, расходуемые на операцию:

18 секунд.

Условия, при соблюдении которых возможно выполнение операции:

Приложение запущено

Подготовительные действия:

Не требуются.

Основные действия в требуемой последовательности:

В появившемся окне Авторизации, нужно ввести логин и пароль

Заключительные действия:

Заполнив поля, нажать на кнопку «Вход»

Ресурсы, расходуемые на операцию:

20 секунд.

Операций 2: ввод данных

Задача: «Добавление информации о врачах»

Условия, при соблюдении которых возможно выполнение операции:

Приложение запущено

Подготовительные действия:

Не требуются.

Основные действия в требуемой последовательности:

1. В главном меню выбрать кнопку «Врачи»
2. На странице Врачи нажать на кнопку «Добавить»
3. Заполнить все необходимые данные

Заключительные действия:

Для завершения действия нажать на кнопку «Сохранить», окно само закроется и вернет вас к главной странице

Ресурсы, расходуемые на операцию:

Зависит от количества вводимых данных.

Задача: «Удаление данных»

Условия, при соблюдении которых возможно выполнение операции:

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

Приложение запущено

Подготовительные действия:

Не требуются.

Основные действия в требуемой последовательности:

1. Выбрать на главной странице нужные данные для удаления
2. Нажать на кнопку «Удалить» левой кнопкой мыши.

Заключительные действия:

В появившемся предупреждении нажать на кнопку «Да»

Ресурсы, расходуемые на операцию:

12 секунд

Задача: «Поиск данных»

Условия, при соблюдении которых возможно выполнение операции:

Приложение запущено

Основные действия в требуемой последовательности:

1. На главной странице навести курсор на поле поиска
2. Начать вводить информацию которую нужно найти

Заключительные действия:

Нажать на критерий

Ресурсы, расходуемые на операцию:

15 секунд

Аварийные ситуации

В случае возникновения ошибок при работе программы не описанных ниже в данном разделе, необходимо обращаться к сотруднику подразделения технической поддержки. Возможные ошибки при работе с программой указаны ниже (таблица 10).

Таблица 10

Возможные ошибки при работе с программным продуктом

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Класс ошибки	Ошибка	Описание ошибки	Требуемые действия пользователя при возникновении ошибки
Ошибка при авторизации	Неверный логин или пароль	Введен неверный логин или пароль	Обратиться к администратору или перепроверить введенные данные
Проблемы с подключением к базе данных	Проблемы с базой данных	Неверные данные для подключения	Обратиться к администратору
Сбой в электропитании рабочей станции	Нет электропитания рабочей станции или произошел сбой в электропитании.	Рабочая станция выключилась или перезагрузилась.	Перезагрузить рабочую станцию.

Рекомендации по освоению

Рекомендуемая литература:

- данное руководство пользователя.

С руководством пользователя на английском языке можно

ознакомиться в приложении Б.

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

4.3 Руководство администратора по ГОСТ 19.503-79

Наименование программы

Программа АИС «Программа для проверки остроты зрения».

Программа написана на языке программирования С# в среде компилятора Visual Studio 2019.

Назначение программы

Данная программа предназначена для автоматизации проверки на остроту зрения.

Условия выполнения программы

Программа была разработана на компьютере с ОС Windows 10. Процессор intel(r) xeon(r) cpu e5-2620 v3 @ 2.40ghz 2.40 ghz, 64-разрядная операционная система, клавиатура и мышь. При таких условиях программа будет функционировать. Также обязательно требуется набор библиотек .NetFramework и сервер.

Выполнение программы

Для работы программы необходима база данных расположенная на удалённом или локальном сервере. Для корректного подключения необходимо подключиться к базе введя имя пользователя, пароль, порт, адрес сервера.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
id_polz	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
login	VARCHAR(30)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
parol	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
type	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Рисунок 37 – Структура базы данных

Помощь с ВКР под ключ! пишем-вкр-сами.рф

В базе данных необходимо создать 4 таблицы (polzovatel, vrach, pacient, test).

Таблица «polzovatel» содержит данные для входа пользователей и содержит четыре столбца (id_polz, login, parol, type) рисунок 15

id_polz	login	parol	type
1	Admin	12345	Admin
2	Justy	123	NULL
3	a	12345	NULL

Рисунок 38 – Структура таблицы «polzovatel»

- id_polz: уникальный идентификатор пользователя;
- login: логин пользователя (обязателен к заполнению, не более 30 символов);
- parol: пароль пользователя (обязателен к заполнению);

- type: тип пользователя (админ или врач).

Таблица «vrach» содержит данные о врачах и может быть заполнена из программы. таблица содержит 6 столбцов

	id_vrach	Name_vrach	Surname_vrach	Patronymic_vrach	Tel_Rab	Vrema_Rab
▶	21	Виктор	Лимонов	Николаевич	8956382757	Будни 9-17
	22	Дмитрий	Венков	Викторович	8954654664	Будни 8- 16
	35	Николай	Пирогов	Олегович	89572666385	Будни 8-17
	40	Александр	Полянский	Сергеевич	89645645644	Будни 17-9
	41	Ольга	Поваляева	Николаевна	89634524554	Будни 17- 9
*	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 39 – Структура таблицы «vrach»

- id_vrach: уникальный идентификатор врача
- Name_vrach: имя врача (обязательно к заполнению, не более 19 символов);
- Surname_vrach: фамилия врача (обязателен к заполнению, не более 25 символов);
- Patronymic_vrach: отчество врача (обязателен к заполнению, не более 25 символов);
- Tel_Rab: рабочий номер телефона (обязателен к заполнению);
- Vrema_Rab: время работы (обязателен к заполнению)

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Таблица «pacient» содержит данные о пациентах и может быть заполнена из программы. таблица содержит 8 столбцов

id_pac	Name_pac	Surname_pac	Patronymic_pac	Data_roshd	Adres	Polis_strah	Telefon
1	Александр	Пирогов	Николаевич	1960-08-12	ул Антресольская д 23 кв 5	345-067-069 45	3454353454
3	Анастасия	Поваляева	Сергеевна	1994-02-25	ул. гагаринская д24 кв7	678-059-285 67	89456456567
4	Ольга	Говердовская	Александровна	1989-05-27	ул.Кирова д 12 кв2	456-657-324 45	89456456567
5	Андрей	Полянский	Васильев	1980-10-01	ул. Челюскинцев д17 кв3	688-079-385 77	89456737854
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 40 – Структура таблицы «pacient»

- id_pac: уникальный идентификатор пациента
- Name_pac: имя пациента (обязательно к заполнению, не более 19 символов);

- Surname_pac: фамилия пациента (обязательно к заполнению, не более 25 символов);
- Patronymic_pac: отчество пациента (обязательно к заполнению, не более 25 символов);
- Data_roshd: дата рождения пациента (обязательно к заполнению);
- Adres: адрес (обязательно к заполнению, не более 200 символов);
- Polis_strah: номер страхового полиса (обязательно к заполнению, не более 50 символов);
- Telefon: номер телефона (обязательно к заполнению)

Таблица «test» содержит данные о результатах проверки и может быть заполнена из программы. таблица содержит 6 столбцов

	id_test	id_vrach	id_pac	Data_test	Eye	Result
▶	1	22	5	2003-05-03	Правый	0.4
	13	22	4	2019-08-19	Правый	0.3
	14	22	1	2020-07-23	Левый	0.5
	27	21	3	2021-05-11	Правый	0.2
	31	21	4	2021-05-11	Левый	0.2
	38	35	3	2021-05-11	Левый	0.3
	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 41 – Структура таблицы «test»

- id_test: уникальный идентификатор тестирования
- id_vrach: уникальный идентификатор врача (обязательно к заполнению);
- id_pac: уникальный идентификатор пациента (обязательно к заполнению);
- Data_test: дата тестирования (обязательно к заполнению);
- Eye: глаз который проверялся (обязательно к заполнению , не более 10 символов);
- Result: результат проверки (обязательно к заполнению , не более 5 символов);

Подключение удаленной базы данных к программе происходит с помощью драйвера MySQL Connector ODBC. Строка подключения расположена в коде программы, а ее параметры расположены в текстовых файлах, расположенных в папке с программой.

Рекомендации по освоению

Рекомендуемая литература:

- данное руководство администратора.

С руководством администратора на английском языке можно ознакомиться в приложении В.

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

ГЛАВА 5 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ

5.1 Разработка мер защиты информации от несанкционированного доступа

Разработка мер защиты информации от несанкционированного доступа

Защита программного обеспечения преследует цели:

- ограничение несанкционированного доступа к программам или их преднамеренное разрушение и хищение;
- исключение несанкционированного копирования (тиражирования) программ.

Программный продукт и базы данных должны быть защищены от нескольких направлений воздействия:

1. человека — хищение машинных носителей и документации программного обеспечения, нарушение работоспособности программного продукта и др.;
2. аппаратуры — подключение к компьютеру аппаратных средств для считывания программ и данных или их физического разрушения;
3. специализированных программ — приведение программного продукта или базы данных в неработоспособное состояние (например, вирусное заражение), несанкционированное копирование программ и базы данных и т.д.

