

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Инженерная школа информационных технологий и робототехники
 Направление подготовки 09.04.03. Прикладная информатика
 Отделение школы (НОЦ) Отделение информационных технологий

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Информационная система обработки медицинских справок о результатах общего анализа крови

УДК 004:612.1-047.44

Студент

Группа	ФИО	Подпись	Дата
8KM71	Богданов Данила Динарович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Аксёнов Сергей Владимирович	к.т.н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор отделения СГН	Сосковец Любовь Ивановна	д.и.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД	Атепаева Наталья Александровна			

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Марухина Ольга Владимировна	к.т.н.		

**Планируемые результаты обучения по направлению подготовки 09.04.03
прикладная информатика**

Код результата	Результат обучения (выпускник должен быть готов)
<i>Профессиональные компетенции</i>	
Р1	Применять базовые и специальные знания в области современных информационно-коммуникационных технологий для решения междисциплинарных инженерных задач.
Р2	Проводить теоретические и экспериментальные исследования, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретацию полученных данных в области информатизации и автоматизации прикладных процессов и создания, внедрения, эксплуатации и управления информационными системами в прикладных областях.
Р3	Внедрять, сопровождать и эксплуатировать современные информационные системы, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья и безопасности труда, выполнять требования по защите окружающей среды.
Р4	Активно владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты инновационной инженерной деятельности.
Р5	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе глобальных компьютерных сетей.
Р6	Эффективно работать индивидуально, в качестве члена и руководителя группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.
Р7	Самостоятельно учиться и непрерывно повышать квалификацию в течение всего периода профессиональной деятельности.
<i>Профиль «Системы корпоративного управления»</i>	
Р8	Применять глубокие профессиональные знания основ построения информационных технологий и систем, достаточные для решения научных и профессиональных задач производства. Знать современные проблемы и методы прикладной информатики и научно-технического развития информационных технологий.
Р9	Ставить и решать задачи комплексного анализа, связанные с информатизацией и автоматизацией прикладных процессов; созданием, внедрением, эксплуатацией и управлением информационными системами в прикладных областях, с использованием базовых и специальных знаний, современных аналитических методов и моделей.

Р10	Организовывать работы по моделированию прикладных информационных систем и реинжинирингу прикладных и информационных процессов предприятия и организации. Управлять проектами по информатизации прикладных задач и созданию информационных систем предприятий и организаций.
-----	--

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа информационных технологий и робототехники
Направление подготовки 09.04.03 Прикладная информатика
Отделение школы (НОЦ) Отделение информационных технологий

УТВЕРЖДАЮ:
Руководитель ООП
_____ Марухина О.В.
(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

Магистерской диссертации

Студенту:

Группа	ФИО
8KM71	Богданов Данила Динарович

Тема работы:

Информационная система обработки медицинских справок о результатах общего анализа крови	
Утверждена приказом директора (дата, номер)	07.03.2019г. №1787

Срок сдачи студентом выполненной работы:	03.06.2019 г.
--	---------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе <i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Объектом исследования являются цифровые обезличенные результаты анализа крови пациентов, которые в дальнейшем будут обрабатываться информационной системой, с целью получения оценки состояния здоровья и визуализации выводов исследования.</p>
--	---

<p>Перечень подлежащих исследованию, проектированию и разработке вопросов <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<p>Создание информационной системы для обработки медицинских справок общего анализа крови Финансовый менеджмент, ресурсоэффективность и ресурсосбережение Социальная ответственность</p>
--	--

<p>Перечень графического материала</p>	<p>Мультимедийная презентация в формате .pptx</p>
---	---

Консультанты по разделам выпускной квалификационной работы

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Сосковец Любовь Ивановна, д. и. н., профессор СГН
Социальная ответственность	Атепаева Наталья Александровна, старший преподаватель ШБИП
Английский язык	Диденко Анастасия Владимировна, к.ф.н., доцент ОИЯ

Названия разделов, которые должны быть написаны на русском и иностранном языках:

Постановка задачи. Обзор состояния проблемы исследования
Разработка методики обработки результатов общего анализа крови для оценки общего состояния здоровья пациента
Разработка программных средств анализа электронных справок результатов ОАК
Тестирование разработанного программного обеспечения на контрольных примерах
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение
Социальная ответственность
Technique development to process the blood test results for general health assessment

<p>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</p>	<p>11.02.2019</p>
--	-------------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Аксёнов Сергей Владимирович	К. Т. Н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8KM71	Богданов Данила Динарович		

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа информационных технологий и робототехники
Направление подготовки 09.04.03 Прикладная информатика
Уровень образования магистратура
Отделение школы (НОЦ) Отделение информационных технологий
Период выполнения весенний семестр 2019 учебного года

Форма представления работы:

Магистерская диссертация

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	03.06.2019
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
22.04.2019	Основная часть	60
24.04.2019	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	20
30.04.2019	Социальная ответственность	20

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Аксёнов Сергей Владимирович	К. Т. Н.		

СОГЛАСОВАНО:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Марухина Ольга Владимировна	К. Т. Н.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8KM71	Богданову Даниле Динаровичу

Школа	Инженерная школа информационных технологий и робототехники	Отделение школы (НОЦ)	Отделение информационных технологий
Уровень образования	Магистратура	Направление/специальность	09.04.03 Прикладная информатика

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:	
1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	1. Стоимость расходных материалов 385,43 руб. 2. Оклад руководителя 33664 руб. 3. Стипендия студента 2477 руб.;
2. <i>Нормы и нормативы расходования ресурсов</i>	Норматив потребления электроэнергии 3,42 руб/кВтч
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	1. Отчисления во внебюджетные фонды 27,1% 2. Районный коэффициент 30% 3. Коэффициент дополнительной заработной платы 15%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	Оценка потенциальных потребителей исследования, анализ конкурентных решений, SWOT-анализ.
2. <i>Разработка устава научно-технического проекта</i>	Планирование этапов разработки программы, определение трудоемкости и построение календарного графика, формирование бюджета, построение диаграммы Ганта.
3. <i>Планирование процесса управления НТИ: структура и график проведения, бюджет и организация закупок</i>	Оценка показателей эффективности исследования.

Перечень графического материала (с точным указанием обязательных чертежей):
1. Анализ конкурентоспособности технических решений; 2. Матрица SWOT; 3. График проведения научно-технического исследования (НТИ); 4. Бюджет научно-технического исследования (НТИ); 5. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования (НТИ).

Дата выдачи задания для раздела по линейному графику	01.03.2019
---	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор отделения СГН	Сосковец Любовь Ивановна	Д. и. н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8KM71	Богданов Данила Динарович		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа		ФИО	
8KM71		Богданову Даниле Динаровичу	
Школа	Инженерная школа информационных технологий и робототехники	Отделение (НОЦ)	Информационных технологий
Уровень образования	Магистратура	Направление/специальность	09.04.03 Прикладная информатика

Тема ВКР:

Информационная система обработки медицинских справок о результатах общего анализа крови	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Рабочим местом является офисное помещение. Технологический процесс представляет собой работу с языком программирования python в программной среде PyCharm. Основным оборудованием, на котором производится работа, является персональный компьютер с периферийными устройствами. Объектом исследования является медицинская информационная система, которая может применяться в таких медицинских учреждениях, как поликлиники, больницы, стационары и медицинские лаборатории.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; организационные мероприятия при компоновке рабочей зоны.	<ul style="list-style-type: none"> – СанПиН 2.2.2/2.4.1340 – 03 – организация рабочих мест с электронно-вычислительными машинами. – ГОСТ 12.2.032-78 ССБТ – организация рабочего места при выполнении работ сидя. – Закон Томской области от 9 июля 2003 года № 83-ОЗ Об охране труда в Томской области (с изменениями на 4 июля 2014 года). Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 01.024.2019)
2. Производственная безопасность:	<ul style="list-style-type: none"> – Отклонение показателей микроклимата – Превышение уровня шума

<p>2.1. Анализ выявленных вредных и опасных факторов при разработке и эксплуатации проектируемого решения;</p> <p>2.2.1 Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека;</p> <p>2.2. Обоснование мероприятий по снижению воздействия.</p>	<ul style="list-style-type: none"> – Отсутствие или недостаток естественного света. – Недостаточная освещенность рабочей зоны – Психофизиологические факторы. – Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека
3. Экологическая безопасность:	Анализ воздействия объекта на литосферу: утилизация отходов, связанные с выходом из строя ПК, люминесцентных ламп и др.
4. Безопасность в чрезвычайных ситуациях:	<p>Возможные чрезвычайные ситуации на объекте:</p> <ul style="list-style-type: none"> – Пожар (наиболее вероятен); <p>Землетрясение.</p>

Дата выдачи задания для раздела по линейному графику	01.03.2019
--	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД ШБИП	Атепаева Наталья Александровна			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8KM71	Богданов Данила Динарович		

РЕФЕРАТ

Учебно-исследовательская работа 129 страниц, 39 рис., 26 таблиц, 25 источников.

Ключевые слова: медицинская информационная система, общий анализ крови, цифровые медицинские справки, парсинг данных, визуализация данных, оценка состояния здоровья пациента, лейкоцитарные индексы.

Предметом исследования являются цифровые обезличенные результаты общего анализа крови пациентов.

Целью данной работы является создание и реализация медицинской информационной системы, которая будет проводить анализирующие результаты анализа крови пациентов с целью оценки состояния здоровья и визуализации выводов исследования.

Обработка результатов анализа крови включает в себя оценки индексов воспаления организма по современным медицинским методикам.

В ходе выполнения работы проводилась разработка медицинской информационной системы виде web-приложения на языке Python 3.6.4 и framework Django 2.1.5.

В первой главе представлена постановка задачи и обзор состояния проблемы.

Во второй главе представлена разработка методики обработки результатов общего анализа крови для оценки общего состояния здоровья пациента.

В третьей главе представлена разработка программных средств анализа электронных справок результатов ОАК

В четвёртой главе представлено тестирование разработанного программного обеспечения на контрольных примерах

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

МИС – медицинская информационная система;

ИС – информационная система;

ОАК – общий анализ крови;

ЛИ – лейкоцитарные индексы;

БД – база данных;

СУБД – система управления базы данных.

ОГЛАВЛЕНИЕ

Введение.....	15
1 постановка задачи. Обзор состояния проблемы исследования.....	18
2 Разработка методики обработки результатов общего анализа крови для оценки общего состояния здоровья пациента	22
2.1 Анализ ключевых данных из медицинских справок.....	22
2.2 Создание файлов для хранения ключевых данных	23
2.3 Регулярные выражения	27
2.3.1 Возможности регулярных выражений.....	27
2.4 Загрузка ключевых данных в массив.....	28
2.5 Извлечение необходимых данных из медсправки.....	28
2.6 Дополнительные скрипты на JavaScript	30
2.6.1 Скрипт 1.....	30
2.6.2 Скрипт 2.....	31
2.6.3 Скрипт 3.....	32
2.7 Разработка базы данных для МИС.....	32
2.7.1 Построение инфологической модели данных.....	33
2.7.1.1 Анализ предметной области	33
2.7.1.2 Инфологическая модель данных	34
2.7.1.3 Проектирование целостности базы данных	36
2.7.1.4 Проектирование базы данных	36
2.7.1.5 Проверка нормализации таблиц.....	40
2.7.2 Построение даталогической модели данных	41
2.8 Лейкоцитарные индексы.....	42
2.8.1 Лейкоцитарный индекс интоксикации (Калиф-Калифа)	43
2.8.2 Индекс сдвига лейкоцитов крови (Н. И. Яблучанский).....	44
2.8.3 Индекс Аллергизации (ИА)	45
3 Разработка программных средств анализа электронных справок результатов ОАК.....	46
3.1 Инструментальные средства разработки, предлагаемые для информационной системы.....	46
3.1.1 Front-end	46
3.1.1.1 Bootstrap v4.0.0.....	47
3.1.2 Back-end.....	47
3.1.2.1 Язык программирования Python.....	47
3.1.2.1.1 Python 3: преимущества и недостатки языка	48
3.1.3 Framework Django	49
3.1.4 СУБД MariaDB.....	50
3.2 Файловая структура МИС.....	50
3.3 Пошаговая разработка МИС.....	51

3.3.1	Разработка основного шаблона сайта.....	52
3.3.2	Постройка графиков с помощью плагина D3.js.....	54
3.3.3	Анализ результатов ОАК по нормам крови.....	56
3.3.4	Разработка голосового интерфейса.....	57
3.3.4.1	Экспериментальная часть разработки голосового интерфейса.....	59
3.3.4.2	Результаты работы в создании голосового интерфейса.....	59
3.3.4.3	Проблема с SSL-сертификатом	60
3.3.4.4	Заключение по голосовому интерфейсу.....	61
3.3.5	Модуль Django-allauth авторизации и регистрации	61
3.4	Краткий заключительный обзор всей МИС.....	63
3.4.1	Функция «Загрузить медсправку»	64
3.4.2	Функция «Список загруженных медсправок»	65
3.4.3	Функция «Написать статью».....	65
3.4.4	Функция «Просмотреть графики»	67
3.4.5	Функция «Создать медсправку»	67
3.4.6	Функция «Список пациентов»	68
3.4.7	Функция «Лейкоцитарные индексы»	69
3.4.8	Личный профиль врача	70
4	тестирование разработанного программного обеспечения на контрольных примерах	71
4.1.1	Результаты тестирования web-приложения	72
5	ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ	
	74	
5.1	Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения	75
5.1.1	Потенциальные потребители результатов исследования	75
5.1.2	Анализ конкурентных технических решений.....	76
5.1.3	SWOT - анализ	78
5.2	Планирование научно-исследовательских работ	78
5.2.1	Структура работ в рамках научного исследования	78
5.2.2	Определение трудоемкости выполнения работ	79
5.2.3	Разработка графика проведения научного исследования	81
5.2.4	Бюджет научно-технического исследования (НТИ)	82
5.2.5	Расчет материальных затрат НТИ.....	82
5.2.6	Основная заработная плата исполнителей темы	82
5.2.7	Дополнительная заработная плата исполнителей темы.....	83
5.2.8	Отчисления во внебюджетные фонды (страховые отчисления).....	84
5.2.9	Накладные расходы	84
5.2.10	Формирование бюджета затрат научно-исследовательского проекта.....	84

5.3	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.....	85
5.4	Общий вывод по разделу	87
6	СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ	88
	Введение.....	88
6.1	Правовые и организационные вопросы обеспечения безопасности.....	88
6.1.1	Организационные мероприятия при компоновке рабочей зоны исследования	89
6.2	Производственная безопасность	90
6.2.1	Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при проведении исследований.....	91
6.2.1.1	Отклонение показателей микроклимата.....	91
6.2.1.2	Отсутствие или недостаток естественного света и недостаточная освещенность рабочей зоны	92
6.2.1.3	Превышение уровня шума.....	94
6.2.1.4	Психофизиологические факторы	95
6.2.2	Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения	96
6.2.2.1	Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека;	96
6.3	Экологическая безопасность	97
6.4	Безопасность в чрезвычайных ситуациях	98
6.4.1	Анализ вероятных ЧС, которые может инициировать объект исследований	98
6.4.2	Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований.....	98
6.4.3	Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС	99
6.5	Выводы по разделу.....	100
	Заключение.....	102
	Список литературы.....	103
	Приложение А.....	106
	Приложение Б	117

ВВЕДЕНИЕ

За последние несколько лет информационные системы стали неотъемлемой частью современного мира. Информационные системы (ИС) используются во многих областях, основанных на информации той или иной области, Например: в Бухгалтерском учете, Документообороте, в медицине и т.д.

Информационной системой называется комплекс, включающий вычислительное и коммуникационное оборудование, программное обеспечение, лингвистические средства и информационные ресурсы, а также системный персонал и обеспечивающий поддержку динамической информационной модели некоторой части реального мира для удовлетворения информационных потребностей пользователей [1].

Современная медицинская клиника обладает лабораториями по анализу биологического материала, включая кровь. Одним из важных инструментов, используемых в исследованиях крови являются анализаторы крови. В эти приборы помещаются пробирки с кровью, которые в дальнейшем подвергаются тщательному анализу. После анализа прибор выдает справку об анализе крови в электронном виде (в этой справке содержатся кол-во лейкоцитов, лимфоцитов, эритроцитов и т.д.). Далее эта справка может быть распечатана и получена врачом или пациентом.

К сожалению, большинство лабораторий не хотят работать со справками в электронном виде, предпочитая работать по старинке. Врачи обычно распечатывают справки, клеивают их в медицинскую карту и после этого выдают эту карту пациенту. Я считаю, что Работа с медицинскими справками в электронном виде имеет массу преимуществ, т.к., во-первых, потому что это экономит такой ресурс, как бумага, во-вторых, это экономит много места в архивах, где хранятся медицинские карты пациентов, и, в-третьих, все эти электронные медсправки можно хранить и обрабатывать, на удаленном сервере базы данных. Благодаря этому врач, не выходя из кабинета, с помощью компьютера, может просмотреть необходимые ему справки и быстрее провести

оценку состояния здоровья или тяжести заболевания пациента, что приводит к экономии его рабочего времени.

В архивах поликлиник и больниц хранятся довольно много бумажных медицинских карт. Эти карты хранят непроанализированную медицинскую информацию. И если мы хотим проанализировать все эти данные нам необходимо перевести эти медкарты в электронный вид и загрузить их БД медицинской информационной системы.

Актуальность работы заключается в том, что каждый производитель медицинского оборудования создаёт свои уникальные анализаторы крови, следовательно, каждый прибор будет распечатывать свою уникальную электронную медсправку по анализу крови. Поэтому необходимо создать такую информационную систему, которая будет все эти медсправки приводить к единому виду (стандарту) и сохранять в БД.

Объектом исследования являются медицинские информационной системы обработки электронных медицинских записей, полученные из “анализаторов крови” или отсканированные медицинские справки. Все справки перед обработкой МИС были обезличены (т.е. удалены ФИО и адрес проживания), что обеспечивает безопасность данных и отвечает закону о защите персональных данных.

Предметом исследования являются методы обработки электронных записей общего анализа крови для оценки состояния пациента.

Целью данной работы заключается в повышении эффективности врачебной работы с помощью разработки медицинской информационной системы обработки электронных записей результатов общего анализа крови

Для достижения поставленной цели необходимо решить следующие задачи:

- Исследовать и проанализировать существующие проблемы исследования;
- Разработать методы извлечения предикторов из электронных записей ОАК

- Программно реализовать медицинскую информационную систему, которая обрабатывает электронные записи ОАК и выполняет оценку воспалительных процессов в организме;

- Разработанную МИС проверить на конкретных тестовых примерах.

Научная новизна заключается в повышении качества оценки тяжести заболевания с помощью разработки информационной системы, анализирующей результаты общего анализа крови пациентов с целью получения оценок состояния здоровья и визуализации выводов исследования. Обработка результатов анализа крови включает в себя расчет индексов воспаления организма по современным медицинским методикам.