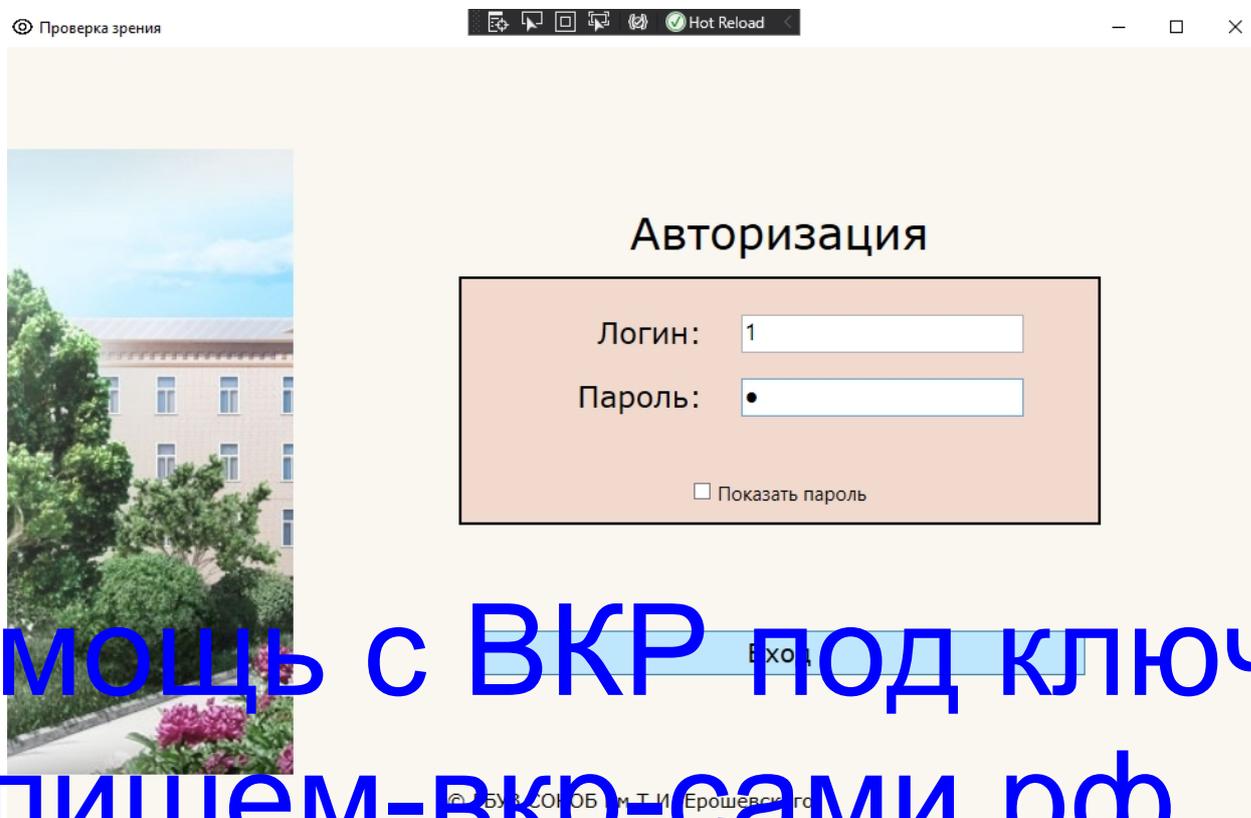
Самый простой и доступный способ защиты программных продуктов и базы данных — ограничение доступа. Контроль доступа к программному продукту и базе данных строится путем:

- парольной защиты программ при их запуске;
- использования ключевой дискеты для запуска программ;

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

- ограничения программ или данных, функций обработки, доступных пользователям, и др.

В программе защита от несанкционированного доступа осуществляется посредством введения формы авторизации, которая запускается первой.



Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Рисунок 42 - Форма авторизации

5.2 Мероприятия по обеспечению техники безопасности и пожарной безопасности

Общие требования безопасности

Настоящая инструкция распространяется эксплуатирующий средства вычислительной техники и периферийное оборудование. Инструкция содержит общие указания по безопасному применению электрооборудования в учреждении. Требования настоящей инструкции являются обязательными, отступления от нее не допускаются. К самостоятельной эксплуатации

электроаппаратуры допускается только специально обученный персонал не моложе 18 лет, пригодный по состоянию здоровья и квалификации к выполнению указанных работ.

Требования безопасности перед началом работы

Перед началом работы следует убедиться в исправности электропроводки, выключателей, штепсельных розеток, при помощи которых оборудование включается в сеть, наличии заземления компьютера, его работоспособности.

Требования безопасности во время работы

Для снижения или предотвращения влияния опасных и вредных факторов необходимо соблюдать санитарные правила и нормы. Гигиенические требования к видео дисплейным терминалам, персональным электронно-вычислительным машинам и организации работы (Утверждено Постановлением Госкомсанэпиднадзора России от 14 июля 1996 г. N 14 СанПиН 2.2.2.542-96).

Во избежание повреждения изоляции проводов и возникновения коротких замыканий не разрешается вешать на провода, закрашивать и белить шнуры и провода, закладывая провода и шнуры за газовые и водопроводные трубы, за батареи отопительной системы, выдергивать штепсельную вилку из розетки за шнур, усилие должно быть приложено к корпусу вилки.

Для исключения поражения электрическим током запрещается: часто включать и выключать компьютер без необходимости, прикасаться к экрану и к тыльной стороне блоков компьютера, работать на средствах вычислительной техники и периферийном оборудовании мокрыми руками, работать на средствах вычислительной техники и периферийном оборудовании, имеющих нарушения целостности корпуса, нарушения изоляции проводов, неисправную индикацию включения питания, с признаками электрического напряжения на корпусе, класть на средства

вычислительной техники и периферийное оборудование посторонние предметы.

Запрещается очищать от пыли и загрязнения электрооборудование под напряжением.

Запрещается проверять работоспособность электрооборудования в непригодных для эксплуатации помещениях с токопроводящими полами, сырых, не позволяющих заземлить доступные металлические части.

Недопустимо под напряжением проводить ремонт средств вычислительной техники и периферийного оборудования. Ремонт электроаппаратуры производится только специалистами-техниками с соблюдением необходимых технических требований.

Во избежание поражения электрическим током, при пользовании электроприборами нельзя касаться одновременно каких-либо трубопроводов, батарей отопления, металлических конструкций, соединенных с землей.

При пользовании электроэнергией в сырых помещениях соблюдать особую осторожность.

Требуется безопасная аварийная ситуация!

При обнаружении неисправности немедленно электрооборудование, оповестить администрацию. Продолжение работы возможно только после устранения неисправности.

При обнаружении оборвавшегося провода необходимо немедленно сообщить об этом администрации, принять меры по исключению контакта с ним людей. Прикосновение к проводу опасно для жизни.

Во всех случаях поражения человека электрическим током немедленно вызывают врача. До прибытия врача нужно, не теряя времени, приступить к оказанию первой помощи пострадавшему.

На рабочем месте запрещается иметь огнеопасные вещества.

При возникновении пожароопасной ситуации или пожара персонал должен немедленно принять необходимые меры для его ликвидации,

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

одновременно оповестить о пожаре администрацию.

Помещения с электрооборудованием должны быть оснащены огнетушителями типа ОУ-2 или ОУБ-3.

Требования безопасности по окончании работы

После окончания работы необходимо обесточить все средства вычислительной техники и периферийное оборудование. В случае непрерывного производственного процесса необходимо оставить включенными только необходимое оборудование.

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

ЗАКЛЮЧЕНИЕ

В ходе выполнения данного дипломного проекта разработано программное обеспечение, предназначенное для сбора, обработки и представления информации в соответствии с составленным техническим заданием. Был произведен выбор соответствующих программ. Была разработана удаленная база данных для хранения информации о результатах проверки на остроту зрения, данных о врачах и пациентах, а также для хранения самих тестов (таблица Сивцева, таблица Головина, таблица Снеллена и таблица Орловой).

База данных была связана с программным продуктом и все функции программного продукта были протестированы. Программный продукт отвечает всем требованиям, которые были поставлены по техническому заданию.

Также при выполнении работы над дипломным проектом гипотеза была подтверждена и выполнены все поставленные задачи, а именно:

- проведен анализ объекта, процесса, предметной области как объекта информатизации;
- проведен анализ информационного обеспечения объекта информатизации;
- построены организационные и функциональные схемы объектов информатизации;
- выполнено описание категорий пользователей и потоков данных объекта информатизации;
- проведен анализ функциональных требований программного продукта;
- проведен анализ входных и выходных данных;
- построена структура данных;
- спроектирован функционал разрабатываемой системы;

- разработаны тестовые сценарии, программы и методики испытаний;
- разработано техническое задание для программного продукта;
- разработаны и оптимизированы схемы данных для программного продукта;
- выбрано программное обеспечение для реализации (разработки) программного продукта;
- выбрано программное обеспечение для разработки и эксплуатации удаленных базы данных;
- разработан графический интерфейс на русском и английском языках для программного продукта;
- разработана удаленная база данных и необходимые запросы;
- реализованы функциональные требования;
- описана интеграция программных модулей;
- выполнена отладка программного продукта;
- проведено тестирование программного продукта;
- выполнено документирование программного кода;
- разработано руководство пользователя для программного продукта;
- разработаны меры защиты информации от несанкционированного доступа;
- описаны мероприятия по обеспечению техники безопасности и пожарной безопасности

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Нормативные материалы

1. ГОСТ 19.105-78 ЕСПД. Общие требования к программным документам.– М.: Стандартформ, 2019
2. РД 50-34.698-90 АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ. ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТОВ (в редакции 2019 года) (дата обращения: 31.05.2022).
3. РД 50-34.698-90. Подразделом 3.4 Руководство пользователя автоматизированной системы (в редакции 2019 года) (дата обращения: 31.05.2022).
4. ГОСТ 19.201-78 ЕСПД. Техническое задание. Требования к содержанию и оформлению.– М.: Стандартформ, 2019
5. ГОСТ 19.404-79 ЕСПД. Пояснительная записка. Требования к содержанию и оформлению.– М.: Стандартформ, 2019
6. ГОСТ 19.602-78 ЕСПД. Правила дублирования, учета и хранения программных документов, выполненных печатным способом.– М.: Стандартформ, 2019
7. ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.– М.: Стандартформ, 2019
8. ГОСТ 34.602-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.– М.: Стандартформ, 2019
9. ГОСТ Р 51188-98 Защита информации. Испытания программных средств на наличие компьютерных вирусов. Типовое руководство.– М.: Стандартформ, 2019

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

10. ГОСТ Р ИСО/МЭК 15504-1-2009 Информационные технологии. Оценка процессов. Часть 1. Концепция и словарь.– М.: Стандартформ, 2019
11. ГОСТ Р ИСО/МЭК 15910-2002 Информационная технология. Процесс создания документации пользователя программного средства.– М.: Стандартформ, 2019

Научные, технические и учебно-методические издания

12. Албахари Б., Албахари Д. С# 7.0. Справочник. Полное описание языка. – М.: Издательский дом «Вильямс», 2018. – 1024 с.
13. Афонин, В. В. Моделирование систем / В.В. Афонин, С.А. Федосин. – М.: Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2021. – 221 с.
14. Бёрд, Ричард. Жемчужины проектирования алгоритмов. Функциональный подход / Ричард Бёрд. – М.: ДМК Пресс, 2019. – 330 с.
15. Васильев А. Программирование на С# для начинающих. Особенности языка. – М.: Издат. Дом «Бомбора», 2018. – 528 с.
16. Васильев А. С# 6.0. Программирование на С# для начинающих. Основные сведения. – М.: Издательский дом «Эксмо», 2018. – 592 с.
17. Гагарина, Л. Г. Разработка и эксплуатация автоматизированных информационных систем. Учебное пособие / Л.Г. Гагарина. – М.: Форум, Инфра-М, 2018. – 384 с.
18. Гвоздева, В. А. Введение в специальность программиста / В.А. Гвоздева. – М.: Форум, Инфра-М, 2018. – 208 с.
19. Гончаров, В. А. Методы оптимизации. Учебное пособие / В.А. Гончаров. – М.: Юрайт, 2018. – 191 с.
20. Зыков, С. В. Основы современного программирования / С.В. Зыков. –

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

М.: Горячая линия – Телеком, 2021. – 400 с.

21. Исаев, Г. Н. Проектирование информационных систем. Учебное пособие / Г.Н. Исаев. – М.: Омега-Л, 2018. – 248 с.
22. Макаровских, Т. А. Документирование программного обеспечения. В помощь техническому писателю. Учебное пособие / Т.А. Макаровских. – М.: Ленанд, 2018. – 266 с.
23. Макконнелл С. Совершенный код. – М.: Издательский дом «Русская редакция», 2017. – 896 с.
24. Мартин Р. Чистый код. Создание, анализ и рефакторинг. Библиотека программиста. – П.: Издат. Дом «Питер», 2017. – 464 с.
25. Рихтер Д. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C#. 4-е изд., 2016, с. 70-105.
26. Соколова, В.В. Вычислительная техника и информационные технологии. Разработка мобильных приложений. Учебное пособие для прикладного бакалавриата / В.В. Соколова. – М.: Юрайт, 2020. – 175 с.
27. Стилмен Э., Грин Д. HeadFirst. Изучаем C#. 3-е изд. 2017. – 384 с.
28. Фленов М. Библия C#. – П.: Изд. Дом «РДВ», 2021. – 544 с.
29. Рихтер Д. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C#. 4-е изд., с. 70-105.
30. Стилмен Э., Грин Д. HeadFirst. Изучаем C#. 3-е изд. 2017. – 384 с.

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Ресурсы сети Интернет

31. Википедия – свободная энциклопедия [Электронный ресурс]: Описание MySQL – Режим доступа: <https://ru.wikipedia.org/wiki/MySQL>
32. Ресурс для IT-специалистов [Электронный ресурс]: Руководство MySQL Workbench – Режим доступа: <https://simple-scada.com/help/manual/mysql-workbench.html>
33. Википедия – свободная энциклопедия [Электронный ресурс]:

- Описание Таблица Сивцева – Режим доступа:
https://ru.wikipedia.org/wiki/Таблица_Сивцева
34. Википедия – свободная энциклопедия [Электронный ресурс]:
 Описание Таблица Орловой – Режим доступа:
https://ru.wikipedia.org/wiki/Таблица_Орловой
35. Википедия – свободная энциклопедия [Электронный ресурс]:
 Описание Таблица Снеллеа – Режим доступа:
https://ru.wikipedia.org/wiki/Таблица_Снеллеа
36. Википедия – свободная энциклопедия [Электронный ресурс]:
 Описание Таблица Головина – Режим доступа:
https://ru.wikipedia.org/wiki/Таблица_Головина
37. Википедия – свободная энциклопедия [Электронный ресурс]:
 Описание Microsoft Visual Studio – Режим доступа:
https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio
38. Википедия – свободная энциклопедия [Электронный ресурс]:
 Описание Java – Режим доступа: <https://ru.wikipedia.org/wiki/Java>
39. Ресурс для IT-специалистов [Электронный ресурс]: Как можно
 добавить данные из Excel в listBox? – Режим доступа:
<http://www.cyberforum.ru/csharp-net/thread376275.html>
40. Ресурс для IT-специалистов [Электронный ресурс]: Свойство
 DateTime.Today – Режим доступа: [https://msdn.microsoft.com/ru-ru/library/system.datetime.today\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/system.datetime.today(v=vs.110).aspx)
41. Ресурс для IT-специалистов [Электронный ресурс]: Работа с классами
 c# – Режим доступа: <https://metanit.com/sharp/tutorial/3.1.php>
42. Ресурс для IT-специалистов [Электронный ресурс]: Основы NPOI –
 Режим доступа: <https://stackoverflow.com/questions/5855813/how-to-read-file-using-npoi>
43. Ресурс для IT-специалистов [Электронный ресурс]: Основы NPOI в
 Excel – Режим доступа:

**Помощь с ВКР под ключ!
 пишем-вкр-сами.рф**

<http://codemonkeydeveloper.blogspot.com/2015/03/npoi.html>

44. Ресурс для IT-специалистов [Электронный ресурс]: Редактирование ячеек NPOI в Excel – Режим доступа: <http://www.cyberforum.ru/csharp-beginners/thread2409698.html>

45. Ресурс для IT-специалистов [Электронный ресурс]: Ответы на 7 самых частых вопросов по WinForms – Режим доступа: <http://www.cyberforum.ru/windows-forms/thread110436.html>

46. Ресурс для IT-специалистов [Электронный ресурс]: Удаленная база данных – Режим доступа: <http://www.realcoding.net/articles/udalennaya-baza-dannykh.html>

47. Ресурс для IT-специалистов [Электронный ресурс]: Оператор SQL UPDATE – Режим доступа: <http://2sql.ru/novosti/sql-update/>

48. Ресурс для IT-специалистов [Электронный ресурс]: Оператор SQL SELECT – Режим доступа: [https://ru.wikipedia.org/wiki/Select_\(SQL\)](https://ru.wikipedia.org/wiki/Select_(SQL))

49. Ресурс для IT-специалистов [Электронный ресурс]: MySQL: особенности и сферы применения – Режим доступа: <https://www.bytemag.ru/articles/detail.php?ID=6547>

50. Ресурс для IT-специалистов [Электронный ресурс]: Методы тестирования ПО – Режим доступа: <http://juice-health.ru/program/software-testing/495-software-testing-methods>