1 ПОСТАНОВКА ЗАДАЧИ. ОБЗОР СОСТОЯНИЯ ПРОБЛЕМЫ ИССЛЕДОВАНИЯ

Каждый производитель медицинских аппаратов производят свои уникальные устройства по анализу крови, следовательно, каждый прибор будет выдавать свою уникальную электронную медсправку по анализу крови. Данные приборы представлены на рисунке 1.



Рисунок 1 – Уникальные анализаторы крови от разных производителей (слева анализатор BA 88A Mindray, по середине анализатор, ABL800 FLEX, справа анализатор BC 5800)

В зависимости от стоимости анализаторы могут обладать самыми разными функциями. Например, анализатор BA 88A Mindray обладает открытой системой, которая легко адаптируется к реагентам разных производителей; анализатор ABL800 FLEX обладает возможностью определения до 18 параметров экспресс-диагностики неотложных состояний на основании одного образца крови; анализатор BC 5800 может проводить до 90 тестов в час.

Распечатанные медицинские справки могут быть самых разных форматов, начиная от страницы A4 и заканчивая в виде чека. В рамках магистерской диссертационной работы использовались справки формата A4 и A5 рисунок 2 и рисунок 3.

Пол: Муж
Возраст: 1 год
 Дата взятия образца: 23.01.2016 08:05
 Дата поступления образца: 24.01.2016 07:15
 Дата печати результата: 24.01.2016 12:22



Клинический анализ крови

Исследование	Результат	Единицы	Референсные значения	Комментарий
Гематокрит	32.2	%	32.0 - 40.0	
Гемоглобин	10.7*	г/дл	11.0 - 14.0	
Эритроциты	4.29	млн/мкл	3.80 - 4.80	
MCV (ср. объем эритроц.)	75.1	фл	73.0 - 85.0	
RDW (шир. распр. эритроц.)	15.1*	%	11.6 - 14.8	
МСН (ср. содер. Нв в эр.)	24.9	пг	22.0 - 30.0	
МСНС (ср. конц. Нв в эр.)	33.2	г/дл	32.0 - 38.0	
Тромбоциты	229	тыс/мкл	206 - 445	
Лейкоциты	10.58	тыс/мкл	6.00 - 17.00	
Нейтрофилы (общ.число), %	57.2*	%	29.0 - 54.0	При исследовании крови на гематологическом анализаторе патологических клеток не обнаружено. Количество палочкоядерных нейтрофилов не превышает 6%
Лимфоциты, %	33.5*	%	37.0 - 60.0	
Моноциты, %	7.2	%	3.0 - 10.0	
Эозинофилы, %	1.9	%	1.0 - 7.0	
Базофилы, %	0.2	%	< 1.0	
Нейтрофилы, абс.	6.06	тыс/мкл	1.50 - 8.50	
Лимфоциты, абс.	3.54	тыс/мкл	3.00 - 9.50	
Моноциты, абс.	0.76	тыс/мкл	0.00 - 1.00	
Эозинофилы, абс.	0.20	тыс/мкл	0.00 - 0.70	
Базофилы, абс.	0.02	тыс/мкл	0.01 - 0.07	
СОЭ (по Вестергрену)	6	мм/ч	< 10	

Рисунок 2 – Вид справки, формата А4

ПОКАЗАТЕЛЬ	ЗНАЧЕНИЕ	ЕД. ИЗМ.	НОРМА
Клинический анализ крови			
>!	Лейкоциты	15.1	10E9/л (4.0 - 10.0)
>!	Лимфоциты	11.5	10E9/л (0 - 5)
	Абсолютное значение популяции ср клеток	0.8	10E9/л (0 - 2)
	Гранулоциты	2.8	10E9/л (1 - 8)
	Эритроциты	4.6	10E12/л (3.5 - 5.5)
	Гемоглобин	130.0	г/л (115.0 - 165.0)
	Гематокрит	36.3	% (35 - 55)
	Средний объем эритроцита	78.8	фл (75 - 100)
	Среднее содержание HGB в эритроците	28.3	пг (25.0 - 35.0)
	Средняя концентрация HGB в эритроците	359.0	г/л (310.0 - 380.0)
>	Ширина распределения эритроцита	16.2	% (11 - 16)
	Ширина распределения эритроцита	57.2	фл (30 - 150)
	Тромбоциты	214	10E9/л (180 - 400)
	Средний объем тромбоцитов	8.0	фл (8.0 - 11.0)
	Ширина распределения тромбоцитов	12.7	фл (0.1 - 99.9)
	Тромбокрит	0.13	% (0 - 10)
	Макротромбоциты	19.0	% (0.1 - 99.9)
>!	Процент лимфоцитов	76.3	% (15 - 50)
	Процент популяции ср клеток	4.7	% (2 - 15)
<<	Процент гранулоцитов	19.0	% (35 - 80)
	СОЭ	7	мм/ч

Рисунок 3 – Вид справки, формата А5

Поэтому огромной необходимостью является создание информационной системы, которая будет все эти медицинские справки приводить к единому электронному формату (стандарту) и сохранять в БД.

На рисунке 4 представлены сервисы предлагаемой информационной системы при работе с данными из справок общего анализа крови.



Рисунок 4 – Сервисы разрабатываемой ИС

Одной из главных особенностей информационной системы (помимо обработки и хранения в БД медсправок) заключается в объективной оценке состояния тяжести пациента по показателю крови. Для оценки состояния используется вычисление «Лейкоцитарной формулы». Лейкоцитарная формула – это процентное соотношение разных видов лейкоцитов в периферической крови. В периферической крови встречаются пять популяций лейкоцитов. Нейтрофилы, базофилы, эозинофилы, относящиеся к гранулоцитарному ряду. При исследовании крови на гематологических анализаторах производится автоматический подсчет лейкоцитарной формулы с определением пяти основных популяций лейкоцитов. Технологии подсчета лейкоцитарной формулы различаются у разных производителей [2].

Ниже представлены подробные этапы обработки (парсинга) медицинских справок ОАК:

1. Загрузить ключевые данные из файлов с расширением .dat в массивы программы;
2. Провести сравнительный анализ медицинской справки ОАК с массивами на предмет совпадения ключевых данных;

3. Все обнаруженные совпадения сохранить в два массива. В первый массив сохранить типы клеток крови с их числовыми значениями, во второй массив сохранить все найденные размерности;
4. Два полученных массива сохранить в один общий массив.
5. Полученный массив сохранить в два выходных файла формата .txt и .csv.

Также необходимо отметить, что информационная система разрабатывается специально для Сибирского государственного медицинского университета «Кафедры инфекционных болезней и эпидемиологии».

2 РАЗРАБОТКА МЕТОДИКИ ОБРАБОТКИ РЕЗУЛЬТАТОВ ОБЩЕГО АНАЛИЗА КРОВИ ДЛЯ ОЦЕНКИ ОБЩЕГО СОСТОЯНИЯ ЗДОРОВЬЯ ПАЦИЕНТА

Перед тем, как начать разрабатывать программу по обработке (парсингу) электронных медицинских справок необходимо выбрать язык программирования, на котором в дальнейшем и будет строиться сам парсер.

После долгого размышления было принято решение использовать язык программирования Python.

2.1 Анализ ключевых данных из медицинских справок

Перед тем, как приступить к разработке программы необходимо вручную проанализировать все ключевые данные, которые используются в медицинских справках. То есть, нам необходимо составить обширную базу данных для всех типов клеток и их размерностей. Во многих медицинских учреждениях России, врачи по-разному указывают имена типов клеток и их размерностей в медицинских справках. Например, тип клетки «гемоглобин» может писаться как по-русски, так и по-английски «hemoglobin» или может указываться в аббревиатуре Hgb. Точно также дело обстоит и с размерностями. На примере того же гемоглобина размерность может указываться в г/л, в Gm/100ml и т.д.

В результате анализа всех доступных медицинских справок, был составлен excel файл, в котором записаны всевозможные имена типов клеток крови и их размерности. Рисунок 5.

	B	C	D	E	F	G	H	I	J	K	L	M	N
34	Средняя концентрация гемоглобина в эритроците	MCHC	Средняя концентра	Ср. конц. Hb в эр	mean corpuscular	26-37	г/дл	г/%	g/dl	g/dL	g%, Gms/dl, Gms/%		
35	Ширина распределения эритроцитов по объему, коэффициент	RDW-CV	Ширина распредел	Расп. эрит. по V, RDWc		11,5-14,5	redcelldis %						
36	Ширина распределения объема эритроцитов, стандартное отклонение	RDW-SD	Ширина распредел	Расп. эрит. по V, станд откл		37-54	redcelldis fl		fl	фл			
37	Нормобласты, ядросодержащие эритроциты	nRBC, nucleated red cells, NRCs				0		%					
38	Нормобласты, ядросодержащие эритроциты	nRBC, nucleated red cells, NRCs				0		абс					
39	Нормобласты, ядросодержащие эритроциты	nRBC, nucleated red cells, NRCs				0		кл/100 лейкоц.					
40													
41	ВИДЫ ЛЕЙКОЦИТОВ												
42	Относительные значения	%							Для 3-дифф.	Для 5-дифф.	Микроскопия		
43	Абсолютные значения	#	abs	абс									
44	Относительные значения												
45	Миелобласты	M_BL	Myeloblast			0	%						
46	Промиелоциты	PROM	Promyelocytes			0	%						
47	Миелоциты	MIEL	Myelocytes, myelos			0	%						
48	Метамиелоциты	META	Metamyelocytes, ju	Юные		0	%						
49	Палочкоядерные нейтрофилы	BAN	Band, stabs, bands	Wand-nuclear ne	Палочкоядерные	0-5	%						
50	Сегментоядерные нейтрофилы	SEG	Segmented neutrophils, segs, polymo	Сегментоядерные		40-70	%						
51	Эозинофилы	EO	Eosinophils	EOS, Eos, Eo, eos		0-5	%						
52	Базофилы	BASO	Basophils	BA,BAS, BASO, basos		0-0,5	%						
53	Моноциты	MONO	Monocytes	MO, MON, monos		1-8	%						
54	Лимфоциты	LYMPH	Lymphocytes	LY, LYM, LYMP, LYMF, lymphs		19-37	%						
55	Плазматические клетки	PLASM	Plasmocytes	Plasma cells		0-1	%						
56	Пролимфоциты	Pro_LY	Prolymphocytes			0	%						

Рисунок 5 – Excel файл с ключевыми медицинскими данными

2.2 Создание файлов для хранения ключевых данных

После создания excel файла с ключевыми медицинскими данными, появился другой немаловажный вопрос: «Где и в каком виде хранить ключевые данные, чтобы программа без проблем могла работать с ними?».

Данные лучше хранить в отдельных файлах, нежели создать массив и хранить их в коде программы. Потому, что если мы создадим массив и запишем туда все наши ключевые данные, то размер файла **main_lib.py** увеличится в несколько раз. Из-за этого программа будет работать медленнее, да и разработчику труднее будет редактировать и добавлять новые данные в массив.

Было решено хранить данные на веб-сервере в папке **blood**. Данная папка находится в каталоге **\content\data**. Данная структура файлового дерева содержит в себе смысловую нагрузку. А именно:

- папка **content** – эта папка, где хранятся все текстовые медицинские справки, а также ключевые данные;
- папка **data** – эта папка, где могут храниться имена папок, конкретных ключевых медицинских данных (Например, в папке **blood** хранятся все ключевые данные крови, и т.д.).

Данная реализация файлового дерева обусловлена тем, что данные хранятся на веб-сервере в четком структурированном порядке. Это означает,

что любой разработчик, который будет работать с данной программой, сможет быстро находить нужные файлы.

Было решено разделить типы клеток и размерности друг от друга и хранить в отдельных **dat** файлах. Файлы называются **measureBlood.dat** и **varBlood.dat** рисунок 5. В **varBlood.dat** находятся имена типов клеток крови, а в **measureBlood.dat** находятся размерности крови рисунок 6.

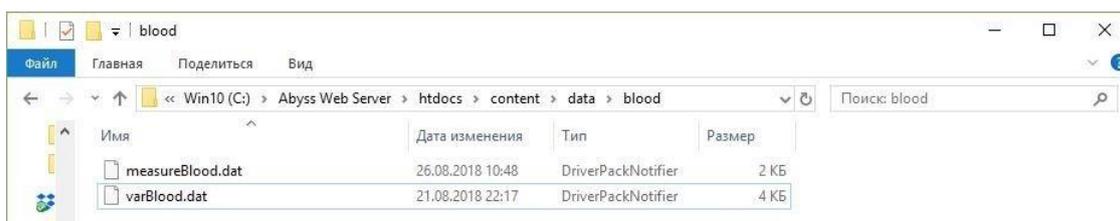


Рисунок 6– Структура хранения файлов в папке «*blood*»

Использование расширения **.dat** было обусловлено двумя причинами. Во-первых, расширение **.dat** несет в себе смысловую нагрузку. Это расширение обозначает, что в данном файле хранятся некоторые необходимые данные, которые используются в программе. Во-вторых, данное расширение предохраняет от случайного изменения данных неопытным пользователем.

Структура хранения ключевых данных внутри файлах с расширением **.dat** объясняется следующим образом. Каждое ключевое значение выделяется квадратными скобками и начинается с новой строки. Это делается для того, чтобы в дальнейшем регулярное выражение могло без проблем (ориентируясь по квадратным скобкам) вытаскивать все значения и записывать их в массив. Также данная структура записей позволяет разработчику легко делать правки и добавлять новые значения рисунок 6, рисунок 7.

```

1 [лейкоциты]
2 [общее количество лейкоцитов]
3 [wbc]
4 [wbcs]
5 [leucocytes]
6 [whitebloodcell]
7 [лимфоциты]
8 [абсолютное значение популяции ср клеток]
9 [гранулоциты]
10 [эритроциты]
11 [общее количество эритроцитов (муж)]
12 [общее количество эритроцитов (жен)]
13 [redbloodcells]
14 [rbc]
15 [rbcs]
16 [eritrocites]
17 [гемоглобин]
18 [гемоглобин (муж)]
19 [гемоглобин (жен)]
20 [гемоглобин (муж)]
21 [гемоглобин (жен)]
22 [hb]
23 [среднее содержание hgb в эритроците]
24 [hgb]
25 [hemoglobin]
26 [гематокрит]
27 [гематокрит (муж)]
28 [ht]
29 [hct]
30 [hematocrit]
31 [гематокрит (жен)]
32 [средний объем эритроцита]
33 [средний объем эритроцитов]
34 [mcv]
35 [ср. объем эритроц.]
36 [сред. объем эритроц.]

```

Рисунок 7– Структура хранения ключевых данных в файле *varBlood.dat*

```

1 [x10?121л]
2 [г/л]
3 [Г/Л]
4 [10^12/л]
5 [x10*9/л]
6 [x10?121л] # русская x
7 [x10?121/л] # русская x
8 [x10*121л] # русская x
9 [x10*121/л] # русская x
10 [x10?121л] # английская x
11 [x10?121/л] # английская x
12 [x10*121л] # английская x
13 [x10*121/л] # английская x
14 [10^9/л]
15 [x10*9/л] # английская x
16 [мм/ч]
17 [пг/кл]
18 [мм/час]
19 [фентолитр]
20 [пикограмм]
21 [абсолютное количество]
22 [относительное количество]
23 [г/л]
24 [г%]
25 [10*12/л]
26 [10*9/л]
27 [мм/ч]
28 [фл]
29 [пг]
30 [nr] # RP иногда неправильно переводит пг
31 [отн]
32 [г/дл]
33 [г/дл] # RP иногда неправильно переводит г/дл
34 [10*12/л]
35 [10*9/л]
36 [fl]

```

Рисунок 8– Структура хранения ключевых данных в файле *measureBlood.dat*

В таблице 1 представлена подробное соотношение типы клеток крови к размерностям.

Таблица 1 – Соотношение типы клеток крови к размерностям

Тип клетки	Единица измерения
Гемоглобин	г/л
Общее количество эритроцитов	10^{12} клеток/л
Общее количество лейкоцитов	10^9 клеток/л
Скорость оседания эритроцитов	мм/час
Гематокрит	%
Ретикулоцит	%
Миелобласты	%
Промиелобласты	%
Миелоциты	%
Метамиелоциты	%
Палочкоядерные нейтрофилы	%
Сегментоядерные нейтрофилы	%
Эозинофилы	%
Базофилы	%
Моноциты	%
Лимфоциты	%
Плазматические клетки	%
Пролимфоциты	%
Лимфобласты	%

2.3 Регулярные выражения

Для того чтобы успешно обрабатывать и извлекать нужные данные из медсправок, необходимо воспользоваться таким инструментом как регулярные выражения.

Регулярные выражения (regular expressions) – это формальный язык поиска и осуществления манипуляций с подстроками в тексте, основанный на использовании метасимволов. Для поиска используется строка-образец, состоящая из символов и метасимволов и задающая правило поиска. Для манипуляций с текстом дополнительно задаётся строка замены, которая также может содержать в себе специальные символы.

2.3.1 Возможности регулярных выражений

Регулярные выражения используются некоторыми текстовыми редакторами и утилитами для поиска и подстановки текста. Например, при помощи регулярных выражений можно задать шаблоны, позволяющие:

- найти все последовательности символов «кот» в любом контексте;
- найти отдельно стоящее слово «кот» и заменить его на «кошка»;
- убрать из текста все предложения, в которых упоминается слово

кот.

Регулярные выражения позволяют задавать и гораздо более сложные шаблоны поиска или замены.

Результатом работы с регулярным выражением может быть:

- проверка наличия искомого образца в заданном тексте;
- определение подстроки текста, которая сопоставляется образцу;
- определение групп символов, соответствующих отдельным частям образца.

Если регулярное выражение используется для замены текста, то результатом работы будет новая текстовая строка, представляющая из себя исходный текст, из которого удалены найденные подстроки (сопоставленные

образцу), а вместо них подставлены строки замены (возможно, модифицированные запомненными при разборе группами символов из исходного текста). Частным случаем модификации текста является удаление всех вхождений найденного образца – для чего строка замены указывается пустой [5].

2.4 Загрузка ключевых данных в массив

Программе (во время работы с медицинскими данными) необходимо иметь загруженные ключевые данные (из файлов с расширением .dat) в массиве. В дальнейшем, с помощью этого массива, программа будет проводить сравнительный анализ ключевых данных с медицинской справкой. А также вытаскивать нужные данные с помощью регулярных выражений.

Были созданы две функции `open_read_varBlood()` и `open_read_measureBlood()` листинг 2.

Функция `open_read_varBlood()` полностью считывает файл `varBlood.dat`. А потом с помощью шаблона `(r'\s*([\w\s\(\)\.|\-%\#]+)\s*)'` загружает нужные данные в массив `varBlood`. Данная функция загружает типы клеток крови.

Функция `open_read_measureBlood()` полностью считывает файл `measureBlood.dat`. А потом с помощью шаблона `(r'\s*([\w\s\(\)\.|\-%\#\^\^*\|=:\?|\+]+)\s*)'` загружает нужные данные в массив `measureBlood`. Данная функция загружает размерности крови.