Помощь с ВКР под ключ!
пишем вкр сами.рф

ПРИЛОЖЕНИЕ А

UML - диаграммы

UML – диаграммы

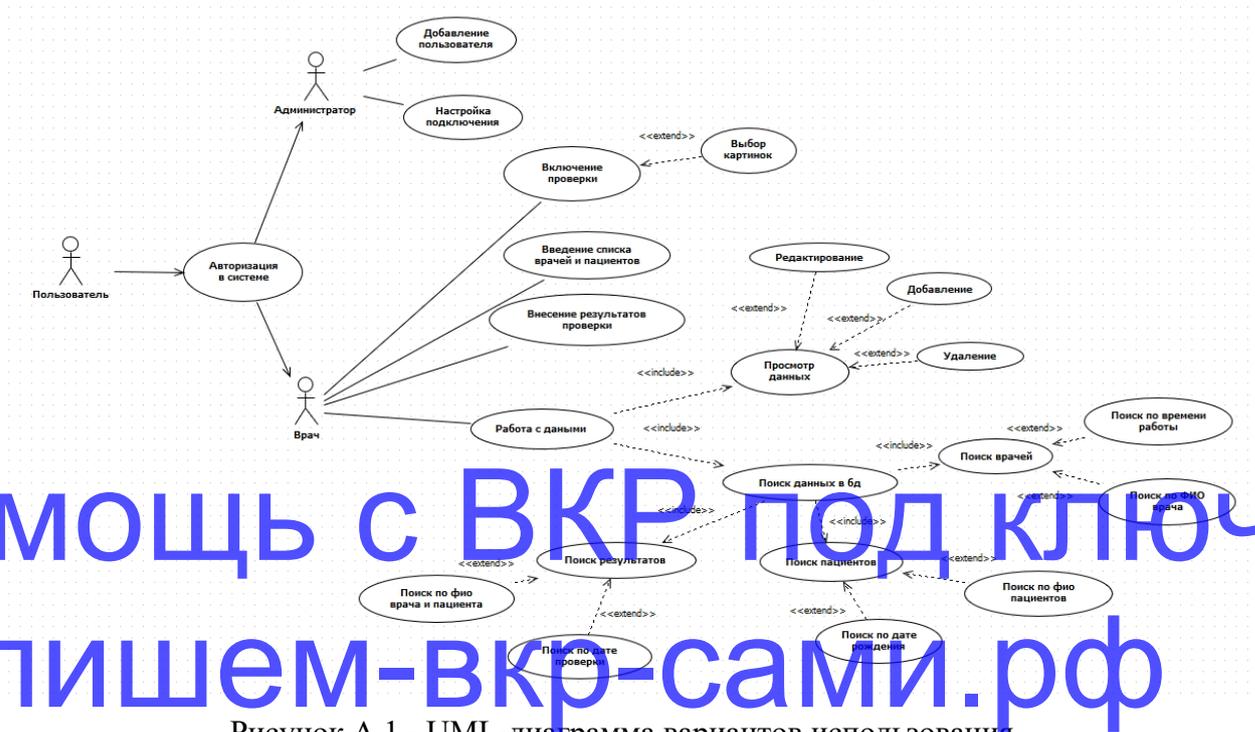


Рисунок А.1 - UML-диаграмма вариантов использования.

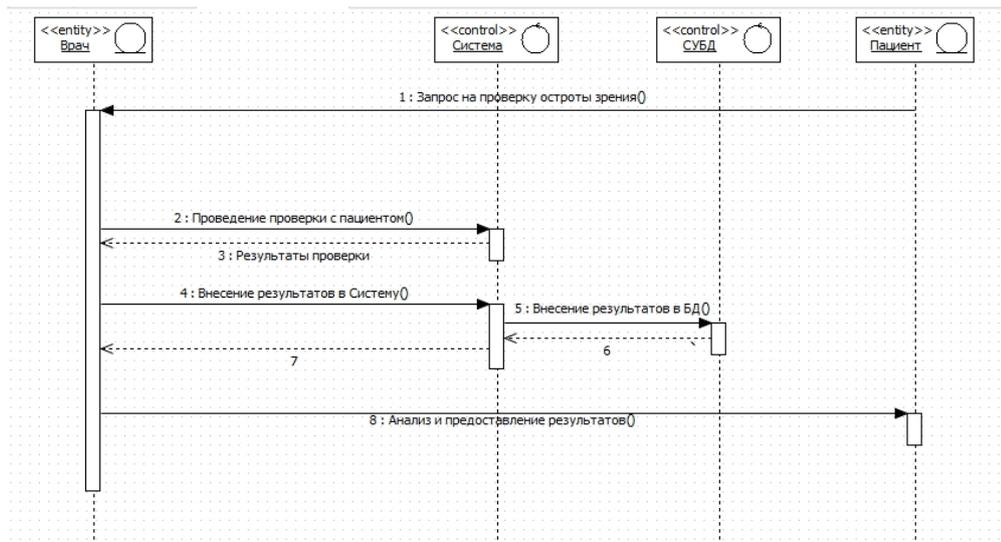


Рисунок А.2 - UML-диаграмма последовательностей

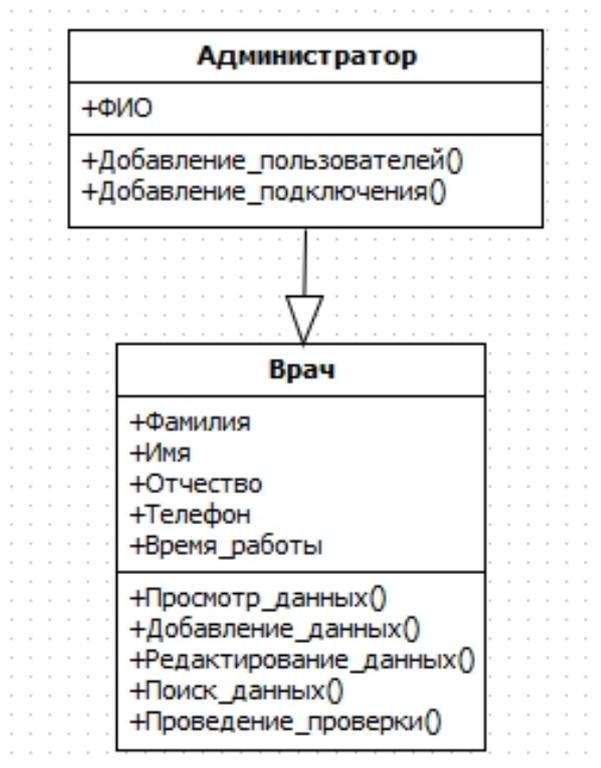


Рисунок А.3 - UML-диаграмма классов

Помощь с ВКР под ключ! пишем-вкр-сами.рф

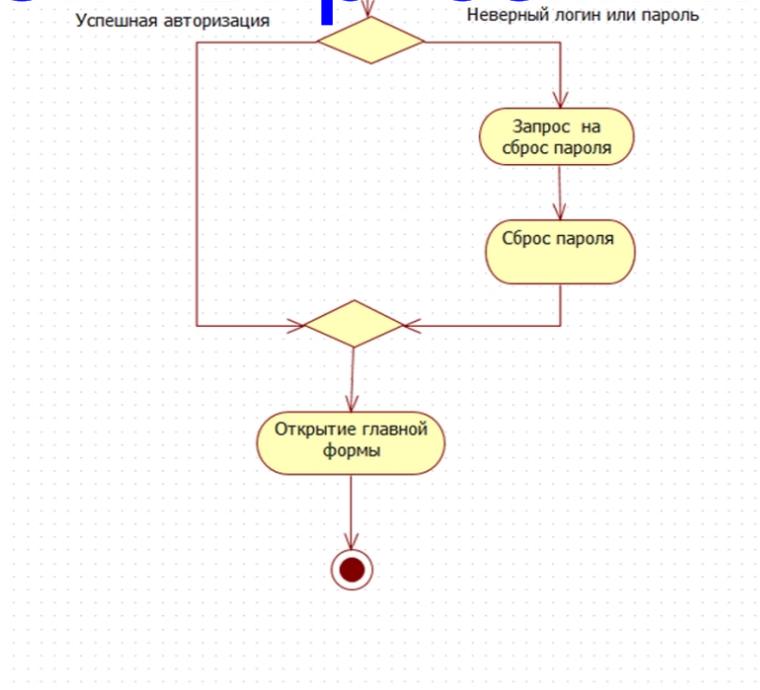


Рисунок А.4 - Диаграмма активностей

ПРИЛОЖЕНИЕ Б

Руководство пользователя (на английском)

Руководство пользователя (на английском)

Introduction

Scope of application

The requirements of this document apply when:

- preliminary complex tests;
- trial operation;
- acceptance tests;
- industrial operation.

A brief description of the features

The software product provides the implementation of the following functionality:

1. Maintaining a list of patients:

- adding patients to the database;
- editing patient data;
- data deletion;

2. Maintaining a list of doctors:

- adding doctors to the database;
- editing doctors' data;
- data deletion;

3. Viewing data:

- view and sort the list of patients;
- view and sort the list of doctors;
- viewing and sorting results;

4. Conducting an inspection:

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

- change of images from the set presented;
- the correct ratio of the verification parameters

User training level

Users of the automated control system must have qualifications that provide at least:

- basic skills of working on a personal computer with a graphical user interface;
- basic skills of using a standard client program (Microsoft Internet Explorer 6.0 and higher or Mozilla FireFox 3.0 and higher) in an Internet environment;
- basic skills of using a standard mail program (creating, sending and receiving e-mail);
- knowledge of the basics of information security;

knowledge of the principles of organizing the process of preparation and publication of materials.

The list of operational documentation that the user needs to read

Before starting to work with the automated control system, the user is recommended to familiarize himself with this document.

Preparation for work

Composition and content of the distributive data carrier

To work with the software, you need the following software:

- Windows operating system;
- a set of libraries .NET Framework for running a program written on this software platform;
- Google Chrome;
- MySQL Connector ODBC driver for connecting a remote database to the program.

The order of loading data and programs

Before you start working with the application "Information system for vision testing" at the user's workplace, you must perform the following actions:

Select the executable file with the program and double-click on the icon.

The procedure for checking the operability

To check the availability of the application "Information system for vision testing" from the user's workplace, you must perform the following actions:

- Open the application, to do this, double-click on the program icon;
- Make sure that the application "Information System for vision check" has opened in the window;
- If the application does not start, you should contact the support service.

Description of operations

Functions and tasks performed

The application "Information System for vision testing" performs the functions and tasks listed in the table below (Table 8).

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Table 8

Functions and tasks of the software product

Functions	Task	Description
Authorization	Data protection from outsiders and distribution of functionality by roles It is necessary to enter a password and login in order to log in to the system.	You must enter your password and login in order to log in.
Home page	Providing a list of data and all available program functions	This page presents all the data and all the functions from which the user's further work is already underway
Adding and editing data	Adding and editing data to a remote database	Adding/editing data to the database from the application, via the Add/Edit page.
Deleting data	Deleting data in a remote database	Deleting data from the database via the application
Real-time search	Data output on request	Displays data on the necessary criteria in real time
Vision check	Output of images with a certain size in random order to check vision	Showing images with the ability to change the size, or a picture from a certain set

Cards	Changing a set of images	Gives the user a choice of 3 sets: letters, circles, shapes
-------	--------------------------	-------------------------------------------------------------

Description of the operations of the technological process of data processing required to perform tasks

Below is a description of user operations to perform each of the tasks.

Task: "User authentication"

Task: "Authorization"

Operation 1: Start the program

1. The user's computer is loaded.
2. The application is installed.

Preparatory actions:

Start the program

The main actions in the required sequence:

After starting the program, the Authorization window appears

Final actions:

Not required.

Resources spent on the operation:

18 seconds.

Conditions under which the operation can be performed:

The application is running

Preparatory actions:

Not required.

The main actions in the required sequence:

In the Authorization window that appears, you need to enter your username and password

Final actions:

After filling in the fields, click on the "Login" button

Resources spent on the operation:

20 seconds.

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Operations 2: Data entry

Task: "Adding information about doctors"

Conditions under which it is possible to perform the operation:

The application is running

Preparatory actions:

Not required.

Basic actions in the required sequence:

1. In the main menu, select the "Doctors" button
2. On the Doctors page, click on the "Add" button
3. Fill in all the necessary data

Final actions:

To complete the action, click on the "Save" button, the window will close itself and return you to the main page

Resources spent on the operation:

Depends on the amount of data entered.

Task: "Data deletion"

Conditions under which it is possible to perform the operation:

The application is running

Preparatory actions:

Not required.

Basic actions in the required sequence:

1. Select the necessary data to delete on the main page
2. Click on the "Delete" button with the left mouse button.

Final steps:

In the warning that appears, click on the "Yes" button.

The resources spent on the operation:

10 seconds

Task: "Data search"

Conditions under which it is possible to perform the operation:

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

The application is running

Basic actions in the required sequence:

1. On the main page, hover the cursor over the search field
2. Start entering the information you need to find

Final actions:

Click on the criteria

Resources spent on the operation:

15 seconds

Emergency situations

In case of errors during the operation of the program not described below in this section, it is necessary to contact an employee of the technical support department. Possible errors when working with the program are listed below (Table 9).

Table 9
Possible errors when working with the software product

Error class	Error	Error description	Required user actions when an error occurs
Authorization error	Invalid username or password	Invalid username or password entered	Contact the administrator or double-check the entered data
Problems connecting to the database	Problems with the database	Invalid connection data	Contact the administrator
Power failure of the workstation	There is no power supply to the workstation or there has been a power failure.	The workstation has shut down or rebooted.	Restart the workstation.

Recommendations for mastering

Recommended literature:

- this user manual.

ПРИЛОЖЕНИЕ В

Руководство администратора (на английском)

Руководство администратора (на английском)

Name of the program

AIS program "Program for visual acuity testing".

The program is written in the C# programming language in the Visual Studio 2019 compiler environment.

Purpose of the program

This program is designed to automate visual acuity testing.

Program execution conditions

The program was developed on a Windows 10 computer. Intel(R) XEON(R) CPU E5-2620 v3 @ 2.40GHz 2.40 GHz, 64-bit operating system, keyboard and mouse. Under such conditions, the program will function. A set of libraries is also required .NetFramework and the server.

Program execution

For the program to work, a database located on a remote or local server is required. To connect correctly, you need to connect to the database by entering the user name, password, port, server address.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
id_polz	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
login	VARCHAR(30)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
parol	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
type	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure B.1 – Database structure

In the database, you need to create 4 tables (polzovatel, vrach, pacient, test).

The "polzovatel" table contains user login data and contains four columns (id_polz, login, parol, type) figure 15

id_polz	login	parol	type
1	Admin	12345	Admin
2	Justy	123	NULL
3	a	12345	NULL

Figure B.2 – Structure of the "polzovatel" table

- Id_polz: unique user ID;
- login: user login (required, no more than 30 characters);
- parol: user password (required);
- type: user type (admin or doctor).

The "vrach" table contains data about doctors and can be filled in from the program. the table contains 6 columns

id_vrach	Name_vrach	Surname_vrach	Patronymic_vrach	Tel_Rab	Vrema_Rab
21	Виктор	Лимонов	Николаевич	8956382757	Будни 9-17
22	Дмитрий	Венков	Викторович	8954654664	Будни 8- 16
35	Николай	Пирогов	Слегович	8957267335	Будни 9-17
40	Александр	Полянский	Сергеевич	8965645644	Будни 17-9
41	Ольга	Поваляева	Николаевна	8963452454	Будни 17-9
NULL	NULL	NULL	NULL	NULL	NULL

Figure B.3 – Structure of the "vrach" table

- id_vrach: unique identifier of the doctor
- Name_vrach: name of the doctor (required, no more than 19 characters);
- Surname_vrach: surname of the doctor (required, no more than 25 characters);
- Patronymic_vrach: patronymic of the doctor (required, no more than 25 characters);
- Tel_Rab: work phone number (required);
- Vrema_Rab: working hours (required)

The "pacient" table contains data about patients and can be filled in from the program. the table contains 8 columns

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

id_pac	Name_pac	Surname_pac	Patronymic_pac	Data_roshd	Adres	Polis_strah	Telefon
1	Александр	Пирогов	Николаевич	1960-08-12	ул Антресольская д 23 кв 5	345-067-069 45	3454353454
3	Анастасия	Поваляева	Сергеевна	1994-02-25	ул. гагаринская д24 кв7	678-059-285 67	89456456567
4	Ольга	Говердовская	Александровна	1989-05-27	ул.Кирова д 12 кв2	456-657-324 45	89456456567
5	Андрей	Полянский	Васильев	1980-10-01	ул. Челюскинцев д 17 кв3	688-079-385 77	89456737854
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure B.4– Structure of the "pacient" table

- id_pac: unique patient ID
- Name_pac: patient's name (required, no more than 19 characters);
- Surname_pac: patient's last name (required, no more than 25 characters);
- Patronymic_pac: patronymic of the patient (required, no more than 25 characters);
- Data_roshd: date of birth of the patient (required);
- Adres: address (required, no more than 200 characters);
- Polis_strah: insurance policy number (required, no more than 50 characters);
- Telefon: phone number (required)

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

The "test" table contains data on the test results and can be filled in from the program, the table contains 6 columns

	id_test	id_vrach	id_pac	Data_test	Eye	Result
▶	1	22	5	2003-05-03	Правый	0.4
	13	22	4	2019-08-19	Правый	0.3
	14	22	1	2020-07-23	Левый	0.5
	27	21	3	2021-05-11	Правый	0.2
	31	21	4	2021-05-11	Левый	0.2
	38	35	3	2021-05-11	Левый	0.3
▲	NULL	NULL	NULL	NULL	NULL	NULL

Figure B.5 – Structure of the "test" table

- id_test: unique testing identifier
- id_vrach: unique identifier of the doctor (required);
- id_pac: unique patient ID (required);
- Data_test: test date (required);
- Eye: the eye that was checked (required to be filled in, no more than 10

- characters);
- Result: the result of the check (required, no more than 5 characters);

The connection of the remote database to the program occurs with using the MySQL Connector ODBC driver. The connection string is located in the program code, and its parameters are located in text files located in the program folder.