2.5 Извлечение необходимых данных из медсправки

После загрузки ключевых данных в массивы `measureBlood`, `varBlood`, а также загрузки исходной справки в переменную `content`, необходимо извлечь нужные данные из `content` и записать их в переменные `matches_list` и `res`.

Функция `extract_data_from_blood()` с помощью регулярного выражения `(r'\s*([\r\n](\s*[\r\n](\d*[.,]\d+|\d+))\s*)'` извлекает из медицинской справки названия типов крови, их числовые значения и записывает полученные данные в массив `matches_list` листинг 2.

Функция `extract_measure_blood()` проводит сравнительный анализ медицинской справки с массивом `measureBlood`. Если функция обнаруживает совпадение размерностей между массивом и справкой, то функция записывает совпадающую размерность в массив `res` листинг 2.

Функция `concatenation_result()` объединяет массивы `matches_list` и `res` в единое целое. После соединения массивов, функция сохраняет полученные общие данные справки в два отдельных файла `result.txt` и `result.csv` листинг 2. Полученный результат можно увидеть на рисунке 9 и рисунке 10.

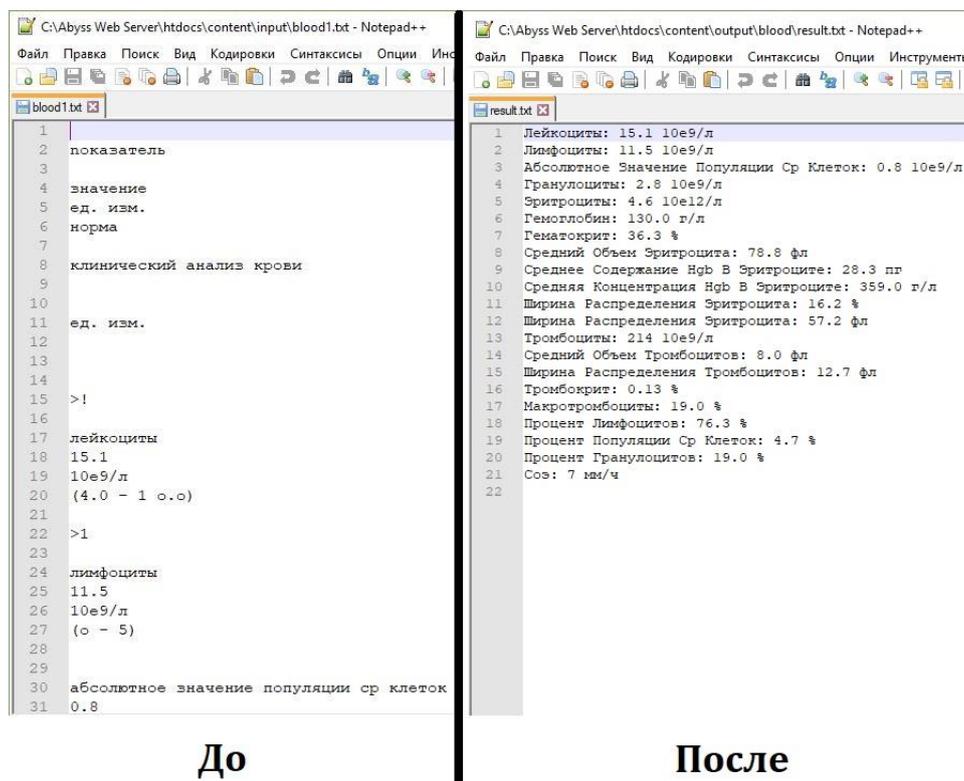


Рисунок 9– Результат обработки медицинской электронной справки в формате txt

	A	B	C	D	E	F
1	Показатель,Значение,ед. измерения					
2	лейкоциты,15.1,10e9/л					
3	лимфоциты,11.5,10e9/л					
4	абсолютное значение популяции ср клеток,0.8,10e9/л					
5	гранулоциты,2.8,10e9/л					
6	эритроциты,4.6,10e12/л					
7	гемоглобин,130.0,г/л					
8	гематокрит,36.3,%					
9	средний объем эритроцита,78.8,фл					
10	среднее содержание hgb в эритроците,28.3,пг					
11	средняя концентрация hgb в эритроците,359.0,г/л					
12	ширина распределения эритроцита,16.2,%					
13	ширина распределения эритроцита,57.2,фл					
14	тромбоциты,214,10e9/л					
15	средний объем тромбоцитов,8.0,фл					
16	ширина распределения тромбоцитов,12.7,фл					
17	тромбокрит,0.13,%					
18	макротромбоциты,19.0,%					
19	процент лимфоцитов,76.3,%					
20	процент популяции ср клеток,4.7,%					
21	процент гранулоцитов,19.0,%					
22	соз,7,мм/ч					
23						

Рисунок 10– Результат обработки медицинской электронной справки в формате CSV

2.6 Дополнительные скрипты на JavaScript

Чтобы полностью исключить случайный фактор загрузки неправильных данных на веб-сервер, было решено создать дополнительные сценарии на JavaScript. Эти сценарии проверяют на стороне клиента загруженные данные по некоторым критериям. Если загруженные данные проходят по этим критериям, то данные отправляются на веб-сервер. Если загруженные данные не проходят хоть по одному критерию, то браузер предупреждает пользователя, что он загрузил некорректные данные и не отправляет данные на веб-сервер.

По результатам исследований были написаны и опубликованы несколько статей [23] [24].

2.6.1 Скрипт 1

Если пользователь забыл загрузить текстовый файл и пытается отправить пустую форму, то программа выводит модальное окно, в котором говорится, чтобы пользователь выбрал файл для загрузки. На рисунке 10 показано выполнение функции **empty_form()**.

```
function empty_form(){ // функция проверяет на наличие пустой формы
    var txt = document.getElementById('file-chooser').value;
```

```

    if (txt == '') {
        alert('Выберите файл для загрузки');
        return false;
    }
    return true;
}

```

Скрипт функции `empty_form()`

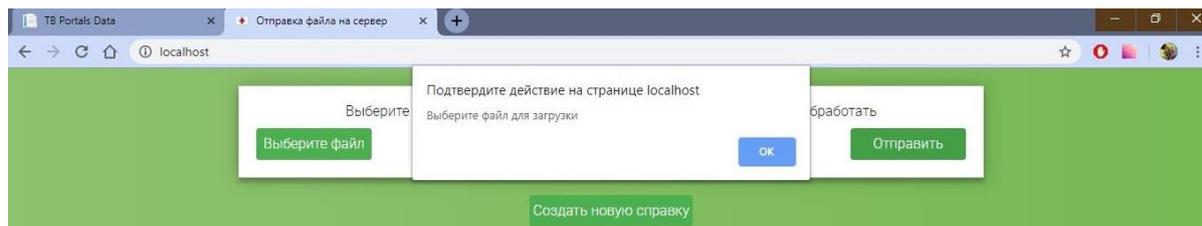


Рисунок 11– Результат работы функции `empty_form()`

2.6.2 Скрипт 2

Если пользователь загружает другой файл (не текстовый) и пытается отправить файл на веб-сервер, то программа выводит модальное окно, в котором говорится, чтобы пользователь выбрал файл с расширением `.txt` листинг 1. На рисунке 11 показано выполнение функции `validateTxt()`.

```

function validateTxt(){ // функция проверяет на наличие расширения .txt

    var uploader = document.getElementById("file-chooser");

    var fileUrl = uploader.value;

    var parts, ext = ( parts = fileUrl.split("/").pop().split(".") ).length >
1 ? parts.pop() : "";

    if (ext !== 'txt'){ //очищает заполненное поле ввода типа file, если
загружен файл не с расширением .txt

        alert( "ОШИБКА. Файл должен быть с расширением .txt Например:
Имя_файла.txt ");

        $("#form-upload")[0].reset();

    }

}

```

Скрипт функции `validateTxt()`

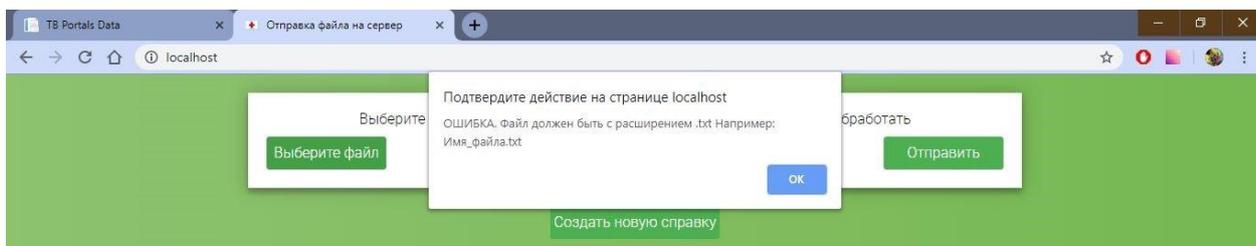


Рисунок 12– Результат работы функции **validateTxt()**

2.6.3 Скрипт 3

Если пользователь загружает пустой текстовый файл и пытается отправить данный файл на веб-сервер, то программа выводит модальное окно, в котором говорится, чтобы пользователь загрузил не пустой текстовый файл
 листинг 1. На рисунке 12 показано выполнение функции **empty_file()**.

```
function empty_file() { // функция проверяет на пустоту текстового файла
    var fileInput = $("#file-chooser")[0];
    var size = Number(fileInput.files[0].size); // Size returned in bytes.
    if (size == 0) {
        alert("ОШИБКА. Вы пытаетесь загрузить пустой txt файл. Пожалуйста, загрузите текстовый файл медицинской справки");
        $("#form-upload")[0].reset();
        return false;
    }
    return true;
}
```

Скрипт функции **empty_file()**

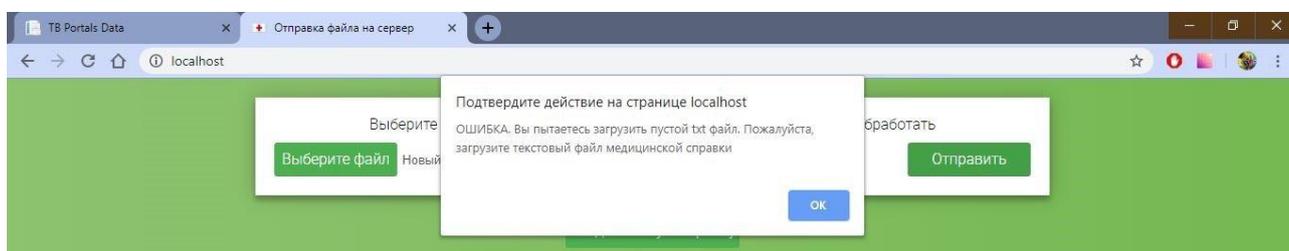


Рисунок 13– Результат работы функции **empty_file()**

2.7 Разработка базы данных для МИС

База данных — это совокупность данных, организованных в соответствии с концептуальной структурой, описывающей характеристики этих

данных и взаимоотношения между ними, причём такое собрание данных, которое поддерживает одну или более областей применения [6].

Вся информация БД хранится в одной или нескольких таблицах. Любая таблица с данными состоит из списка одинаковых записей, расположенных последовательно друг за другом. Они представляют собой строки таблицы, которые можно изменять (UPDATE), удалять (DELETE) или добавлять (INSERT).

Любая запись хранящиеся в БД является комплектом именованных полей, которые могут хранить разную информацию. Например, целочисленное значение, вещественное значение, текстовое сообщение и т. д. Однотипные поля разных записей складывают столбец таблицы.

Как только будет создана одна таблица, мы получаем полноценную базу данных. Но в реальности структуры БД, а именно способы их создания, намного сложнее.

2.7.1 Построение инфологической модели данных

Первым этапом в процессе проектирования базы данных является построение инфологической модели. Цель инфологического моделирования – обеспечение наиболее естественных для человека способов сбора и представления той информации, которую предполагается хранить в создаваемой базе данных. В этом разделе анализируется предметная область и на основе полученных данных строится инфологическая модель будущей базы данных.

2.7.1.1 Анализ предметной области

База данных предназначена в основном для хранения историй болезней пациентов и их медицинских справок ОАК. То есть, описание анамнеза жизни, анамнеза ВТЭ, диагноз при поступлении, количество лейкоцитов в крови, количество эритроцитов в крови и т.д. и т.п. Также в базе данных содержится информация об пациентах и врачах (ФИО, Пол, дата рождения, мобильный телефон, место жительства и т.д.).

В результате анализа в данной предметной области можно выделить следующие сущности:

1. Данные пациентов;
2. Данные врачей;
3. Диагнозы;
4. Истории болезней;
5. Медицинские справки;
6. Результаты медицинских справок;
7. Типы клеток;
8. Размерности клеток;
9. Нормы крови.

2.7.1.2 Инфологическая модель данных

Анализ определённых выше сущностей проектируемой базы данных поможет построить ее инфологическую модель.

1. Данные пациентов (**ID пациента**, Фамилия, Имя, Отчество, Электронная почта, Дата рождения, Пол, Домашний телефон, Мобильный телефон, фотография пациента, домашний адрес).

2. Данные врачей (**ID пациента**, Фамилия, Имя, Отчество, Электронная почта, Дата рождения, Пол, Домашний телефон, Мобильный телефон, фотография пациента, домашний адрес).

3. Диагнозы (**ID диагноз**, имя диагноза, описание).

4. Истории болезней [**ID пациента**, **ID доктора**, **ID диагноза**] (**ID истории болезни**, Дата и время осмотра, Жалобы, Анамнез болезни, Анамнез жизни, Анамнез ВТЭ, Объективный статус, Локальный статус, Диагноз при поступлении, Обоснование диагноза, Диагноз).

5. Медицинские справки [**ID истории болезни**, **ID пациента**] (**ID медицинской справки**, Индекс интоксикации, Индекс сдвига лейкоцитов крови, Индекс алергизации, Дата и время создания медицинской справки, Дата и время создания электронной медицинской справки).

6. Результаты медицинских справок [**ID медицинской справки**, **ID типа клеток**, **ID размерности клетки**] (Числовое значение клетки).

7. Типы клеток (**ID типа клеток**, Тип клетки, Развернутое имя типа клетки, Описание типа клетки).

8. Размерности клеток (**ID размерности клетки**, Имя размерности клетки, Описание размерности клетки).

9. Нормы крови [**ID типа клеток справки**, **ID размерности типа клеток**] (Мужская норма крови, Женская норма крови).

На рисунке 14 изображена инфологическую модель на языке «Таблицы-связи»:

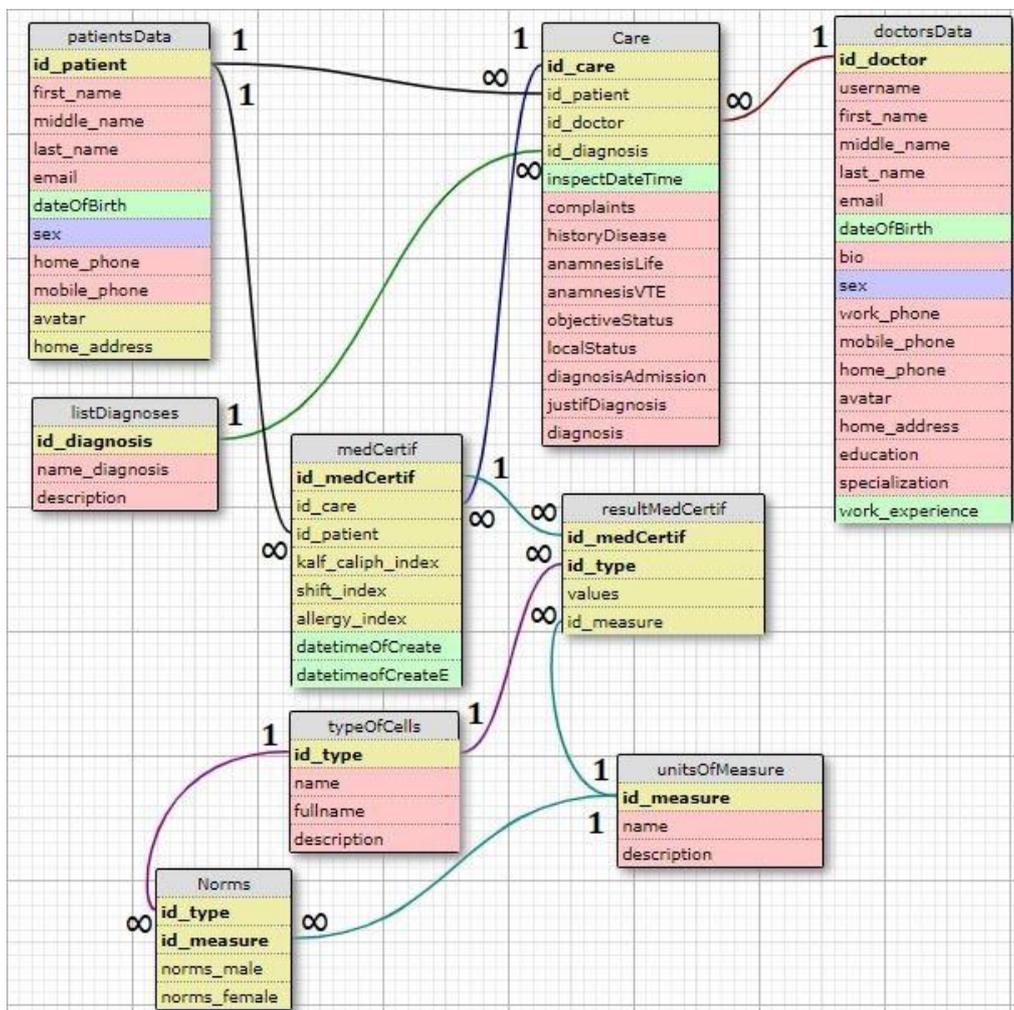


Рисунок 14– Инфологическая модель базы данных МИС, построенная с помощью языка "Таблицы-связи"

2.7.1.3 Проектирование целостности базы данных

Целостность данных означает систему правил, используемых для поддержания связей между записями в связанных таблицах, а также для обеспечения защиты от случайного удаления или изменения связанных данных [7].

Так, в проектируемой базе данных условия целостности соблюдаются следующим образом:

- При изменении значений полей первичного ключа в родительском отношении автоматически осуществляется каскадное изменение всех соответствующих значений в дочернем отношении;
- При удалении записи в родительском отношении автоматически выполняется каскадное удаление всех записей из дочернего отношения, связанных с удаляемой записью. При выполнении операции каскадирования в дочерние сущности вместо старых записей устанавливается значение null.

2.7.1.4 Проектирование базы данных

В соответствие с операцией проектирования БД каждая из полученных сущностей должна быть представлена базовой таблицей. Эти таблицы можно описать следующим образом:

Во таблице 2 и таблице 3 представлены все возможные типы данных у пациентов и врачей.