Recommendations for mastering

Recommended literature:

- this administrator's guide.

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

ПРИЛОЖЕНИЕ Г

Документированный код программного продукта

Документированный код программного продукта

```
//Подключение

public partial class EfModel : DbContext
{
    public static EfModel instance;
    public static EfModel init()
    {
        if(instance==null)
        {
            instance = new EfModel();
        }
        return instance;
    }

    public EfModel()
        : base("server="+ Properties.Settings.Default.Server
        +";user id="+ Properties.Settings.Default.userid +";password="+
        Properties.Settings.Default.password
        +";persistsecurityinfo=True;database="+
        Properties.Settings.Default.database +";")
    {
    }

    public virtual DbSet<KardBukvi> KardBukvis { get; set; }
    public virtual DbSet<KardLetter> KardLetters { get; set; }
}

public virtual DbSet<KardsCircle> KardsCircles { get;
set; }

public virtual DbSet<pacient> patients { get; set; }
public virtual DbSet<polzovatel> polzovatels { get; set; }
}

public virtual DbSet<test> tests { get; set; }
public virtual DbSet<vrach> vraches { get; set; }

protected override void OnModelCreating(DbModelBuilder
modelBuilder)
{
    modelBuilder.Entity<KardBukvi>()
        .Property(e => e.Name)
        .IsUnicode(false);

    modelBuilder.Entity<KardLetter>()
```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

        .Property(e => e.Name)
        .IsUnicode(false);

modelBuilder.Entity<KardsCircle>()
    .Property(e => e.Name)
    .IsUnicode(false);

modelBuilder.Entity<pacient>()
    .Property(e => e.Name_pac)
    .IsUnicode(false);

modelBuilder.Entity<pacient>()
    .Property(e => e.Surname_pac)
    .IsUnicode(false);

modelBuilder.Entity<pacient>()
    .Property(e => e.Patronymic_pac)
    .IsUnicode(false);

modelBuilder.Entity<pacient>()
    .Property(e => e.Adres)
    .IsUnicode(false);

modelBuilder.Entity<pacient>()
    .Property(e => e.Polis_surnam)
    .IsUnicode(false);

modelBuilder.Entity<pacient>()
    .HasMany(e => e.testes)
    .WithRequired(e => e.pacient)
    .WillCascadeOnDelete(false);

modelBuilder.Entity<polzovatel>()
    .Property(e => e.login)
    .IsUnicode(false);

modelBuilder.Entity<polzovatel>()
    .Property(e => e.type)
    .IsUnicode(false);

modelBuilder.Entity<test>()
    .Property(e => e.Eye_Right)
    .IsUnicode(false);

modelBuilder.Entity<test>()
    .Property(e => e.Eye_Left)
    .IsUnicode(false);

modelBuilder.Entity<vrach>()
    .Property(e => e.Name_vrach)
    .IsUnicode(false);

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

        modelBuilder.Entity<vrach>()
            .Property(e => e.Surname_vrach)
            .IsUnicode(false);

        modelBuilder.Entity<vrach>()
            .Property(e => e.Patronymic_vrach)
            .IsUnicode(false);

        modelBuilder.Entity<vrach>()
            .Property(e => e.Vrema_Rab)
            .IsUnicode(false);

        modelBuilder.Entity<vrach>()
            .HasMany(e => e.tests)
            .WithRequired(e => e.vrach)
            .WillCascadeOnDelete(false);
    }
}

//Добавление/проверка подключения

public partial class Connection_page : Page
{
    public Connection_page()
    {
        InitializeComponent();
        Tb_Data_Base.Text =
        Properties.Settings.Default.database;
        Tb_Pas.Text = Properties.Settings.Default.password;
        Tb_Server.Text = Properties.Settings.Default.Server;
        Tb_User.Text = Properties.Settings.Default.userid;
    }

    private void Button_Save_Connect_Click(object sender,
        RoutedEventArgs e)
    {
        if ((Tb_Data_Base.Text==null) || (Tb_Server.Text==null)
            || (Tb_User.Text==null))
        {
            MessageBox.Show("Заполните все поля!!!");
        }
        else
        {
            Properties.Settings.Default.database =
            Tb_Data_Base.Text;
            Properties.Settings.Default.password =
            Tb_Pas.Text;
            Properties.Settings.Default.Server =
            Tb_Server.Text;
            Properties.Settings.Default.userid =
            Tb_User.Text;
        }
    }
}

```

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

```

        Properties.Settings.Default.Save();

        MessageBox.Show("Данные успешно сохранены!!!");
    }
}

private bool IsConnect()
{
    MySqlConnectionStringBuilder bilder = new
MySqlConnectionStringBuilder();

    bilder.Database = Tb_Data_Base.Text;
    bilder.Password = Tb_Pass.Text;
    bilder.Server = Tb_Server.Text;
    bilder.UserID = Tb_User.Text;

    MySqlConnection connect = new
MySqlConnection(bilder.ConnectionString);

    try
    {
        connect.Open();
        return true;
    }
    catch
    {
        return false;
    }
}

private void Button_Check_Connect_Click(object sender,
RoutedEventArgs e)
{
    if(IsConnect())
    {
        MessageBox.Show("Подключение прошло
успешно!!!");
    }

    else
    {
        MessageBox.Show("Ошибка!!! Подключение не
удалось!!!");
    }
}
}
}

```

Помощь с ВКР под ключ! пишем-вкр-сами.рф

Авторизация

```
public partial class Autorisation_page : Page
```

```

    {
        public Autorisation_page()
        {
            InitializeComponent();
        }

        private void Button_Autorisation_Click(object sender,
RoutedEventArgs e)
        {

            if (TB_Parol.Text == null || TB_Login.Text == null)
            {
                MessageBox.Show(Messages.FillAllFields);
            }

            else
            {
                polzovatel polzovatel =
EfModel.init().polzovatels.FirstOrDefault(
                p => p.login == TB_Login.Text &&
                (p.parol.ToString() == PB_Parol.Password
||
                p.parol.ToString() == TB_Parol.Text));

                if (polzovatel != null)
                {
                    AuthClass.polzovatel = polzovatel;
                    if (polzovatel.type == "Admin")
                    {
                        NavigationService.Navigate(new
Users_page());
                    }

                    else
                    {
                        NavigationService.Navigate(new
Proverka_Bukvi_page());
                    }

                    else
                    {
                        MessageBox.Show(Messages.UncorrectLoginPass)
;
                    }
                }
            }

            private void Show_Pas(object sender, RoutedEventArgs e)
            {
                TB_Parol.Text = PB_Parol.Password;
                TB_Parol.Visibility = Visibility.Visible;
            }
        }
    }

```

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

```

        PB_Parol.Visibility = Visibility.Collapsed;
    }

    private void Hide_Pas(object sender, RoutedEventArgs e)
    {
        PB_Parol.Password = TB_Parol.Text;
        TB_Parol.Visibility = Visibility.Collapsed;
        PB_Parol.Visibility = Visibility.Visible;
    }
}

//Выбор карточек

public partial class Cards_page : Page
{
    public Cards_page()
    {
        InitializeComponent();
    }

    private void Figure_Card(object sender,
    MouseButtonEventArgs e)
    {
        NavigationService.Navigate(new
    Proverka_Figure_page());
    }

    private void Letter_Card(object sender,
    MouseButtonEventArgs e)
    {
        NavigationService.Navigate(new
    Proverka_Bukvi_page());
    }

    private void Circl_Card(object sender,
    MouseButtonEventArgs e)
    {
        NavigationService.Navigate(new
    Proverka_Circl_page());
    }
}

//Добавление пациента

public partial class Pacient_add_page : Page
{
    patient patient;
    public Pacient_add_page(patient patient)
    {
        this.pacient = patient;
        DataContext = patient;
    }
}

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

        InitializeComponent();
    }

    private void Button_Add_Click(object sender,
RoutedEventArgs e)
    {
        if (pacient.id_pac == 0)
        {
            EfModel.init().patients.Add(pacient);
        }

        EfModel.init().SaveChanges();
        NavigationService.GoBack();
    }

    private void Pac_Vis_Change(object sender,
DependencyPropertyChangedEventArgs e)
    {
        if (pacient.id_pac != 0)
        {
            EfModel.init().Entry(pacient).Reload();
        }
    }
}
}

```

Помощь с ВКР под ключ!

//Страница пациенты

пишем-вкр-сами.рф

```

public partial class Orient_Page : Page
{
    public void UpdateData()
    {
        List<pacient> patients =
EfModel.init().patients.Where(t =>
t.Name_pac.Contains(TB_Search.Text) ||
t.Surname_pac.Contains(TB_Search.Text)
|| t.Patronymic_pac.Contains(TB_Search.Text) ||
t.Telefon.ToString().Contains(TB_Search.Text)).ToList();

        Lv_Pac.ItemsSource = patients;

        if (patients.Count <= 0)
        {
            Tb_NoRes.Visibility = Visibility.Visible;
        }

        else
        {
            Tb_NoRes.Visibility = Visibility.Hidden;
        }
    }
}

```