Таблица 2 – Данные пациентов

Первичные ключи	Поле	Тип данных
ПК	ID пациента	Целое число
	Фамилия	Текст
	Имя	Текст
	Отчество	Текст
	Электронная почта	Текст
	Дата рождения	Дата

	Пол	Целое число
	Домашний телефон	Целое число
	Мобильный телефон	Целое число
	Фотография пациента	Текст
	Домашний адрес	Текст

Таблица 3 – Данные врачей

Первичные ключи	Поле	Тип данных
ПК	ID пациента	Целое число
	Фамилия	Текст
	Имя	Текст
	Отчество	Текст
	Электронная почта	Текст
	Дата рождения	Дата
	Пол	Целое число
	Домашний телефон	Целое число
	Мобильный телефон	Целое число
	Фотография пациента	Текст
	Домашний адрес	Текст
	Поле	Тип данных
	ID отца	Целое число
	Фамилия	Текст
	Имя	Текст
	Отчество	Текст
	Мобильный телефон	Целое число
	Дата рождения	Дата/время

В таблице 4 представлены типы данных диагнозов

Таблица 4 – Диагнозы

Первичные ключи	Поле	Тип данных
ПК	ID диагноз	Целое число
	Имя диагноза	Текст
	Описание	Текст

В таблице 5 представлена структура хранения данных в истории болезней.

Таблица 5 – Истории болезней

Первичные ключи	Поле	Тип данных
ПК	ID пациента	Целое число
	ID доктора	Целое число
	ID диагноза	Целое число
	ID истории болезни	Целое число
	Дата и время осмотра	Дата/время
	Жалобы	Текст
	Анамнез болезни	Текст
	Анамнез жизни	Текст
	Анамнез ВТЭ	Текст
	Объективный статус	Текст
	Локальный статус	Текст
	Диагноз при поступлении	Текст
	Обоснование диагноза	Текст
	Диагноз	Текст

В таблице 6 и таблице 7 представлены типы данных медицинских справок и их результаты.

Таблица 6 – Медицинские справки

Первичные ключи	Поле	Тип данных
ПК	ID истории болезни	Целое число
	ID пациента	Целое число
	ID медицинской справки	Целое число
	Индекс интоксикации	Вещественное число
	Индекс сдвига лейкоцитов крови	Вещественное число
	Индекс аллергизации	Вещественное число
	Дата и время создания медицинской справки	Дата/время
	Дата и время создания электронной медицинской справки	Дата/время

Таблица 7 – Результаты медицинских справок

Первичные ключи	Поле	Тип данных
ПК	ID медицинской справки	Целое число
ПК	ID типа клеток	Целое число
	ID размерности клетки	Целое число
	Числовое значение клетки	Вещественное число

В таблице 8 представлена структура хранения данных в таблице «Типы клеток».

Таблица 8 – Типы клеток

Первичные ключи	Поле	Тип данных
ПК	ID типа клеток	Целое число
	Тип клетки	Текст

В таблице 9 показано хранение данных размерностей клеток.

Таблица 9 – Размерности клеток

Первичные ключи	Поле	Тип данных
ПК	ID размерности клетки	Целое число
	Имя размерности клетки	Текст
	Описание размерности клетки	Текст

В таблице 10 представлена структура хранения данных в таблице «Нормы крови».

Таблица 10 – Нормы крови

Первичные ключи	Поле	Тип данных
ПК	ID типа клеток справки	Целое число
ПК	ID размерности типа клетки	Целое число
	Мужская норма крови	Вещественное число
	Женская норма крови	Вещественное число

2.7.1.5 Проверка нормализации таблиц

На основе анализа предметной области и полученных сущностей необходимо провести нормализацию этих сущностей и привести их к 3 нормальной форме (ЗНФ).

Проверка показала, что:

1. Сущности «Данные пациентов», «Данные врачей», «Диагнозы», «Типы клеток», «Размерности клеток» автоматически нормализованы, потому что они имеют несоставной первичный ключ и единственное не ключевое поле.

2. Сущности «Истории болезней», «Медицинские справки», «Результаты медицинских справок», «Нормы крови», показали, что в они не

имеют функциональных связей между неключевыми полями. Данные поля находятся в полной функциональной зависимости от ключа.

2.7.2 Построение даталогической модели данных

Оператор CREATE TABLE создает таблицу с заданным именем в текущей базе данных. Если нет активной текущей базы данных или указанная таблица уже существует, то возникает ошибка выполнения команды [8].

Ключевое поле — это одно или несколько полей, комбинация значений которых однозначно определяет каждую запись в таблице. Если для таблицы определены ключевые поля, то СУБД предотвращает дублирование или ввод пустых значений в ключевое поле. Ключевые поля используются для быстрого поиска и связи данных из разных таблиц при помощи запросов [9].

PRIMARY KEY может быть составным. Для создания составного первичного ключа, необходимо использовать синтаксис PRIMARY KEY(Имя_столбца_1, Имя_столбца_2, ...).

Так как я использовал framework Django для создания МИС, то БД была описана в виде модельного представления. Модель – это суть представления базы данных, ее таблиц, полей, индексов и связей в терминологии языка программирования, на котором пишется данное приложение листинг 4, листинг 5 (подробнее об этом шаблоне проектирования будет описан в 3 главе) [10].

К сожалению модельное представление Django не умеет создавать таблицы с составными ключами, поэтому было решено создать таблицы «Результаты медицинских справок» и «Нормы крови» с помощью SQL-запросов листинг 6.

```
CREATE TABLE medinfodb.patientslist_resultmedcertif (  
    id int(11) NOT NULL,  
    value varchar(30) DEFAULT NULL,  
    measure_id_id int(11) NOT NULL,  
    type_id_id int(11) NOT NULL,  
    PRIMARY KEY (id, type_id_id)  
)
```

```
CREATE TABLE medinfodb.patientslist_norms (  
    type_id int(11) NOT NULL,  
    norms_male varchar(40) DEFAULT NULL,  
    norms_female varchar(40) DEFAULT NULL,  
    measure_id_id int(11) NOT NULL,  
    PRIMARY KEY (type_id, measure_id_id)  
)
```

Sql-запросы для создания таблиц *resultmedcertif* и *norms*

2.8 Лейкоцитарные индексы

Главная особенность информационной системы (помимо обработки и хранения в БД медсправок) заключается в объективной оценке состояния тяжести пациента по показателю крови. Для оценки состояния используется вычисление «Лейкоцитарной формулы» (Лейкоцитарные индексы). Лейкоцитарная формула – процентное соотношение различных видов лейкоцитов в периферической крови. В периферической крови встречаются пять популяций лейкоцитов. Нейтрофилы, базофилы, эозинофилы, относящиеся к гранулоцитарному ряду (их цитоплазма зерниста, содержит большое количество гранул, включающих, например, миелопероксидазу, эластазу, лизоцим); моноциты и лимфоциты (В-клетки, Т-клетки). При исследовании крови на гематологических анализаторах производится автоматический подсчет лейкоцитарной формулы с определением пяти основных популяций лейкоцитов. Технологии подсчета лейкоцитарной формулы различаются у разных производителей.

Есть много разных Лейкоцитарных индексов с их клиническими значениями. Благодаря этим индексам можно определить, в каком состоянии сейчас находится пациент. Также, если у пациента несколько раз делали анализ крови во время лечения болезни, то информационная система может составить оценку эффективности лечения данного пациента на основе лейкоцитарных индексов. Также для наглядности можно составить график оценки

эффективности лечения. Благодаря этой оценке, врачи могут создать быструю и эффективную стратегию по лечению той или иной болезни. Данную стратегию лечения можно использовать на пациентах с идентичной болезнью.

2.8.1 Лейкоцитарный индекс интоксикации (Калиф-Калифа)

ЛИИ – числовой показатель, который отражает степень интоксикации при заболевании, а также остроту воспалительной реакции. В определенной степени он показывает адаптационные силы всего организма.

Во врачебной практике используется для:

- оценочной характеристики состояния организма при патологии; оценки общего состояния иммунной системы;
- определения сопротивляемости организма к заболеванию;
- выявления реактивной способности иммунитета (силы и скорости ответной реакции на внедрение внешнего антигена) [11].

Выражается формулой:

$$\frac{(4\text{Миело}\% + 3\text{Ю}\% + 2\text{ПЯ}\% + 1\text{СЯ}\%) \times (\text{Плазм}\% + 1)}{(\text{Моно}\% + \text{Лимфо}\%) \times (\text{Эо}\% + 1)}, \quad (2.1)$$

где, Миело – миелоциты, %;

Ю – юные, %;

ПЯ – палочкоядерные, %;

СЯ – сегментоядерные, %;

Плазм – плазматические клетки Тюрка, %;

Моно – моноциты, %;

Лимфо – лимфоциты, %;

Эо – эозинофилы, %.

Нормы интоксикации:

1±0,6 – нормальная величина;

5-8 – начальная интоксикация;

- 10-15 – умеренная интоксикация;
- 18-25 – выраженная интоксикация;
- >25 – терминальная интоксикация.

2.8.2 Индекс сдвига лейкоцитов крови (Н. И. Яблучанский)

ИСЛК – это отношение суммы эозинофилов, базофилов, палочкоядерных нейтрофилов и сегментоядерных нейтрофилов к сумме моноцитов и лимфоцитов.

ИСЛК не зависит от общего числа лейкоцитов крови. Повышение ИСЛК свидетельствует об активном воспалительном процессе и нарушении иммунологической реактивности. Его увеличение связано со снижением числа эозинофилов и повышением количества палочко- и сегментоядерных нейтрофилов. При адекватном лечении этот индекс снижается, высокие показатели зафиксированы у умерших больных. ИСЛК является маркером реактивности организма при остром воспалительном процессе [22].

Выражается формулой:

$$\frac{\text{Эо}\% + \text{Баз}\% + \text{ПЯ}\% + \text{СЯ}\%}{\text{Лимфо}\% + \text{Моно}\%}, \quad (2.2)$$

где Эо – эозинофилы, %;

Баз – базофилилы, %;

ПЯ – палочкоядерные, %;

СЯ – сегментоядерные, %;

Лимфо – лимфоциты, %;

Моно – моноциты, %.

Норма ИСЛК составляет $1,96 \pm 0,56$ и не зависит от общего числа лейкоцитов крови. Показатели указанной формулы можно выразить и в абсолютных значениях [12].

2.8.3 Индекс Аллергизации (ИА)

ИА – это отношение суммы лимфоцитов и эозинофилов к остальным клеткам белой крови.

Выражается формулой:

$$\frac{\text{Лимфо}\% + 10 \times (\text{Эо}\% + 1)}{\text{ПЯ}\% + \text{СЯ}\% + \text{Моно}\% + \text{Баз}\%} \quad (2.3)$$

где Лимфо – лимфоциты, %;

Эо – эозинофилы, %;

ПЯ – палочкоядерные, %;

СЯ – сегментоядерные, %;

Моно – моноциты, %;

Баз – базофилы, %.

Норма ИА составляет 0,79–1,08 [13].

3 РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ АНАЛИЗА ЭЛЕКТРОННЫХ СПРАВОК РЕЗУЛЬТАТОВ ОАК

После долгого размышления было решено разработать МИС в виде web-приложения. Чтобы web-приложения заработала ее необходимо разместить на web-сервере, который в свою очередь должен быть установлен на серверной машине и иметь выход во всемирную паутину. Почему же было отдано предпочтение именно web-приложению, нежели desktop приложению? Потому что после переноса программы на веб-сервер, она автоматически становится кроссплатформенной (т.е. не зависит от конкретной операционной системы) и ей не требуется установка дополнительного программного обеспечения. Для удаленного подключения к web-приложению компьютеру требуется только наличие веб-браузера и доступ к интернету.

Следовательно, для разработки web-приложения нам необходимо выбрать стек технологий для front-end и back-end на котором будет работать данное приложение.

3.1 Инструментальные средства разработки, предлагаемые для информационной системы

3.1.1 Front-end

Front-end — это клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса.

В общем случае, front-end базируется на трёх фундаментальных технологиях. В данном случае web-приложение не исключение. Поэтому front-end данной программы будет реализован на стеке технологий HTML5, CSS3 и JavaScript.

Кроме этих фундаментальных технологий, будут задействованы несколько плагинов, которые написаны на таком языке программирования, как JavaScript. Например, Bootstrap v4.0.0.

3.1.1.1 Bootstrap v4.0.0

Bootstrap - это framework с открытым исходным кодом для разработки с помощью HTML, CSS и JS. Благодаря этому плагину, можно использовать переменные Sass и миксины, гибкую систему сеток, множество готовых компонентов и мощных плагинов, основанных на jQuery [14].

Остальные используемые плагины будут описаны дальше в этой главе.

3.1.2 Back-end.

Back-end — это программно-аппаратная часть сервиса.

В предыдущей главе уже говорилось, что основным языком программирования будет использован Python. Этот язык может создавать не только desktop приложения, но и web-приложения тоже. Следовательно, Python будет задействован в бэкенде.

Но на сегодняшний день, трудно найти людей, которые разрабатывают web-приложения на чистом языке программирования Python, т.к. это требует много затрат во времени, денег и сил разработчиков. Поэтому чтобы ускорить разработку разных проектов и сэкономить усилия разработчиков, поверх Python устанавливают разные популярные фреймворки. Фреймворк — это программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. Эта платформа подходит для создания web-приложений, бизнес-приложений и веб-сервисов. Мной было принято решение использовать такой популярный framework, как Django.

3.1.2.1 Язык программирования Python

Python - высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Python — активно развивающийся язык программирования, новые версии (с добавлением/изменением языковых свойств) выходят примерно раз в два с половиной года [3].

Данный язык программирования - это мощный инструмент для создания программ самого разнообразного назначения, доступный даже для новичков. С его помощью можно решать задачи различных типов. Также Python очень популярен и имеет большую аудиторию. Именно поэтому мое предпочтение остановилось на языке Python.

3.1.2.1 Python 3: преимущества и недостатки языка

Язык Python обладает некоторыми примечательными особенностями, которые обуславливают его широкое распространение. Поэтому следует рассказать о его достоинствах и недостатках.

1. Python - интерпретируемый язык программирования. С одной стороны, это позволяет значительно упростить отладку программ, с другой - обуславливает сравнительно низкую скорость выполнения.

2. Динамическая типизация. В Python не надо заранее объявлять тип переменной, что очень удобно при разработке.

3. Хорошая поддержка модульности. Вы можете легко написать свой модуль и использовать его в других программах.

4. Встроенная поддержка Unicode в строках. В Python необязательно писать всё на английском языке, в программах вполне может использоваться ваш родной язык.

5. Поддержка объектно-ориентированного программирования. При этом его реализация в Python является одной из самых понятных.

6. Автоматическая сборка мусора, отсутствие утечек памяти.

7. Интеграция с C/C++, если возможностей Python недостаточно.

8. Понятный и лаконичный синтаксис, способствующий ясному отображению кода. Удобная система функций позволяет при грамотном подходе создавать код, в котором будет легко разобраться другому человеку в случае необходимости. Также вы сможете научиться читать программы и модули, написанные другими людьми.

9. Огромное количество модулей, как входящих в стандартную поставку Python 3, так и сторонних. В некоторых случаях для написания

программы достаточно лишь найти подходящие модули и правильно их скомбинировать. Таким образом, вы можете думать о составлении программы на более высоком уровне, работая с уже готовыми элементами, выполняющими различные действия.

10. Кроссплатформенность. Программа, написанная на Python, будет функционировать совершенно одинаково вне зависимости от того, в какой операционной системе она запущена. Отличия возникают лишь в редких случаях, и их легко заранее предусмотреть благодаря наличию подробной документации [4].

3.1.3 Framework Django

Django - это фреймворк, который разработан именно для веб-приложений на языке Python. Один из главных принципов фреймворка - DRY (don't repeat yourself). Вебсайты, как правило на Django, строятся из одного или нескольких приложений, которые советуется делать независимыми друг от друга и подключаемыми. Это одно из сильных архитектурных отличий этого фреймворка от других (например, Ruby on Rails). Также, в отличие от многих других фреймворков, обработчики URL в Django конфигурируются явно (при помощи регулярных выражений), а не автоматически задаются из структуры контроллеров.

Django был спроектирован для работы под управлением Apache (с модулем `mod_python`) и с использованием PostgreSQL в качестве базы данных. Для работы с базой данных Django использует собственный ORM, в котором модель данных описывается классами Python, и по ней генерируется схема базы данных.

Архитектура паттерного проектирования Django похожа на «Модель-Представление-Контроллер» (MVC). Контроллер классической модели MVC примерно соответствует уровню, который в Django называется Представление (View), а презентационная логика Представления реализуется в Django уровнем Шаблонов (Templates). Из-за этого архитектуру Django часто называют «Модель-Шаблон-Представление» (MTV) [15].

3.1.4 СУБД MariaDB

В качестве хранилища всей информации МИС (хранение таких данных, как справки ОАК, истории болезней, БД пациентов и т.д.) была выбрана система управления базами данных MariaDB.

MariaDB – это ответвление от системы управления базами данных MySQL, разрабатываемое сообществом под лицензией GNU GPL (т.е. на лицензии свободного программного обеспечения). Толчком к созданию стала необходимость обеспечения свободного статуса СУБД, в противовес политике лицензирования MySQL компанией Oracle. Система лицензирования MariaDB обязывает участников, желающих добавить свой код в основную ветку СУБД, обмениваться своими авторскими правами с MariaDB Foundation для охраны лицензии и возможности создавать критические исправления для MySQL.

MariaDB поддерживает высокую совместимость с MySQL, обеспечивая точное соответствие с API и командами MySQL [16].

3.2 Файловая структура МИС

После того, как мы определились со стеком технологий, необходимо продумать основную структуру хранения данных в МИС MedInfo рисунок 15.

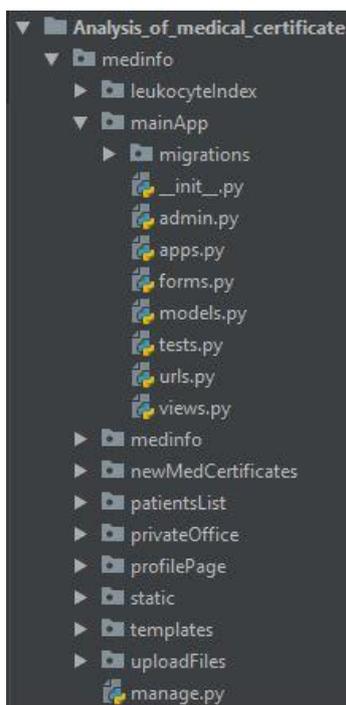


Рисунок 15– Файловая структура проекта MedInfo

Данная структура грамотно продумана по паттерному проектированию «MTV».

Все шаблоны, а именно файлы формата html хранятся в каталоге «**templates**». Файлы формата css и js хранятся в файле «**static**», также в данной папке хранятся все картинки формата jpeg и png.

Каждая папка является отдельным приложением, которые подключены к общему проекту «**medinfo**». Каждое приложение имеет примерное одинаковую структуру (согласно паттерному проектированию). В таблице 11 представлены основные файлы, которые должны быть в каждом приложении проекта.