```

public Pacient_page()
{
    InitializeComponent();
    UpdateData();
}

private void Button_Edit_Pac_Click(object sender,
RoutedEventArgs e)
{
    Pacient pacient = (sender as Button).DataContext as
pacient;
    NavigationService.Navigate(new
Pacient_add_page(pacient));
}

private void Button_Delete_Pac_Click(object sender,
RoutedEventArgs e)
{
    Pacient pac = (sender as Button).DataContext as
pacient;

    if (MessageBox.Show("Вы точно хотите удалить запись
" + pac.Name_pac + "?", "Удаление", MessageBoxButton.YesNo) ==
MessageBoxResult.Yes)
    {
        foreach (test test in pac.tests.ToList())
        {
            EfModel.init().tests.Remove(test);
        }

        EfModel.init().patients.Remove(pac);
        EfModel.init().SaveChanges();

        UpdateData();
    }
}

private void Button_Add_Pac_Click(object sender,
RoutedEventArgs e)
{
    NavigationService.Navigate(new Pacient_add_page(new
pacient()));
}

private void Search_Pac(object sender,
TextChangedEventArgs e)
{
    UpdateData();
}

private void Pac_Vis_Change(object sender,

```

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

```

DependencyPropertyChangedEventArgs e)
    {
        UpdateData();
    }
}

//Страница проверки

public partial class Proverka_Bukvi_page : Page
{
    public Proverka_Bukvi_page()
    {
        InitializeComponent();
        UpdateCard();
    }

    public int b = 1;

    public KardBukvi Bukvi { get; set; }

    int Pos = 0;

    Random rand = new Random();

    private int nextRand()
    {
        int nextPos =
            rand.Next(EfModel.init().KardBukvis.Count());
        if (Pos == nextPos)
            nextPos = nextRand();
    }

    return nextPos;
}

public void UpdateCard()
{
    Pos = nextRand();
    Bukvi = EfModel.init().KardBukvis.OrderBy(k =>
k.idKards).Skip(Pos).First();
    ImgTest.DataContext = Bukvi;
}

private void Button_Bigger_Click(object sender,
RoutedEventArgs e)
{
    b--;
    if (b <= 0)
    {
        b = 1;
        Tb_Max.Visibility = Visibility.Visible;
    }
}

```

Помощь с ВКР под ключ! пишем-вкр-сами.рф

```

        Tb_Min.Visibility = Visibility.Hidden;
    }
else
{
    Tb_Max.Visibility = Visibility.Hidden;
    Tb_Min.Visibility = Visibility.Hidden;

    switch (b)
    {
        case 1:
            ImgTest.Width = 370;

            TextBlock_V.Text = "V= 0.1";
            TextBlock_D.Text = "D= 50.0";
            UpdateCard();

            return;

        case 2:
            ImgTest.Width = 185;

            TextBlock_V.Text = "V= 0.2";
            TextBlock_D.Text = "D= 25.0";
            UpdateCard();
            return;

        case 3:
            ImgTest.Width = 120;

            TextBlock_V.Text = "V= 0.3";
            TextBlock_D.Text = "D= 16.67";
            UpdateCard();

            return;

        case 4:
            ImgTest.Width = 95;

            TextBlock_V.Text = "V= 0.4";
            TextBlock_D.Text = "D= 12.5";
            UpdateCard();

            return;

        case 5:
            ImgTest.Width = 70;

            TextBlock_V.Text = "V= 0.5";
            TextBlock_D.Text = "D= 10.0";
            UpdateCard();
    }
}

```

Помощь с ВКР под ключ!
 пишем-вкр-сами.рф

```

return;

case 6:
    ImgTest.Width = 60;

    TextBlock_V.Text = "V= 0.6";
    TextBlock_D.Text = "D= 8.33";
    UpdateCard();

    return;

case 7:
    ImgTest.Width = 51;

    TextBlock_V.Text = "V= 0.7";
    TextBlock_D.Text = "D= 7.14";
    UpdateCard();

    return;

case 8:
    ImgTest.Width = 47;
    TextBlock_V.Text = "V= 0.8";
    TextBlock_D.Text = "D= 6.25";
    UpdateCard();

    return;

case 9:
    ImgTest.Width = 43;

    TextBlock_V.Text = "V= 0.9";
    TextBlock_D.Text = "D= 5.55";
    UpdateCard();

    return;

case 10:
    ImgTest.Width = 39;

    TextBlock_V.Text = "V= 1.0";
    TextBlock_D.Text = "D= 5.0";
    UpdateCard();

    return;

case 11:
    ImgTest.Width = 25;

    TextBlock_V.Text = "V= 1.5";

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

        TextBlock_D.Text = "D= 3.33";
        UpdateCard();

        return;

    case 12:
        ImgTest.Width = 20;

        TextBlock_V.Text = "V= 2.0";
        TextBlock_D.Text = "D= 2.5";
        UpdateCard();
        b = 12;

        break;
    }
}

private void Button_Smoler_Click(object sender,
RoutedEventArgs e)
{
    b++;
    if (b >= 13)
    {
        b = 12;
        Tb_Min.Visibility = Visibility.Visible;
        Tb_Max.Visibility = Visibility.Hidden;
    }
    else
    {
        Tb_Min.Visibility = Visibility.Hidden;
        Tb_Max.Visibility = Visibility.Hidden;

        switch (b)
        {
            case 1:
                ImgTest.Width = 370;

                TextBlock_V.Text = "V= 0.1";
                TextBlock_D.Text = "D= 50.0";
                UpdateCard();

                return;

            case 2:
                ImgTest.Width = 185;

                TextBlock_V.Text = "V= 0.2";
                TextBlock_D.Text = "D= 25.0";
                UpdateCard();

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

return;

case 3:
    ImgTest.Width = 120;

    TextBlock_V.Text = "V= 0.3";
    TextBlock_D.Text = "D= 16.67";
    UpdateCard();

    return;

case 4:
    ImgTest.Width = 95;

    TextBlock_V.Text = "V= 0.4";
    TextBlock_D.Text = "D= 12.5";
    UpdateCard();

    return;

case 5:
    ImgTest.Width = 70;

    TextBlock_V.Text = "V= 0.5";
    TextBlock_D.Text = "D= 11.0";
    UpdateCard();

    return;

case 6:
    ImgTest.Width = 60;

    TextBlock_V.Text = "V= 0.6";
    TextBlock_D.Text = "D= 8.33";
    UpdateCard();

    return;

case 7:
    ImgTest.Width = 51;

    TextBlock_V.Text = "V= 0.7";
    TextBlock_D.Text = "D= 7.14";
    UpdateCard();

    return;

case 8:
    ImgTest.Width = 47;

    TextBlock_V.Text = "V= 0.8";
    TextBlock_D.Text = "D= 6.25";

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

Помощь с ВКР под ключ! пишем-вкр-сами.рф

```
        UpdateCard();

        return;

    case 9:
        ImgTest.Width = 43;

        TextBlock_V.Text = "V= 0.9";
        TextBlock_D.Text = "D= 5.55";
        UpdateCard();

        return;

    case 10:
        ImgTest.Width = 39;

        TextBlock_V.Text = "V= 1.0";
        TextBlock_D.Text = "D= 5.0";
        UpdateCard();

        return;

    case 11:
        ImgTest.Width = 25;
        TextBlock_V.Text = "V= 1.5";
        TextBlock_D.Text = "D= 3.33";
        UpdateCard();
        return;

    case 12:
        ImgTest.Width = 20;

        TextBlock_V.Text = "V= 2.0";
        TextBlock_D.Text = "D= 2.5";
        UpdateCard();

        break;
    }
}

private void Button_Next_Click(object sender,
RoutedEventArgs e)
{
    UpdateCard();
    Tb_Min.Visibility = Visibility.Hidden;
    Tb_Max.Visibility = Visibility.Hidden;
}

private void Button_Cards_Click(object sender,
```

```

RoutedEventArgs e)
    {
        NavigationService.Navigate(new Cards_page());
    }

    private void Button_Result_Click(object sender,
RoutedEventArgs e)
    {
        NavigationService.Navigate(new Result_add_page(new
test()));
    }
}

```

//Страница добавления результатов

```

public partial class Result_add_page : Page
    {
        test result;
        public Result_add_page(test result)
        {
            InitializeComponent();

            this.result = result;
            DataContext = result;
            Pac_Cb.ItemsSource =
EfModel.init().patients.ToList();
            Res_Cb.ItemsSource =
EfModel.init().visches.ToList();
        }
    }

```

```

    private void Button_Add_Result_Click(object sender,
RoutedEventArgs e)
    {
        if (result.id_test == 0)
        {
            EfModel.init().tests.Add(result);
        }

        EfModel.init().SaveChanges();

        NavigationService.GoBack();
    }