Таблица 11 – Описание основных файлов приложения

Имена файлов	Описание
views.py	Данный файл представляет главную часть приложения, выполняющий непосредственную обработку данных.
models.py	Данный файл представляет БД, её таблицы, поля индексы и связей в терминологии языка программирования, на котором пишется данное приложение.
urls.py	Данный файл хранит сведения о привязке приложений к интернет-адресам.
admin.py	Данный файл хранит код, задающий параметры административного приложения, что входит в состав Django.
settings.py	Данный файл хранит настройки проекта в виде набора переменных.

3.3 Пошаговая разработка МИС

Перед тем как начать разрабатывать информационную систему было решено дать ей название. В результате размышлений было решено выбрать

название MedInfo, что расшифровывается, как medical information system (рус. Медицинская информационная система).

3.3.1 Разработка основного шаблона сайта

Благодаря HTML5, CSS3, JavaScript (также благодаря библиотеке JQuery) и фреймворку Bootstrap была разработана главная страница сайта МИС рисунок 16 листинг 7.

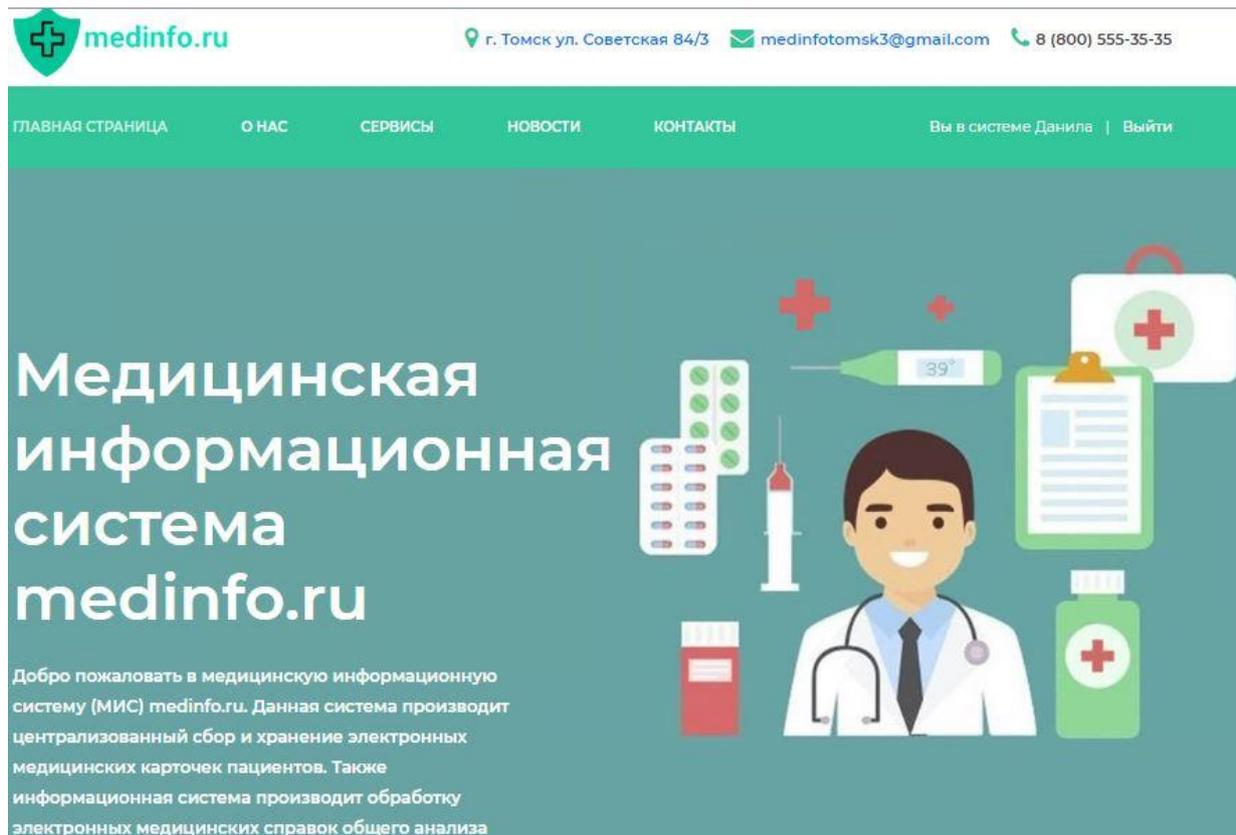


Рисунок 16– Главная страница сайта MedInfo

На основе разработанного дизайна главной страницы сайта были сделаны и другие страницы МИС рисунок 17 и рисунок 18.

При помощи Django был разработан блог новостей. Благодаря этому блогу, врачи могут создавать свои уникальные научные статьи на медицинскую тематику и обмениваться ими между своими коллегами рисунок 17.

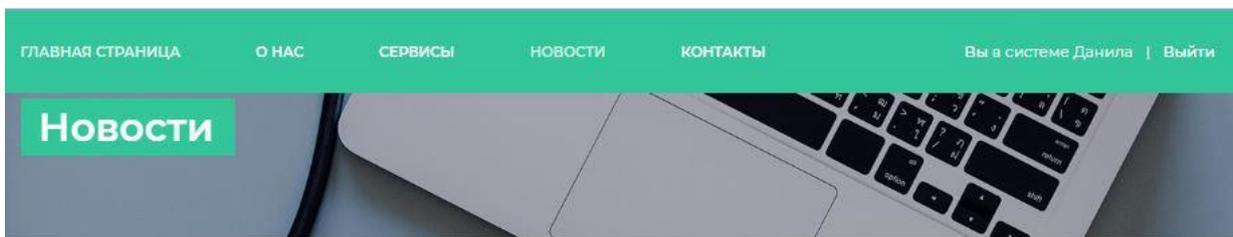


Рисунок 17– Страница новостей сайта MedInfo

На странице с контактами была разработана форма с обратной связью. Благодаря этой форме, посетители сайта Medinfo могут отправлять сообщения врачам по email рисунок 18.

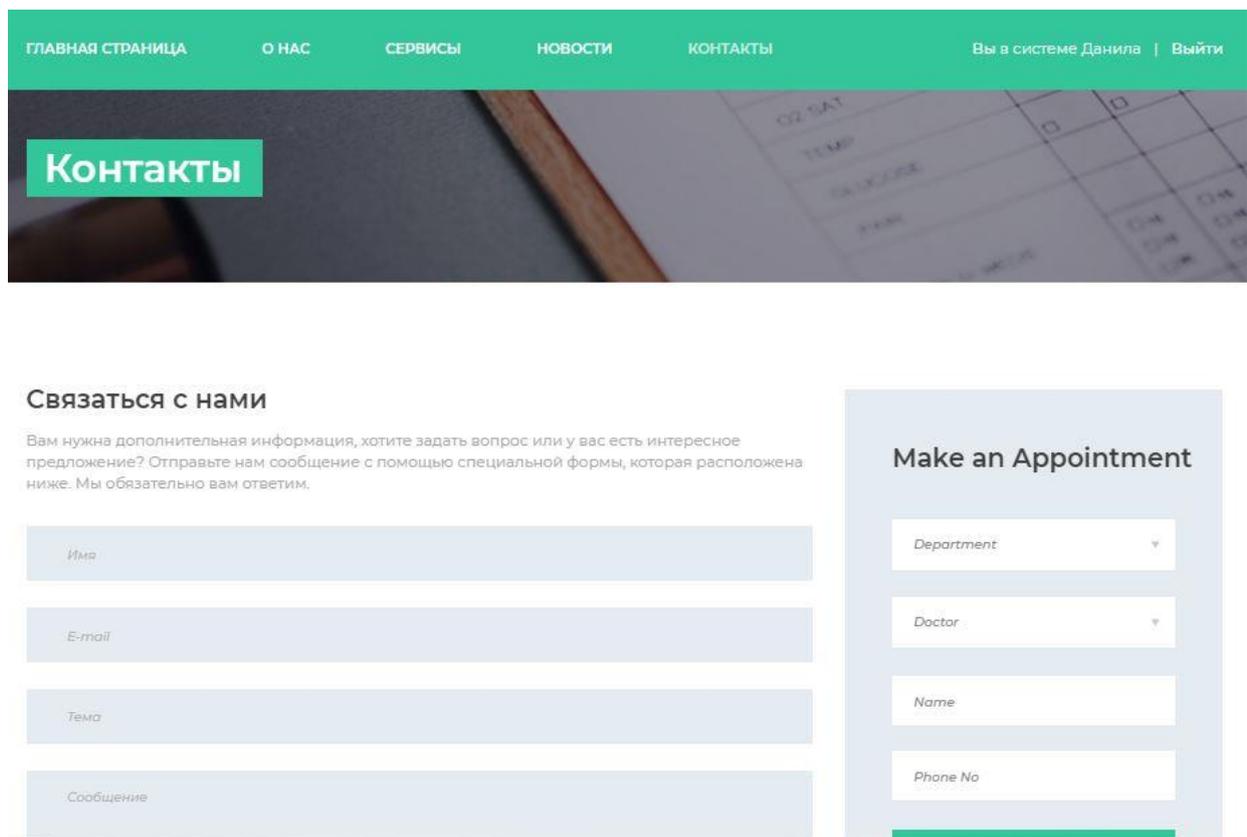


Рисунок 18– Страница обратной связи сайта MedInfo

3.3.2 Постройка графиков с помощью плагина D3.js

Для визуализации данных из справок по анализу крови было решено использовать плагин d3.js. D3.js (или просто D3) являющийся JavaScript-библиотекой для обработки и визуализации данных. Она предоставляет удобные утилиты для обработки и загрузки массивов данных и создания DOM-элементов [17].

Например, возьмем все результаты ОАК пациента под номером 8 за время проведенного в больнице (все реальные имена были обезличены, поэтому для обозначения пациентов использовались id номера) рисунок 19

Hbg	Rbc	Wbc	Soe_Esr	Hct	ReticCt	MyelBl	ProMielBl	MielCt	MetMiel	Band_Neu	Sgmt_Neu	EosPh	BasoPh	MonoCt	LymphCt	PlasmCt	ProLymphCt	LymphBl
97	3,3	9	56	27	0,1	0	0	0	0	20	68	1	0	3	8	0	0	0
123	3,85	7,3	50	37	0,4	0	0	0	0	13	51	1	0	6	28	1	0	0
124	3,95	6,7	46	38	0,9	0	0	0	0	2	53	2	0	10	33	0	0	0
131	4,2	12,6	34	41	1,2	0	0	0	0	5	75	0	0	3	17	0	0	0
110	3,5	10,5	45	33	0,9	0	0	0	0	4	66	0	0	8	22	0	0	0
127	4,7	9,2	20	43	0,8	0	0	0	0	3	61	2	0	9	25	0	0	0
129	4,6	9,2	11	45	0,8	0	0	0	0	6	61	6	0	8	19	0	0	0
128	4,4	4,8	13	40	0,6	0	0	0	0	1	47	1	0	13	38	0	0	0

Рисунок 19– Все результаты ОАК пациента №8

На рисунке 20 представлен график общей оценки изменения индекса интоксикации для пациента под номером 8.

График общей оценки изменения индекса интоксикации (Калиф-Калифа) для пациента №8

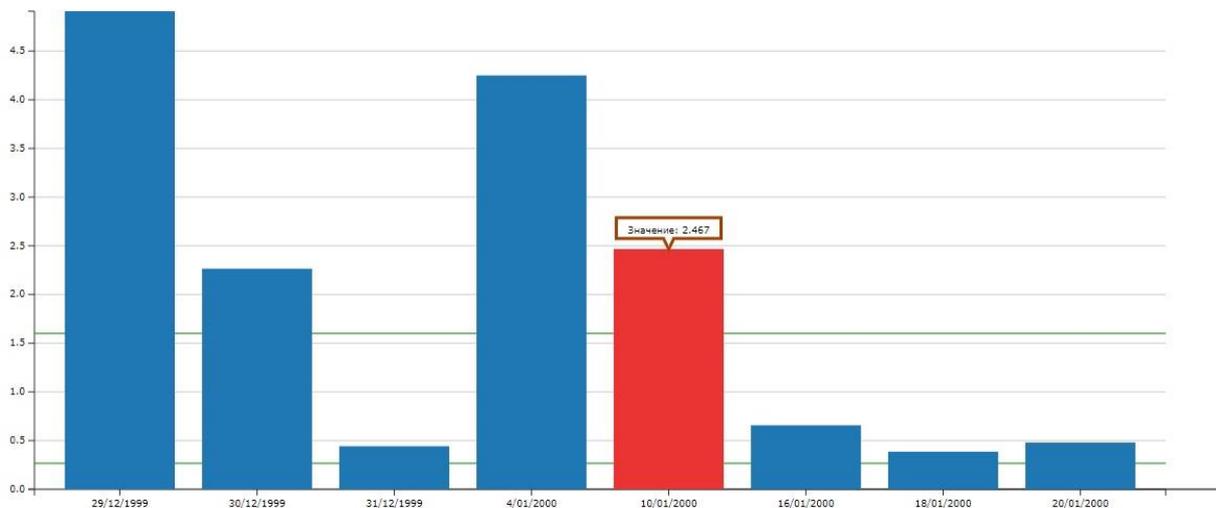


Рисунок 20– График общей оценки изменения индекса интоксикации

На рисунке 21 представлен график общей оценки изменения индекса сдвига лейкоцитов крови для пациента под номером 8.

График общей оценки изменения индекса сдвига лейкоцитов крови для пациента №8

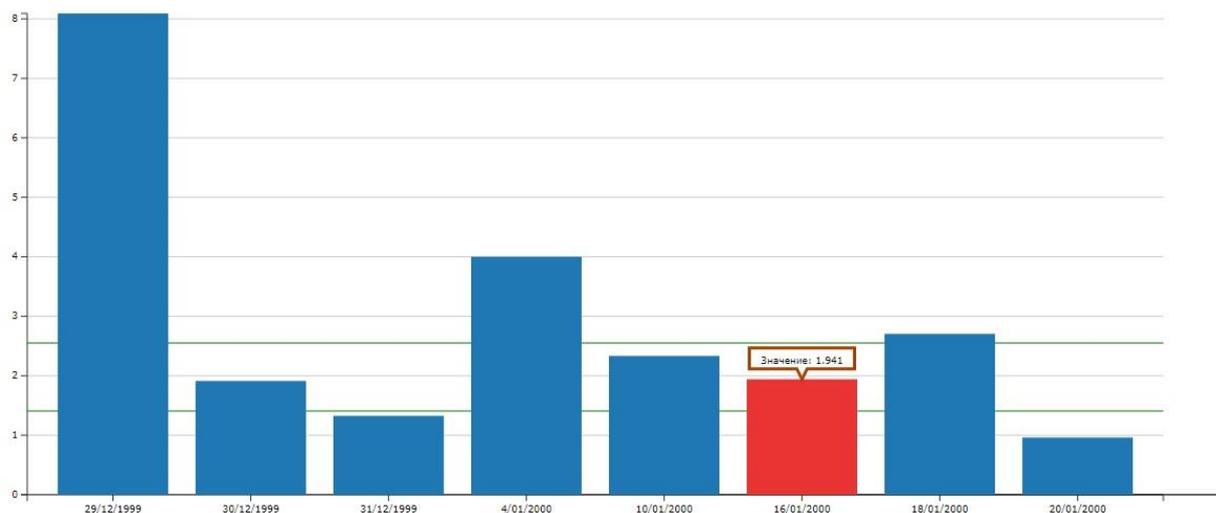


Рисунок 21– График общей оценки изменения индекса сдвига лейкоцитов крови

На рисунке 22 представлен график общей оценки изменения индекса алергизации для пациента под номером 8.

График общей оценки изменения индекса аллергизации (ИА) для пациента №8



Рисунок 22– График общей оценки изменения индекса аллергизации

Таким образом, благодаря этим разным графикам по общим оценкам изменений ЛИ, врачи могут быстро и эффективно проследить динамику лечения пациента

3.3.3 Анализ результатов ОАК по нормам крови

Данная МИС может также анализировать результаты ОАК по нормам крови. Нормы крови – это числовые диапазоны, которые показывают нормальное количество того или иного типа клеток крови для здорового человека. Эти числовые диапазоны могут варьироваться в зависимости от пола и возраста человека рисунок 20. В данной магистерской диссертации были взяты нормы крови для взрослого мужчины и для взрослой женщины.

	1 месяц	1 год	1-6 лет	7-12 лет	13-15 лет	Женщины	Мужчины
Гемоглобин	115-175 Г/л	110-135 Г/л	110-140 Г/л	110-145 Г/л	115-150 Г/л	120-140 Г/л	130-160 Г/л
Эритроциты	$3,8-5,6 \times 10^{12}$	$3,6-4,9 \times 10^{12}$	$3,5-4,5 \times 10^{12}$	$3,5-4,7 \times 10^{12}$	$3,6-5,1 \times 10^{12}$	$3,7 \times 4,7 \times 10^{12}$	$4-5,1 \times 10^{12}$
Цветовой показатель	0,85-1,15	0,85-1,15	0,85-1,15	0,85-1,15	0,85-1,15	0,85-1,15	0,85-1,15
Ретикулоциты	3-15%	3-15%	3-12%	3-12%	2-11%	0,2-1,2%	0,2-1,2%
Тромбоциты	$180-400 \times 10^9$	$180-400 \times 10^9$	$160-390 \times 10^9$	$160-380 \times 10^9$	$160-360 \times 10^9$	$180-320 \times 10^9$	$180-320 \times 10^9$
СОЭ	4-8 мм/ч	4-12 мм/ч	4-12 мм/ч	4-12 мм/ч	4-15 мм/ч	2-15 мм/ч	1-10 мм/ч
Лейкоциты	$6,5-13,8 \times 10^9$	$6-12 \times 10^9$	$5-12 \times 10^9$	$4,5-10 \times 10^9$	$4,3-9,5 \times 10^9$	$4-9 \times 10^9$	$4-9 \times 10^9$
Палочко-ядерные	0,5-4%	0,5-4%	0,5-5%	0,5-5%	0,5-6%	1-6%	1-6%
Сегментоядерные	15-45%	15-45%	25-60%	35-65%	40-65%	47-72%	47-72%
Эозинофилы	0,5-7%	0,5-7%	3,5-7%	0,5-7%	0,5-5%	0-5%	0-5%
Базофилы	0-1%	0-1%	0-1%	0-1%	0-1%	0-1%	0-1%
Лимфоциты	40-76%	38-72%	26-60%	24-54%	25-50%	18-40%	18-40%
Моноциты	2-12%	2-12%	2-10%	2-10%	2-10%	2-9%	2-9%

Рисунок 23– Таблица нормы крови

На рисунке 24 показан результат анализа ОАК пациента №8, который был получен в первый день госпитализации. Как видно на рисунке 24, если значение какого-либо типа крови превышает допустимую норму крови, то данная ячейка с этим значением окрашивается в красный цвет, тем самым облегчает врачу в анализе медицинских справок.