```

```

    private void Res_Vis_Change(object sender,
DependencyPropertyChangedEventArgs e)
    {
        if (result.id_test != 0)
        {
            EfModel.init().Entry(result).Reload();
        }
    }

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

    }
}

//Страница результаты

public partial class Result_page : Page
{
    public Result_page()
    {
        InitializeComponent();
        UpdateData();
    }
    public void UpdateData()
    {
        List<test> tests = EfModel.init().tests.Where(t =>
t.vrach.Name_vrach.Contains(TB_Search.Text)
    || t.vrach.Surname_vrach.Contains(TB_Search.Text) ||
t.vrach.Patronymic_vrach.Contains(TB_Search.Text)
    || t.pacient.Name_pac.Contains(TB_Search.Text) ||
t.pacient.Surname_pac.Contains(TB_Search.Text)
    || t.vrach.Patronymic_vrach.Contains(TB_Search.Text)
|| t.Data_test.ToString().Contains(TB_Search.Text)).ToList();

        Lv_Res.ItemsSource = tests;
        if (tests.Count <= 0)
        {
            Tb_NoRes.Visibility = Visibility.Visible;
        }
        else
        {
            Tb_NoRes.Visibility = Visibility.Hidden;
        }
    }

    private void Button_Edit_Result_Click(object sender,
RoutedEventArgs e)
    {
        test test = (sender as Button).DataContext as test;
        NavigationService.Navigate(new
Result_add_page(test));
    }

    private void Button_Delete_Result_Click(object sender,
RoutedEventArgs e)
    {
        test test1 = (sender as Button).DataContext as test;
        if (MessageBox.Show("Вы точно хотите удалить запись
?", "Удаление", MessageBoxButton.YesNo) == MessageBoxResult.Yes)
        {
            EfModel.init().tests.Remove(test1);
        }
    }
}

```

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

```

        EfModel.init().SaveChanges();

        UpdateData();
    }
}

private void Button_Add_Result_Click(object sender,
RoutedEventArgs e)
{
    NavigationService.Navigate(new Result_add_page(new
test()));
}

private void Search_Res(object sender,
TextChangedEventArgs e)
{
    UpdateData();
}

private void Res_Vis_Change(object sender,
DependencyPropertyChangedEventArgs e)
{
    UpdateData();
}
}
}

```

Помощь с ВКР под ключ!

//Страница с настройками

пишем-вкр-сами.рф

```

public Setting_page()
{
    InitializeComponent();
    cbColorblind.IsChecked =
Properties.Settings.Default.ColorCb;
    if(Properties.Settings.Default.Language == "ru")
    {
        rbRus.IsChecked = true;
    }

    else
    {
        rbEng.IsChecked = true;
    }
}

private void PropReload() {
    ResourceDictionary dictionary = new
ResourceDictionary();

    if (cbColorblind.IsChecked == true)
    {
        Properties.Settings.Default.ColorCb = true;
    }
}
}

```

```

        dictionary.Source = new
Uri("Themes/Theme_colorblind.xaml", UriKind.Relative);
    }

    else
    {
        Properties.Settings.Default.ColorCb = false;

        dictionary.Source = new
Uri("Themes/Theme_normal.xaml", UriKind.Relative);
    }

    Properties.Settings.Default.Save();

    Application.Current.Resources.MergedDictionaries.Cle
ar();
    Application.Current.Resources.MergedDictionaries.Add
(dictionary);
    if (Properties.Settings.Default.Language == "ru")
    {
        Application.Current.Resources.MergedDictionaries
.Add(new ResourceDictionary { Source = new
Uri("Localisations/Localisation_ru.xaml", UriKind.Relative) });
    }
    else
    {
        Application.Current.Resources.MergedDictionaries
.Add(new ResourceDictionary { Source = new
Uri("Localisations/Localisation.xaml", UriKind.Relative) });
    }
}

private void Cb_Colorblind(object sender,
RoutedEventArgs e)
{
    PropReload();
}

private void Rus_Checked(object sender, RoutedEventArgs
e)
{
    ((App)Application.Current).Set_Cult("ru");
    PropReload();
}

private void Eng_Checked(object sender, RoutedEventArgs
e)
{
    ((App)Application.Current).Set_Cult("en");
    PropReload();
}

```

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

```

    }
}

//Главное меню

public MainWindow()
{
    InitializeComponent();

    ResourceDictionary dictionary = new
ResourceDictionary();

    if (Properties.Settings.Default.ColorCb)
    {
        dictionary.Source = new
Uri("Themes/Theme_colorblind.xaml", UriKind.Relative);
    }
    else
    {
        dictionary.Source = new
Uri("Themes/Theme_normal.xaml", UriKind.Relative);
    }

    Application.Current.Resources.MergedDictionaries.Cle
ar();
    Application.Current.Resources.MergedDictionaries.Add
(dictionary);

    if (Properties.Settings.Default.Language == "ru")
    {
        Application.Current.Resources.MergedDictionaries
.Add(new ResourceDictionary{Source = new
Uri("Localisations/Localisation.ru.xaml", UriKind.Relative)});
    }
    else
    {
        Application.Current.Resources.MergedDictionaries
.Add(new ResourceDictionary{Source = new
Uri("Localisations/Localisation.xaml", UriKind.Relative)});
    }

    FrameNavigations.Navigate(new Autorisation_page());

    FrameNavigations.Navigating +=
FrameNavigations_Navigating; ;
}

private void FrameNavigations_Navigating(object sender,
NavigatingCancelEventArgs e)
{
    if(AuthClass.polzovatel != null)
    {

```

**Помощь с ВКР под ключ!
пишем-вкр-сами.рф**

```

        Button_page_User.Visibility =
AuthClass.polzovatel.IsAdmin?Visibility.Visible :
Visibility.Collapsed;
        Button_page_Connection.Visibility =
AuthClass.polzovatel.IsAdmin ? Visibility.Visible :
Visibility.Collapsed;
    }
}

private void Button_page_Proverka_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new
Proverka_Circl_page());
}

private void Button_page_Vrach_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new Vrach_page());
}

private void Button_page_Pacient_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new Pacient_page());
}

private void Button_page_Result_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new Result_page());
}

private void Button_page_Settings_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new Setting_page());
}

private void Button_page_User_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new Users_page());
}

private void Button_page_Connection_Click(object sender,
RoutedEventArgs e)
{
    FrameNavigations.Navigate(new Connection_page());
}

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

        private void Button_page_Back_Click(object sender,
RoutedEventArgs e)
        {
            if (FrameNavigations.CanGoBack)
            {
                FrameNavigations.GoBack();
            }
        }
    }
}

```

//Страница пользователи

```

public partial class Users_page : Page
{
    public Users_page()
    {
        InitializeComponent();
        Update();
    }

```

```

    public void Update()
    {

```

```

        List<polzovatel> polzovatels =
        db.polzovatels.Where(p=>p.login.Contains(Tb_Search_
        text)).ToList();
        Lv_User.ItemsSource = polzovatels;

```

```

        if (polzovatels.Count <= 0)
        {
            Tb_NoRes.Visibility = Visibility.Visible;
        }

```

```

    else
    {

```

```

        Tb_NoRes.Visibility = Visibility.Hidden;
    }
}

```

```

    private void Button_Add_User_Click(object sender,
RoutedEventArgs e)
    {

```

```

        NavigationService.Navigate(new User_add_page(new
polzovatel()));
    }

```

```

    private void Button_Edit_User_Click(object sender,
RoutedEventArgs e)
    {

```

```

        polzovatel polzovatel = (sender as
Button).DataContext as polzovatel;
        NavigationService.Navigate(new

```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

```

User_add_page(polzovatel));
    }

    private void Button_Delete_User_Click(object sender,
RoutedEventArgs e)
    {
        polzovatel polz = (sender as Button).DataContext as
polzovatel;

        if (MessageBox.Show("Вы точно хотите удалить запись
?", "Удаление", MessageBoxButton.YesNo) == MessageBoxResult.Yes)
        {
            EfModel.init().polzovatels.Remove(polz);
        }

        EfModel.init().SaveChanges();
        Update();
    }

    private void Search_User(object sender,
TextChangedEventArgs e)
    {
        Update();
    }

    private void User_Vis_Change(object sender,
DependencyPropertyChangedEventArgs e)
    {
        Update();
    }
}

```

Помощь с ВКР под ключ! пишем-вкр-сами.рф

Страница добавления пользователей

```

public partial class User_add_page : Page
{
    polzovatel polzovatel;
    public User_add_page(polzovatel polzovatel)
    {
        InitializeComponent();

        this.polzovatel = polzovatel;
        DataContext = polzovatel;
    }

    private void Button_Save_Click(object sender,
RoutedEventArgs e)
    {
        if (polzovatel.id_polz == 0)
        {
            EfModel.init().polzovatels.Add(polzovatel);
        }
    }
}

```

```
        EfModel.init().SaveChanges();
        NavigationService.GoBack();
    }

    private void User_add_vis_change(object sender,
DependencyPropertyChangedEventArgs e)
    {
        if (polzovatel.id_polz != 0)
        {
            EfModel.init().Entry(polzovatel).Reload();
        }
    }
}
```

Помощь с ВКР под ключ!
пишем-вкр-сами.рф

ПРИЛОЖЕНИЕ Д

Диск с материалами

Диск с материалами

Помощь с ВКР под ключ!
пишем-вкр-сами.рф