Тип клетки	Значение	Размерность
Гемоглобин	97	г/л
Общее количество эритроцитов	3,3	10 ¹² клеток/л
Общее количество лейкоцитов	9	10 ⁹ клеток/л
Скорость оседания эритроцитов	56	мм/час
Гематокрит	27	%
Ретикулоцит	0,1	%
Миелобласты	0	%
Промиелобласты	0	%
Миелоциты	0	%
Метамиелоциты	0	%
Палочкоядерные нейтрофилы	20	%
Сегментоядерные нейтрофилы	68	%
Эозинофилы	1	%
Базофилы	0	%
Моноциты	3	%
Лимфоциты	8	%
Плазматические клетки	0	%
Пролимфоциты	0	%
Лимфобласты	0	%

Рисунок 24– Пример таблицы результатов ОАК

3.3.4 Разработка голосового интерфейса

На сегодняшний день, самым распространенным устройством ввода информации в компьютер является клавиатура. Но иногда оператору ПК приходится вводить большое количество информации в компьютер. Бывает и такое, что оператор ПК не обладает должным навыком печати на клавиатуре, из-за чего процесс ввода информации в компьютер затягивается. Вследствие чего, происходит потеря такого драгоценного ресурса, как время.

Одним из способов решения данной проблемы является использование речевых технологий. Создание систем автоматического распознавания речи достаточно актуальное направление в сфере развития информационных технологий. Системы распознавания речи — системы, которые позволяют преобразовывать в компьютерную форму, представленную слитную проблемно-ориентированную человеческую речь.

Многим врачам из разных медицинских учреждений приходится заполнять медицинские карточки пациентов. Зачастую в медицинские карточки (истории болезней) вводят много подробной информации о пациентах, из-за чего врачам приходится тратить значительную часть времени на заполнение этих карточек. Поэтому в рамках данной магистерской диссертации предлагается внедрить голосовой интерфейс в МИС с целью ускоренного заполнения врачом истории болезни пациентов.

Согласно [18], к внедряемым в медицинские учреждения информационным системам предъявляются следующие требования: врачи и администрация должны быть освобождены от трудоемкого процесса составления отчетов; необходимо прекратить зависимость от почерка врача и от многократного, повторного ввода данных; рабочее место медицинского сотрудника должно отвечать современным требованиям: удобный и интуитивно понятный пользовательский интерфейс, показ только необходимых для работы форм; качественно изменить уровень медицинского обслуживания в лечебных учреждениях и повысить эффективность их работы.

Использование медицинским персоналом технологии голосового ввода информации в истории болезней пациентов значительно ускорило бы ввод данных. Данное нововведение позволит врачу сэкономить его рабочее время, следовательно, это приведет и к повышению его производительности на рабочем месте.

3.3.4.1 Экспериментальная часть разработки голосового интерфейса

Перед нами была поставлена задача разработки программы с голосовым интерфейсом для ввода и оцифровки информации из медицинских карточек пациентов. Было решено разработать программу в виде web-приложения.

Для реализации программы было решено использовать уже готовый модуль голосового ввода, а именно webkit Speech Recognition (Web Speech API) который используется в Chrome. Данный API заботится о конфиденциальности пользователей и нигде не хранит историю распознавания, а также не хранит нигде введённый текст. Перед использованием распознавания речи, пользователь должен разрешить доступ к микрофону (исключение составляет, если программа запущена через локальный web-сервер) [19]. Для использования данного модуля в web-приложении необходимо подключить его с помощью языка javascript. При этом в коде необходимо указать, что будет использоваться русский язык.

3.3.4.2 Результаты работы в создании голосового интерфейса

Согласно поставленной задаче, было разработано web-страница в МИС для голосового ввода информации в историю болезни пациента. Данный раздел называется «Осмотр врача при поступлении». Разработанная форма содержит все необходимые поля, соответствующие полям документа. В правом верхнем углу каждого поля ввода есть кнопка микрофона, как показано на рисунке 25.

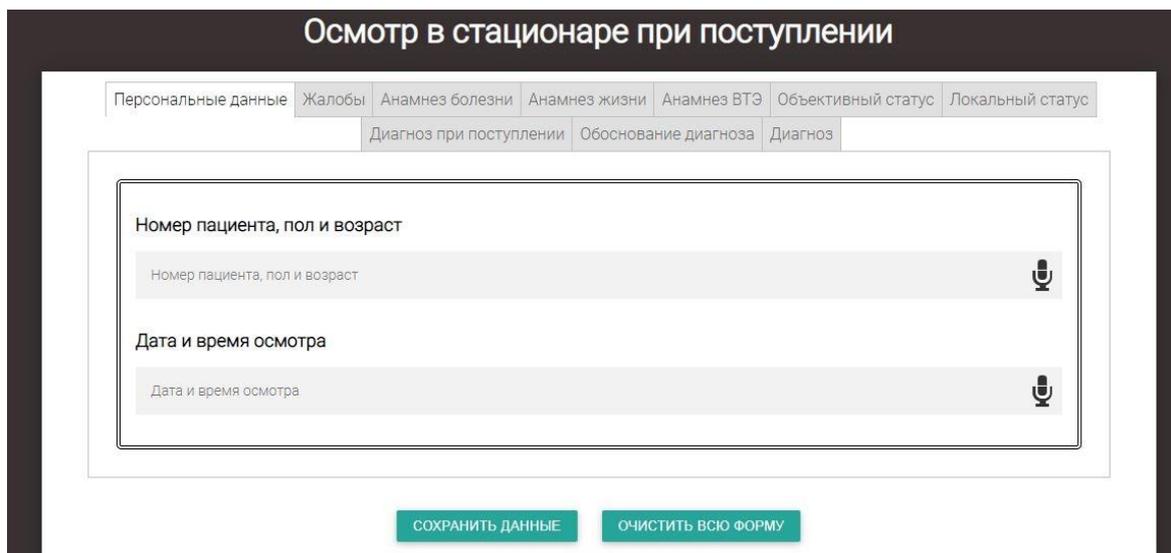


Рисунок 25– Интерфейс web-страницы для ввода историй болезней пациентов в БД

При нажатии на кнопку, начнется запись голоса с помощью периферийного микрофона. После того, как врач перестал говорить, запись остановится через несколько секунд. Также врач может самостоятельно остановить запись при повторном нажатии на кнопку с иконкой микрофона. В процессе голосового ввода границы используемого поля выделяются красным, выделение снимается при остановке голосового ввода любым из описанных выше способов.

По окончании ввода всех данных, врач нажимает на кнопку «Сохранить данные» для последующей записи медицинской карточки в БД.

3.3.4.3 Проблема с SSL-сертификатом

При размещении данной программы на удаленный сервер возникла проблема с предоставлением доступа браузера над микрофоном компьютера. Данная проблема связана с получением SSL – сертификата. Протокол SSL используется в тех случаях, если нужно обеспечить должный уровень защиты информации, которую пользователь передает серверу [20]. Для получения данного сертификата необходимы финансовые вложения.

3.3.4.4 Заключение по голосовому интерфейсу

Таким образом, благодаря голосовому интерфейсу врачи повышают свою производительность труда. При тестировании данного API было выявлено, что он довольно неплохо преобразовывает и записывает тексты в объекты `input` и `textarea` HTML 5. При записи голоса использовался встроенный микрофон. Также в результате данного исследования была написана и опубликована статья «Разработка программы голосового ввода в виде web-приложения для эффективного заполнения медицинских карточек пациентов» [25].

3.3.5 Модуль Django-allauth авторизации и регистрации

Django-allauth это готовый пакет (данный пакет можно скачать с репозитория python) для авторизации пользователей на своём сайте, который берёт на себя рутинные операции по регистрации пользователей обычным способом через форму, а также через популярные социальные сети.

Используя готовую библиотеку **Django-allauth** для авторизации и регистрации, вы без проблем сможете ее настроить под свои нужды. Потому что готовые решения, как правило, берут на себя всю рутину по регистрации, входу на сайт и добавление нескольких учётных записей разных социальных приложений в один аккаунт. Также кроме типичных полей Имя, Фамилия, Email и Пароль вы легко сможете добавить дополнительные поля. Например, Отчество, дату рождения, пол и т.д.

Также данный пакет обладает возможностью добавлять несколько email-ов пользователей и указывать, какой из них должен быть основным. Иногда бывает, что пользователь неожиданно теряет доступ к своему электронному ящику. В данном случае на помощь приходят резервные email-ы, на которые могут прийти пароли или важные письма с вашего сайта [21]. При настройке данного пакета я сделал традиционную регистрацию пользователя в МИС, т.е. пользователь после регистрации, должен подтвердить свою регистрацию путем перехода по ссылке, которую он получит по электронной почте.

На рисунке 26 представлена web-страница регистрации пользователя в МИС.

The image shows a registration form with the following fields and elements:

- Header:** "Зарегистрироваться" (Register)
- Text:** "Уже зарегистрированы? Войдите." (Already registered? Log in.)
- E-mail:** Input field with placeholder "E-mail адрес".
- Имя пользователя (Username):** Input field containing the text "root".
- Фамилия (Surname):** Empty input field.
- Имя (Name):** Empty input field.
- Пароль (Password):** Input field with masked characters ".....".
- Пароль (еще раз) (Password (again)):** Input field with placeholder "Пароль (еще раз)".
- Button:** A blue button labeled "Регистрация" (Registration).

Рисунок 26– страница регистрации пользователя

На рисунке 27 представлена web-страница авторизации пользователя в МИС.

Войти

У Вас нет учетной записи? [Зарегистрироваться](#)

Запомнить меня

[Забыли пароль?](#)

Войти

Рисунок 27– страница авторизации пользователя

3.4 Краткий заключительный обзор всей МИС

В данном подразделе представлен краткий обзор всех функций МИС. Помимо сайта-визитки, которая подробно рассказывает об информационной системе и о больнице к которому она подключена, есть также и приватная часть сайта, доступная только зарегистрированным пользователям. После регистрации и авторизации врача-пользователя, информационная система открывает web-страницу личного кабинета врача рисунок 28.

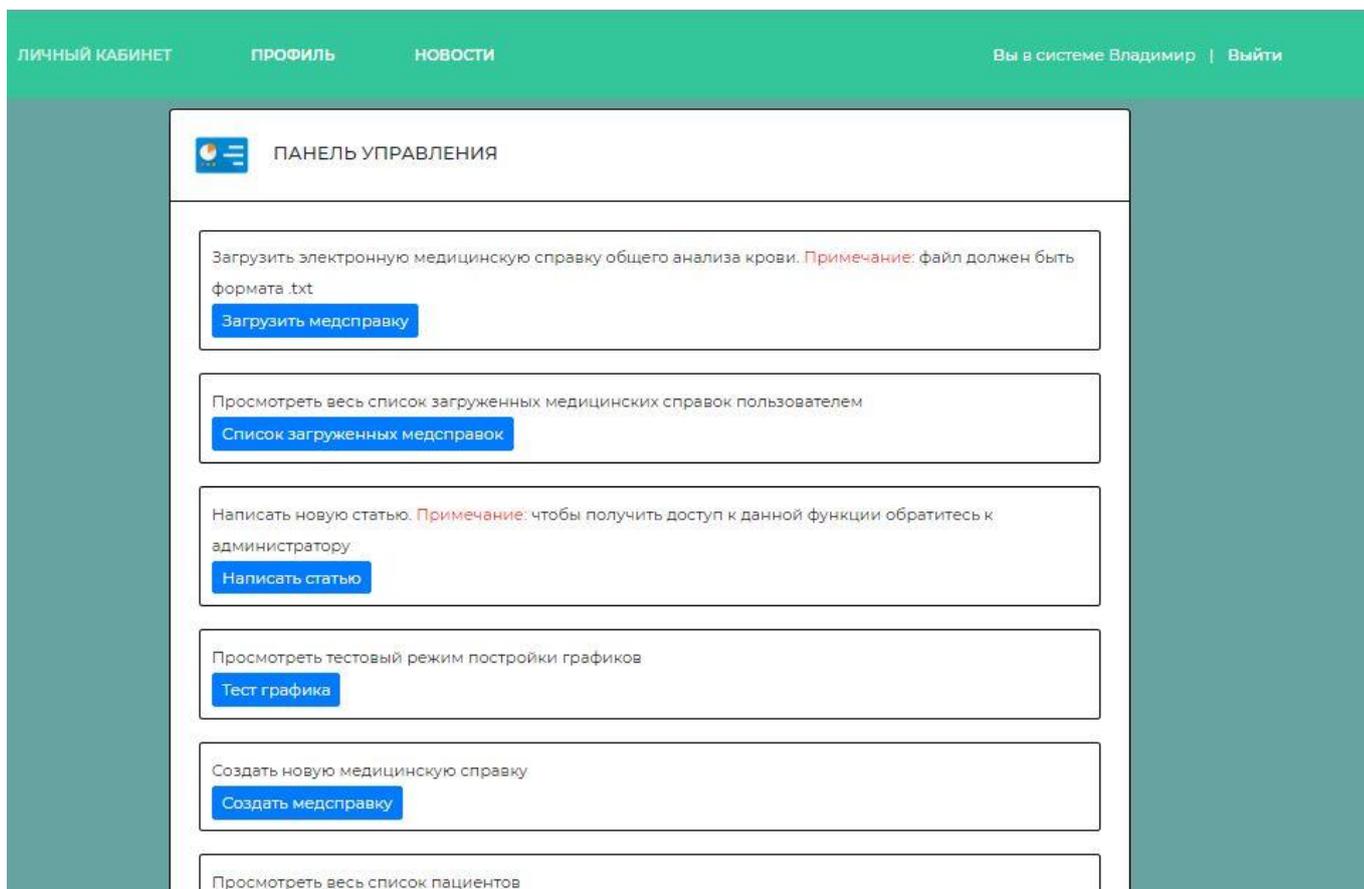


Рисунок 28– страница личного кабинета

На рисунке 28 представлены все доступные на данный момент функции, которые могут использовать врачи.

3.4.1 Функция «Загрузить медсправку»

Данная функция позволяет загрузить еще не обработанную медсправку в БД, которая в дальнейшем будет обработана специальным парсером и загружена по нужным полям в таблицу «**resultmedcertif**» рисунок 29.

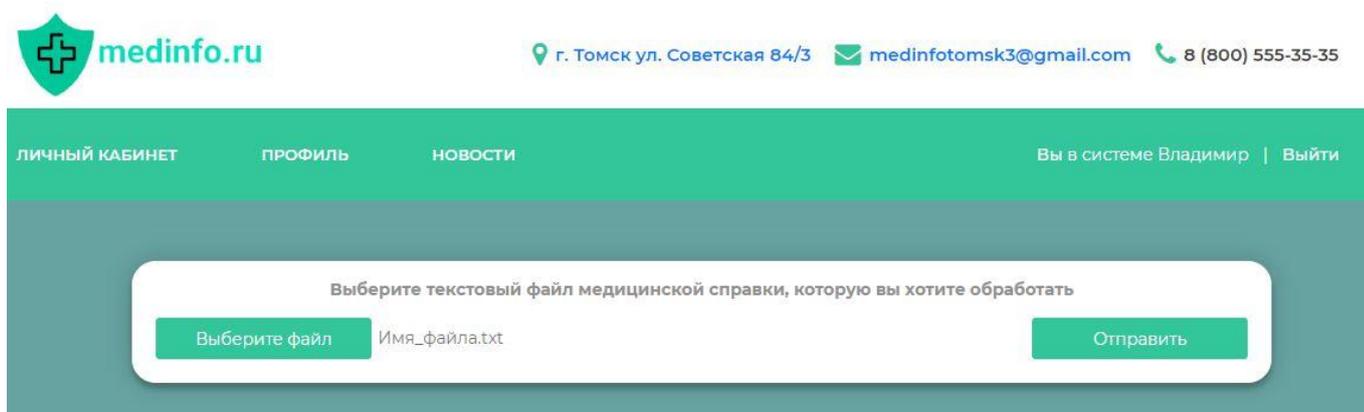
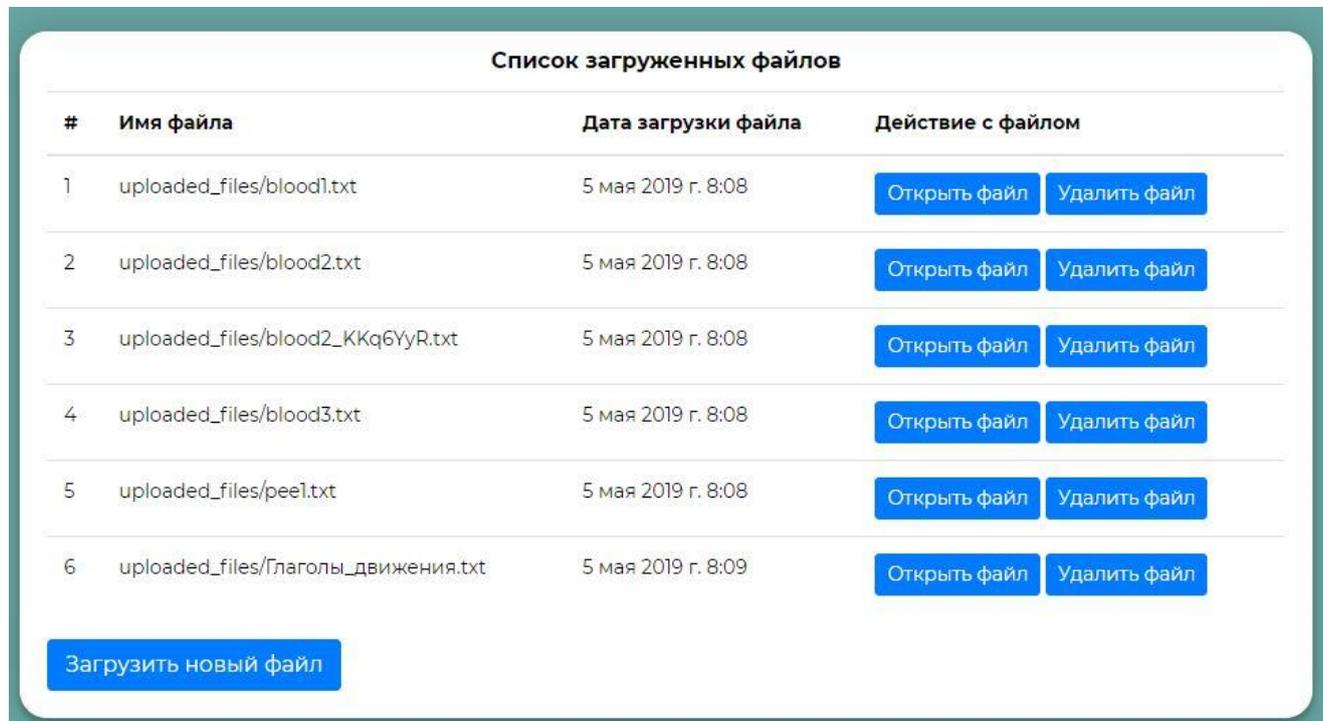


Рисунок 29– Web-страница «Загрузить медсправку»

3.4.2 Функция «Список загруженных медсправок»

Данная функция позволяет просмотреть полный список всех загруженных файлов, которые загружал пользователь за все время работы в МИС рисунок 30.



#	Имя файла	Дата загрузки файла	Действие с файлом
1	uploaded_files/blood1.txt	5 мая 2019 г. 8:08	Открыть файл Удалить файл
2	uploaded_files/blood2.txt	5 мая 2019 г. 8:08	Открыть файл Удалить файл
3	uploaded_files/blood2_KKq6YyR.txt	5 мая 2019 г. 8:08	Открыть файл Удалить файл
4	uploaded_files/blood3.txt	5 мая 2019 г. 8:08	Открыть файл Удалить файл
5	uploaded_files/pee1.txt	5 мая 2019 г. 8:08	Открыть файл Удалить файл
6	uploaded_files/Глаголы_движения.txt	5 мая 2019 г. 8:09	Открыть файл Удалить файл

[Загрузить новый файл](#)

Рисунок 30– Web-страница «Список загруженных медсправок»

Пользователь в специальной таблице может увидеть имя данного файла и дату загрузки файла. Также пользователь может просмотреть содержание данного файла, а также удалить его.

3.4.3 Функция «Написать статью»

Данная функция позволяет врачам создавать свои статьи, которые в дальнейшем сохраняются в БД. После нажатие на кнопку «Написать статью» МИС переносит вас в административную панель Django рисунок 31.

Администрирование Django доб

Начало > Mainapp > Статьи > Добавить Статья

✔ Статья "id: 5 | Тест статьи" был изменен. Вы можете добавить еще один Статья ниже.

Добавить Статья

Name:

Post:

лейкоцитов. Технологии подсчета лейкоцитарной формулы различаются у разных производителей.
Есть много разных Лейкоцитарных индексов с их клиническими значениями. Благодаря этим индексам можно определить, в каком состоянии сейчас находится пациент. Также, если у пациента несколько раз делали анализ крови во время лечения болезни, то информационная система может составить оценку эффективности лечения данного пациента на основе лейкоцитарных индексов. Также для наглядности можно составить график оценки эффективности лечения. Благодаря этой оценке, врачи могут создать быструю и эффективную стратегию по лечению той или иной болезни. Данную стратегию лечения можно использовать на пациентах с идентичной болезнью.

Image: 0-e152524...77447.jpg

Рисунок 31– Административную панель Django для добавления статьи

Пользователь может вставить название статьи и сам текст статьи. Также врач может по желанию прикрепить картинку к данной статье. Дата прикрепляется к статье автоматически после публикации. После сохранения статья автоматически начинает отображаться в блоге новостей рисунок 32.



Опубликовано: 22-05-2019 в 17:39

Тест статьи

автор: Богданов Данила /

Главная особенность информационной системы (помимо обработки и хранения в БД медсправок) заключается в объективной оценке состояния тяжести пациента по показателю крови. Для оценки состояния используется вычисление «Лейкоцитарной формулы» (Лейкоцитарные индексы). Лейкоцитарная формула – процентное соотношение различных видов лейкоцитов в перифер...

[ЧИТАТЬ ДАЛЕЕ](#)

Рисунок 32– Опубликованная статья в блоге новостей

3.4.4 Функция «Просмотреть графики»

Как уже говорилось раньше данная функция позволяет строить графики по общим оценкам ЛИ, какого-либо конкретного пациента рисунок 20, рисунок 21, рисунок 22.

3.4.5 Функция «Создать медсправку»

Данная функция позволяет врачам создавать новые медсправки результатов ОАК пациентов и загружать их в БД рисунок 33.

Медицинская справка ОАК

Персональные данные пациента

Дата создания ОАК: 2019-05-21
 Время создания ОАК: 19:29
 Дата создания электронной медсправки ОАК: 21-5-2019
 Время создания электронной медсправки ОАК: 19:29
 Пол пациента: Мужской
 Фамилия: Романов
 Имя: Александр
 Отчество: Владимирович
 Возраст: 19

Показатель	Значение	Ед.Изм.
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾
<input type="text"/>	<input type="text"/>	Выберите размерность ▾

Рисунок 33– Процесс создания новой медсправки

3.4.6 Функция «Список пациентов»

Данная функция позволяет врачам просмотреть всю базу пациентов, которые проходили обследование или госпитализацию в данной больнице рисунок 34.

г. Томск ул. Советская 84/3

medinfotomsk3@gmail.com

8 (800) 555-35-35

ЛИЧНЫЙ КАБИНЕТ
ПРОФИЛЬ
НОВОСТИ
Вы в системе Владимир | Выйти

Список Пациентов

#	Фамилия	Имя	Отчество	Дата рождения	email	Пол	Мобильный тел.	Домашний тел.	Место жительства
	Иванов	Иван	Иваныч	1 декабря 1979 г.	example@mail.com	Мужской	1234	1234	ул. Пушкина д. Колотушкина
	Цветаева	Мария	Ивановна	14 апреля 1981 г.	example@mail.com	Женский	1234	1234	ул. Пушкина д. Колотушкина
	Бард	Лютик	Лютикович	1 июля 1981 г.	example@mail.com	Мужской	1234	1234	None
	Романов	Александр	Владимирович	11 апреля 1981 г.	example@mail.com	Мужской	1234	1234	ул. Пушкина д. Колотушкина

Рисунок 34– Список всех пациентов БД

3.4.7 Функция «Лейкоцитарные индексы»

Данная функция предоставляет врачам калькуляторы всех доступных ЛИ в МИС рисунок 35.

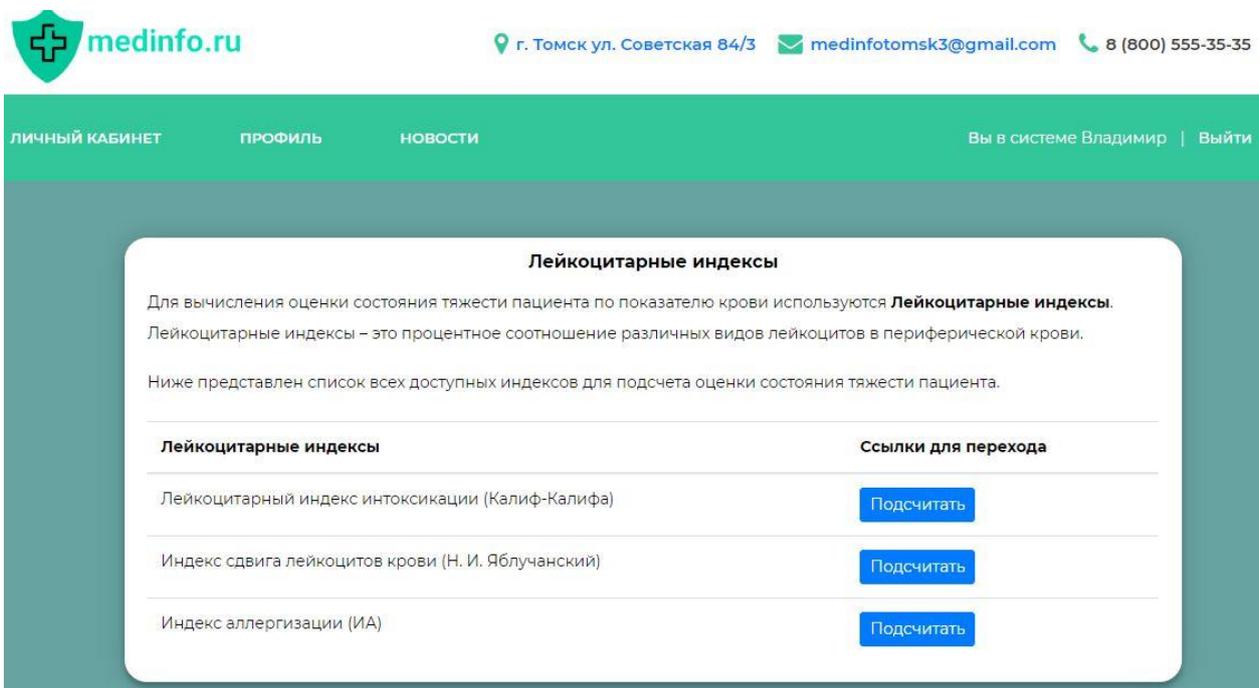


Рисунок 35– Список всех ЛИ

Например, перейдя по ссылке ЛИ интоксикации, загрузится web-страница, на котором будет отображена все информация про ЛИ и его нормы. Ниже будет сам калькулятор по подсчету индекса интоксикации. Введя необходимые данные в поля следует нажать на кнопку «Рассчитать индекс». После нажатия на кнопку, калькулятор автоматически рассчитает и выдаст результат рисунок 36.

ЛИЧНЫЙ КАБИНЕТ ПРОФИЛЬ НОВОСТИ Вы в системе Владимир | Выйти

Нормы интоксикации:

- 0-5 – нормальная величина
- 5-8 – начальная интоксикация
- 10-15 – умеренная интоксикация
- 18-25 – выраженная интоксикация
- 25 – терминальная интоксикация

Миело –	<input type="text" value="0"/>	%
Ю –	<input type="text" value="0"/>	%
ПЯ –	<input type="text" value="20"/>	%
СЯ –	<input type="text" value="68"/>	%
Плазм –	<input type="text" value="0"/>	%
Моно –	<input type="text" value="3"/>	%
Лимфо –	<input type="text" value="8"/>	%
Эо –	<input type="text" value="1"/>	%

Индекс интоксикации: 6.909

Рисунок 36– Результат работы калькулятора расчета ЛИ интоксикации

3.4.8 Личный профиль врача

Также врач-пользователь может ввести свою персональную информацию в свой аккаунт рисунок 37. Это сделано для того, чтобы врачи лучше знали друг друга, а также смогли в случае чего связаться друг с другом.

ЛИЧНЫЙ КАБИНЕТ ПРОФИЛЬ НОВОСТИ Вы в системе Владимир | Выйти

Скворцов Владимир Генадьевич



Имя пользователя:	user
Email:	Danilka00000@rambler.ru
Дата рождения:	2 мая 2019 г.
Домашний адрес:	ул. Пушкина д. Колотушкина
Рабочий телефон:	8-88-88
Домашний телефон:	9-99-99
Мобильный телефон:	+79999999999
Образование:	Сибирский государственный медицинский университет
Специализация:	терапевт
Опыт работы с:	2 мая 2019 г.
Пол:	М

Рисунок 37– Личный профиль пользователя

4 ТЕСТИРОВАНИЕ РАЗРАБОТАННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НА КОНТРОЛЬНЫХ ПРИМЕРАХ

После разработки МИС необходимо протестировать разработанную программу по обработке медицинских справок ОАК на реальных контрольных примерах. В качестве примера были взяты пять разных электронных справок ОАК. Три из них приведены на рисунке 38.

1	Общий анализ крови	8	Наименование исследования	2	ПОКАЗАТЕЛИ
2		9	Результат	3	РЕЗУЛЬТАТ
3		10	Ед. изм.	4	
4		11	Нормальные	5	НОРМЫ
5	Название теста	12	Наименование исследования	6	ЕДИНИЦЫ
6	Результат	13		7	Гематокрит (Ht)
7		14	Результат	8	
8	Ед. Изм.	15		9	50.5
9	Референтные	16	Ед. изм.	10	
10	Лейкоциты /WBC)	17		11	38-55
11	6,9	18	Оттачения	12	%
12		19	ОБЩИЙ АНАЛИЗ КРОВИ	13	Гемоглобин (Hb)
13		20		14	
14	10л9фл	21		15	187
15	4,0 9,0	22		16	
16	Эритроциты (RBC)	23	Гемоглобин	17	130-180
17	5,77	24	131	18	г/л
18	выше	25	г/л	19	Эритроциты (RBC)
19	10*12/л	26	117 - 155	20	
20	4,00-5,00	27	Эритроциты	21	7.42
21	Гемоглобин (HGB)	28	4.72	22	
22	170	29	x10*12/л	23	5,6-8
23	выше	30	3.8 - 5.1	24	X10(12]/л
24	с/л	31	Гематокрит	25	Среднее содержание Hb в эритроците (MCH)
25	130-160	32	38.3	26	
26	Гематокоит /HCT)	33	%	27	25.2

Рисунок 38– Примеры тестовых справок файлов Test_1.txt, Test_2.txt и Test_3.txt

За точность обработанных справок будем считать количество правильных строк, которые в дальнейшем будут переведены в процентное соотношение, т.е. в процент точности обработки справок. Далее после получения процентной точности обработки каждой справки можно будет получить среднее значение процентной точности по обработке медицинских справок ОАК web-приложением.

Также необходимо отметить, что перед извлечением ключевых данных, происходит предобработка справок, а именно весь текст в файле автоматически конвертируется в строчные буквы. Т.к. все ключевые данные в БД, по которым ищутся в совпадения в медсправках, записаны в строчном виде.

4.1.1 Результаты тестирования web-приложения

В результате тестирования данного web-приложения, были получены следующие данные:

- При обработке медсправки Test_1.txt 13 строк из 14 были обработаны правильно, следовательно, процент точности данной справки составляет 92,85%.
- При обработке медсправки Test_2.txt 10 строк из 22 были обработаны правильно, следовательно, процент точности данной справки составляет 45,45%.
- При обработке медсправки Test_3.txt 7 строк из 11 были обработаны правильно, следовательно, процент точности данной справки составляет 63,63%.
- При обработке медсправки Test_4.txt 8 строк из 15 были обработаны правильно, следовательно, процент точности данной справки составляет 53,3%.
- При обработке медсправки Test_5.txt 4 строк из 14 были обработаны правильно, следовательно, процент точности данной справки составляет 28,6%.

Теперь после нахождения процентной точности каждой справки, найдем среднюю точность обработки медсправок данным приложением. В результате подсчёта мы получаем, что данное web-приложение обрабатывает электронные медицинские справки с точностью **56,76%**.

К сожалению, данный результат показывает, что программа может только на половину достать ключевые данные из медицинской справки, что конечно же не является хорошим результатом. Основной проблемой является из-за того, что парсер программы просто пропускает ключевые значения в медсправке, т.к. попросту не замечает её. Это связано с тем, что словарный запас ключевых данных программы не полностью заполнен ключевыми данными. Например, размерность $10^{12}/л$ будет обнаружена и выделена программой, в то время как размерность $10 * 12/л$ не будет обнаружена, т.к. её

попусту нет в БД программы. Что-бы этого избежать, надо просто размерность 10*12/л добавить в БД программы. Также иногда встречаются нетривиальные проблемы по обработке справок. Например, какие-то незнакомые символы, после которых обработка справок прерывается или обработка продолжается с ошибками. В данном случае необходимо вручную проанализировать и устранить ошибки в данной справке. И только после этого медсправку можно обработать, чтобы извлечь ключевые данные.

Во избежание всех этих проблем, необходимо, как можно больше обработать справки данной программой, чтобы найти недостатки в работе web-приложения и устранить их, тем самым повысить точность обработки электронных медицинских справок.

5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Данный раздел является обязательной частью выпускной квалификационной работы и представляет собой детальный анализ экономической эффективности проекта, а также оценку коммерческой ценности разработки. В этой части решались следующие задачи:

- оценка коммерческого потенциала и перспективности проведения научных исследований;
- определение возможных альтернатив проведения научных исследований, отвечающих современным требованиям в области ресурсоэффективности и ресурсосбережения;
- планирование научно-исследовательских работ;
- определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.

Целью представленной магистерской диссертации является разработка медицинской информационной системы для обработки результатов общего анализа крови.

Для достижения данной цели необходимо было выполнить ряд задач, в том числе дать оценку коммерческих возможностей проведенного исследования, выявить его ресурсосберегающий потенциал, определить финансовую эффективность исследования, а также выявить сильные и слабые стороны данной разработки.

Предметом моего исследования являются медицинские справки в виде текстовых файлов, полученные из “анализаторов крови” или же отсканированные медицинские справки сканером, которые были сохранены в текстовом формате. В рамках моей работы использовались справки формата А4 и А5.

Данное исследование призвано решить проблему приведения электронных медицинских справок к единому стандарту, а также в экономии рабочего времени врача.

5.1 Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения

5.1.1 Потенциальные потребители результатов исследования

Разработанная медицинская информационная система (МИС) позволит рассчитывать оценки состояния здоровья пациентов на основе результатов общего анализа крови. В работе реализованы процедуры выполнения расчетов лейкоцитарных индексов по результатам анализа крови пациентов. Данный программный продукт можно отнести к медицинской сфере, в частности, эта информационная система представляет большой интерес для врачей. На данный момент эта МИС специально разрабатывается для Сибирского государственного медицинского университета, кафедры инфекционных болезней и эпидемиологии. В дальнейшем после успешного тестирования МИС, планируется постепенно внедрять его в частные медицинские центры. Это делается для того, чтобы получить положительные отзывы и рекомендации о МИС. Такие отзывы помогут получить сертификат от Минздрава. Благодаря этому сертификату данную ИС можно уже будет внедрять в государственные больницы и поликлиники.

Для анализа потребителей результатов исследования необходимо рассмотреть целевой рынок и провести его сегментирование.

Целевой рынок – сегменты рынка, на котором будет продаваться в будущем разработка. В свою очередь, сегмент рынка – это особым образом выделенная часть рынка, группы потребителей, обладающих определенными общими признаками.

Сегментирование – это разделение покупателей на однородные группы, для каждой из которых может потребоваться определенный товар (услуга).

В зависимости от категории потребителей (коммерческие организации, физические лица) необходимо использовать соответствующие критерии сегментирования. В данном случае в рамках критериях будет использоваться медицинские организации и учреждения, где именно будет использоваться МИС.

Таблица 12 – Сегментирование рынка

		Область применения
		Медицина
Медицинские организации и учреждения.	В государственные поликлиники	Долгосрочная перспектива
	В государственные больницы	Долгосрочная перспектива
	В стационары	Долгосрочная перспектива
	В частные медицинские центры	Краткосрочная перспектива
	В СибГМУ	На этапе внедрения

5.1.2 Анализ конкурентных технических решений

Для достижения оптимальной ресурсоэффективности и востребованности, предлагаемой нами системы, необходимо провести анализ существующих решений по данной задаче. Для этого используется оценочная карта.

В таблице 13 представлен анализ сравнения характеристик разрабатываемой нами МИС и МИС конкурента. Под конкурентом подразумевается Медицинская информационная система под названием «Медицинский помощник», который уже внедрен во многих медицинских учреждениях по всей России. Данная система имеет шифрование персональных данных пациентов, а также бесплатное обучение для будущих пользователей

системы. Позиция представляемой разработки и системы конкурентов была оценена нами по каждому показателю экспертным путем по пятибалльной шкале, где 1 – наиболее слабая позиция, а 5 – наиболее сильная. Веса показателей, определяемые экспертным путем, в сумме составляют 1.

Таблица 13 – Оценочная карта для сравнения конкурентных технических решений

Критерии оценки	Вес критерии	Баллы		Конкурентоспособность	
		Б _ф	Б _{к1}	К _ф	К _{к1}
1	2	3	4	5	6
1. Повышение производительности труда пользователя	0,2	5	3	1	0,6
2. Надежность	0,05	5	4	0,25	0,2
3. Функциональная мощность (предоставляемые возможности)	0,1	5	4	0,5	0,4
4. Потребность в ресурсах памяти	0,04	5	5	0,2	0,2
5. Простота эксплуатации	0,1	5	4	0,5	0,4
6. Конкурентоспособность продукта	0,2	5	4	1	0,8
7. Уровень проникновения на рынок	0,1	5	4	0,5	0,4
8. Перспективность рынка	0,05	5	5	0,25	0,25
9. Цена	0,07	5	3	0,35	0,21
10. Послепродажное обслуживание	0,05	5	5	0,25	0,25
11. Финансовая эффективность научной разработки	0,04	5	4	0,2	0,16
Итого	1			5	3,87

Анализ конкурентных технических решений определяется по формуле

(4.1):

$$K = \sum B_i V_i \quad (5.1)$$

где K – конкурентоспособность научной разработки или конкурента;

V_i – вес показателя (в долях единицы);

B_i – балл i -го показателя.

Анализ конкурентоспособности показал, что технические и экономические показатели оценки обсуждаемой разработки опережают показатели конкурентов. А именно по таким позициям, как по простоте эксплуатации, надёжности и т.д. Основными минусами разработок конкурентов является производительность труда и цена.

5.1.3 SWOT - анализ

SWOT-анализ применяют для исследования внешней и внутренней среды проекта, выявления его слабых и сильных сторон, характера угроз, исходящих из внешней и внутренней среды, поиск дополнительных возможностей для улучшения проекта. Такая работа позволяет видеть перспективу повышения конкурентоспособности нашего продукта.

На первом этапе данного анализа были выявлены сильные стороны и слабые стороны проекта, его возможности и угрозы. Результаты первого этапа представлены в таблице 14.

5.2 Планирование научно-исследовательских работ

5.2.1 Структура работ в рамках научного исследования

Для организации и систематизации работы над проектом был составлен план основных видов деятельности. Наличие такого плана призвано обеспечить своевременное и эффективное выполнения задания ВКР.

В таблице 14 представлен перечень этапов, работ и распределение их исполнителей. В качестве исполнителей были определены следующие участники процесса:

–студент (магистрант);

–руководитель.

Таблица 14 – Перечень работ по проекту

Основные этапы	№ раб.	Содержание работ	Должность исполнителя
Подготовительный этап	1	Выбор темы ВКР	Студент Руководитель
	2	Получение технического задания	Студент Руководитель
	3	Подбор материала, его анализ и обобщение	Студент
	4	Формирование возможных решений поставленной задачи, выбор оптимального решения	Студент
Основной этап	5	Программная реализация компьютерной модели	Студент
	6	Тестирование модели	Студент
	7	Исправление ошибок, доработка модели	Студент
	8	Проведение исследований, получение результатов и их анализ	Студент
Заключительный этап	9	Составление отчета о проделанной работе	Студент
	10	Оценка эффективности полученных результатов	Студент Руководитель
	11	Защита ВКР	Студент

5.2.2 Определение трудоемкости выполнения работ

В таблице 15 представлены временные показатели проведения научного исследования

Таблица 15 – Определение временных затрат на проект

№	Название работы	Трудоёмкость работ			Исполнители	Длительность работ в рабочих днях T_{pi}	Длительность работ в календарных днях T_{ki}
		t_{mini} чел - дни	t_{maxi} чел - дни	$t_{ожи}$ чел - дни			
1	Выбор темы ВКР	2	5	3,2	Студент Руководитель	1,6	2
2	Получение технического задания	1	2	1,4	Студент Руководитель	0,7	1
3	Подбор материала, его анализ и обобщение	15	20	17	Студент	8,5	10
4	Формирование возможных решений поставленной задачи, выбор оптимального решения	30	40	34	Студент	34	42
5	Программная реализация компьютерной модели	5	7	5,8	Студент	2,9	4
6	Тестирование модели	2	5	3,2	Студент	3,2	4
7	Исправление ошибок, доработка модели	5	7	5,8	Студент	2,9	4
8	Проведение исследований, получение результатов и их анализ	1	2	1,4	Студент	1,4	2
9	Планирование и проведение экспериментов	10	15	12	Студент	12	15
10	Составление отчета о проделанной работе	3	5	3,8	Студент	1,9	2
11	Защита ВКР	5	8	6,2	Студент	6,2	8
Итого		Всего				74,8	91,3
		Руководитель				19,4	23,67
		Магистр				62,8	89,3

5.2.3 Разработка графика проведения научного исследования

На основе максимального варианта выполнения работ была построена диаграмма Ганта, которая наглядно отображает оптимальные сроки начала и окончания выполнения работ. Результат представлен в таблице 16:

Таблица 16 – Диаграмма Ганта

№	Название работы	Исполнители	T_{ki} , кал. дн.	Продолжительность выполнения работ																				
				декабрь			январь			февраль			март			апрель			май			июнь		
				1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
1	Выбор темы ВКР	Студент Руководитель	12	█	█																			
2	Получение технического задания	Студент Руководитель	5		█																			
3	Подбор материала, его анализ и обобщение	Студент	10			█																		
4	Формирование возможных решений поставленной задачи, выбор оптимального решения	Студент	21				█	█	█															
5	Программная реализация компьютерной модели	Студент	46							█	█	█												
6	Тестирование модели	Студент	14										█	█	█									
7	Исправление ошибок, доработка модели	Студент	20													█	█	█						
8	Проведение исследований, получение результатов и их анализ	Студент	28																█	█	█			
9	Составление отчета о проделанной работе	Студент	18																			█	█	█
10	Оценка эффективности полученных результатов	Студент Руководитель	13																			█	█	█
11	Защита ВКР	Студент	1																					█
	█ -студент	█ -руководитель																						

5.2.4 Бюджет научно-технического исследования (НТИ)

Поскольку для выполнения работы не закупались никакие дополнительные материалы или оборудование, а расходы на канцелярию включаются в накладные расходы, то в процессе формирования бюджета использовались следующие статьи:

- материальные затраты;
- основная заработная плата исполнителей темы;
- дополнительная заработная плата исполнителей темы;
- отчисления во внебюджетные фонды (страховые отчисления);
- накладные расходы.

5.2.5 Расчет материальных затрат НТИ

Для выполнения работы был приобретен персональный компьютер. В данном случае величина материальных расходов не зависит от исполнения.

Материальные затраты представлены в таблице 17:

Таблица 17 – Расчет материальных затрат

Наименование	Единица измерения	Количество	Цена за ед., руб.	Затраты на материалы, (Зм), руб.
Электроэнергия	кВт	112,7	3,42	385,43
Ноутбук	шт.	1	30000	30000
Итого				30385,43

Общая стоимость материальных затрат данного проекта составила 30385,43рублей.

5.2.6 Основная заработная плата исполнителей темы

Основная заработная плата руководителя и исполнителя от ТПУ рассчитывается на основании отраслевой оплаты труда. Отраслевая система оплаты труда в ТПУ предполагает следующий состав заработной платы:

- оклад – определяется предприятием. В ТПУ оклады распределяются в соответствии с занимаемыми должностями. Таким

образом, оклад исполнителей работы распределяется следующим образом:

- руководитель – ППС, доцент, к.т.н. – 33664 руб.;
- студент – УВП – 9489 руб.;
- районный коэффициент – 1,3.

Расчеты основной заработной платы представлены в таблице 18.

Таблица 18 – Расчет основной заработной платы

Исполнители	Оклад с учетом р.к. (З _{тс} , руб.)	Районный коэффициент (к _р)	Месячный оклад (З _м , руб.)	Среднедневная заработная плата (З _{дн} , руб.)	Продолжительность работ (Т _р , раб. дн.)	Основная заработная плата (З _{осн} , руб.)
Руководитель	43763,2	1,3	33664	1989,24	19,4	38591,26
Студент	1906	1,3	2477,8	125,0	62,8	7850
Итого						46441,26

5.2.7 Дополнительная заработная плата исполнителей темы

Затраты по дополнительной заработной плате исполнителей темы учитывают величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда, а также выплат, связанных с обеспечением гарантий и компенсаций.

Расчеты дополнительной заработной платы представлены в таблице 19.

Таблица 19 – Расчет дополнительной заработной платы

Исполнитель	Основная заработная плата (З _{осн} , руб)	коэффициент дополнительной заработной платы (к _{доп})	Дополнительная заработная плата (З _{доп} , руб)
Руководитель	38591,26	0,15	5788,689
Итого			5788,689

Общая сумма дополнительной заработной платы участников проекта составила 5788,689 рублей.

5.2.8 Отчисления во внебюджетные фонды (страховые отчисления)

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников. Расчеты отчислений представлены в таблице 20.

Таблица 20 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.	Полная заработная плата, руб.
Руководитель	38591,26	38591,26
Коэффициент отчислений во внебюджетные фонды	0,271	
Итого	10458,23	

5.2.9 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, почтовые и телеграфные расходы, размножение материалов и т.д.

Величину коэффициента накладных расходов будем брать в размере 16%.

$$Z_{\text{накл}} = (30385,43 + 46441,26 + 5788,69 + 10458,23) * 0,16 = 14891,8$$

5.2.10 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы (темы) является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции.

Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в таблице 21.

Таблица 21 – Общий бюджет научно-исследовательского проекта

Наименование статьи	Сумма, руб.	Примечание
1. Материальные расходы	30385,43	Пункт 5.1
2. Основная ЗП	46441,26	Пункт 5.2
3. Дополнительная ЗП	5788,69	Пункт 5.3
4. Отчисления	10458,23	Пункт 5.4
5. Накладные расходы	14891,8	16% от суммы ст. 1-4
6. Бюджет затрат НИИ	107965,41	Сумма ст. 1- 5

Общая сумма бюджета затрат проекта составила 107965,41 рублей.

5.3 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Интегральный показатель финансовой эффективности научного исследования получают в ходе оценки бюджета затрат научного исследования (см. табл. 11). Для этого наибольший интегральный показатель реализации технической задачи принимается за базу расчета (как знаменатель), с которым соотносятся финансовые значения.

Интегральный финансовый показатель разработки определяется как:

$$I_{\text{финр}} = \frac{\Phi_p}{\Phi_{\text{max}}}, \quad (5.2)$$

где $I_{\text{финр}}$ – интегральный финансовый показатель разработки;

Φ_p – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Максимальная стоимость составляет 80000 рублей, следовательно:

$$I_{\text{финр}} = \frac{107965,41}{80000} = 1,35$$

Полученная величина интегрального финансового показателя разработки составила 1,35, что отражает соответствующее численное удешевление стоимости разработки в размах.

Интегральный показатель ресурсоэффективности исполнения объекта исследования был определен следующим образом:

$$I_p = \sum a * b, \quad (5.3)$$

где I_p – интегральный показатель ресурсоэффективности для i -го варианта исполнения разработки;

a – весовой коэффициент;

b – бальная оценка, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Расчет интегрального показателя ресурсоэффективности приведен в таблице 22:

Таблица 22 – Сравнительная оценка характеристик вариантов исполнения проекта

Критерии / Объект исследования	Весовой коэффициент параметра	Оценка выполнения
1. Улучшение производительности труда заказчика	0,3	5
2. Функциональная мощность	0,20	5
3. Удобство в эксплуатации	0,15	4
4. Потребность в ресурсах памяти	0,20	5
5. Надежность	0,15	4
ИТОГО	1	

$$I_p = 5*0,3+5*0,20+4*0,15+5*0,2+4*0,15 = 1,5+1+0,6+1+0,6=4,7;$$

Интегральный показатель эффективности вариантов исполнения разработки ($I_{исп}$) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{исп} = \frac{I_p}{I_{финр}} \quad (5.4)$$

$$I_{исп} = \frac{4,7}{0,84} = 5,6$$

Полученное значение интегрального показателя эффективности исполнения разработки превысил максимальный балл в системе оценивания. Это говорит о том, что результат работы можно считать положительным, так

как оценка интегрального показателя ресурсоэффективности близка к максимальной.

5.4 Общий вывод по разделу

Таким образом, в рамках оценки коммерческой ценности разрабатываемого продукта был проведен ряд исследований, включающий в себя следующие разделы:

- оценка коммерческого потенциала и перспективности проведения научных исследований – выполнен анализ конкурентных технических решений, а также SWOT–анализ проекта;
- поиск возможных альтернатив проведения исследований и выбор наиболее оптимального варианта решения поставленной задачи;
- планирование научно-исследовательских работ – определена структура работ в рамках исследования, участники каждой работы, их продолжительность и построен график проведения научных исследований.
- определение ресурсной, бюджетной, социальной и экономической эффективности исследования – выявлен наиболее эффективный вариант исполнения.

В целом данные, полученные при анализе, позволяют сделать вывод, что разработка медицинской информационной системы является перспективной и привлекательной для инвесторов. Особым преимуществом данного МИС является то, что он может рассчитывать лейкоцитарные индексы пациентов, а также визуализировать данные на основе общей оценки состояния пациента.

Общий бюджет НТИ составил 107965,41 рублей. Он включает в себя затраты на основную и дополнительную заработную плату работников и т.д. и т.п.

6 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

ВВЕДЕНИЕ

В данном разделе рассмотрены вопросы, связанные с организацией рабочего места исследователя в процессе написания ВКР в соответствии с нормами производственной санитарии, техники безопасности и охраны труда и окружающей среды. Рабочим местом является 204 аудитория Кибернетического Центра ТПУ, рабочей зоной является компьютерный стол.

Научно-исследовательская работа представляет собой создание медицинской информационной системы по работе с медсправками общего анализа крови. Данную МИС следует отнести к медицинской отрасли, где наиболее часто работают с медицинскими справками общего анализа крови. Данная работа предполагает большой объем работы с ПК, поэтому важным критерием безопасности является организация рабочего места и режима трудовой деятельности.

6.1 Правовые и организационные вопросы обеспечения безопасности

В данном разделе рассмотрены вопросы, связанные с организацией рабочего места исследователя в процессе написания ВКР в соответствии с нормами производственной санитарии, техники безопасности и охраны труда и окружающей среды. Рабочим местом является 204 аудитория Кибернетического Центра ТПУ, рабочей зоной является компьютерный стол.

Характеристика помещения:

- ширина рабочего помещения 6 м, длина – 6 м, высота – 2,8 м;
- площадь – 36 м²;
- объём помещения - 100,8 м³;
- имеется кондиционер, а также естественная вентиляция: двери, окна;
- искусственное освещение;
- естественное освещение.

В данном помещении оборудовано десять рабочих мест, но одновременно в работе обычно задействованы 3-4 человека. Следовательно, в среднем на одного сотрудника приходится не менее 25 м³ объема помещения и не менее 9 м² площади, что удовлетворяет требованиям санитарных норм, согласно которым для одного работника должны быть предусмотрены площадь величиной не менее 6 м² и объем не менее 24 м³, с учетом максимального числа одновременно работающих в смену.

6.1.1 Организационные мероприятия при компоновке рабочей зоны исследования

В процессе работы, все используемые предметы должны находиться в зоне досягаемости. Оптимальное размещение предметов труда и документации в зонах досягаемости приведено на рисунке 39.

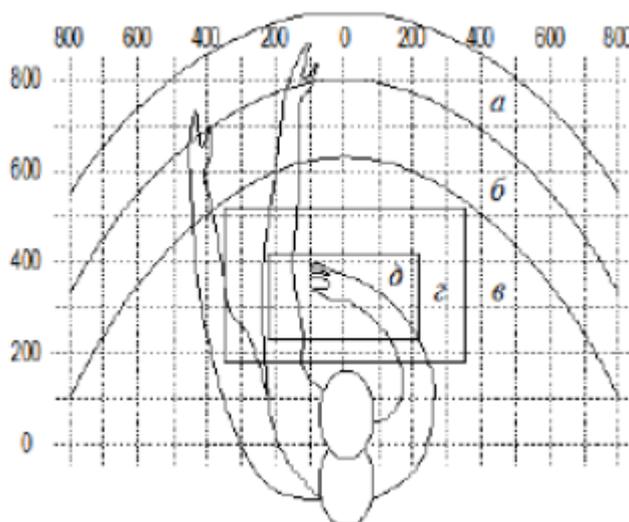


Рисунок 39– Оптимальное размещение предметов труда

На рисунке 39 введены следующие обозначения: а – зона максимальной досягаемости, б – зона досягаемости пальцев при вытянутой руке, в – зона легкой досягаемости ладони, г – оптимальное пространство для грубой ручной работы, д – оптимальное пространство для тонкой ручной работы.

Для повышения комфорта в процессе работы за персональным компьютером, следует соблюдать нормы и требования, изложенные в СанПиН 2.2.2/2.4.1340-03.

Когда характер работы требует постоянного взаимодействия с видео дисплейными терминалами (набор текстов или ввод данных и т.п.) с напряжением внимания и сосредоточенности, при исключении возможности периодического переключения на другие виды трудовой деятельности, не связанные с персональным компьютером, рекомендуется организация перерывов на 10-15 мин через каждые 45-60 мин работы.

6.2 Производственная безопасность

Перечень опасных и вредных факторов представлен в таблице 23.

Таблица 23 – Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015)	Этапы работ			Нормативные документы
	Разрабо тка	Изготов в	Эксплу а	
1.Отклонение показателей микроклимата	+	+	+	1. ГОСТ 12.0.003-2015 2. СанПиН 2.2.4.548-96 3. ГОСТ 12.1.006–84 4. СанПиН 2.2.1/2.1.1.1278-03 5. СанПиН 2.2.2/2.4.1340-03
2. Превышение уровня шума		+	+	
3.Отсутствие или недостаток естественного света	+	+	+	
4.Недостаточная освещенность рабочей зоны		+	+	
5.Психофизиологические факторы: • эмоциональные перегрузки; • умственное перенапряжение; • монотонность труда; • перенапряжение зрения.	+	+	+	
6.Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека				

6.2.1 Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при проведении исследований

6.2.1.1 Отклонение показателей микроклимата

Гигиенические нормативы на параметры микроклимата в рабочей зоне представлены в таблице 2 и таблице 3.

Согласно СанПиН 2.2.4.548-96 выполняемая работа относится к категории легкая (1б) – интенсивность энергозатрат в пределах 121-150 ккал/час, это работы сидя, стоя или связанные с ходьбой с некоторым физическим напряжением таблица 2.

Таблица 24 – Оптимальные величины показателей микроклимата на рабочих местах производственных помещений

Период года	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	Температура воздуха, °С	Температура поверхностей, °С
Холодный	60-40	0,1	21 - 23	20 – 24
Теплый	60-40	0,1	23-25	22-26

Оптимальные величины микроклимата показаны в таблице 25.

Таблица 25 – Допустимые величины показателей микроклимата

Период года	Относительная влажность воздуха, %	Скорость движения воздуха, м/с		Температура воздуха, °С		Температура поверхностей, °С
		для диапазона температур воздуха ниже оптимальных величин, не более	для диапазона температур воздуха выше оптимальных величин, не более	диапазон ниже оптимальных величин	диапазон выше оптимальных величин	
Холодный	15 - 75	0,1	0,2	19,0 - 20,9	23,1 – 24,0	18,0 - 25,0
Теплый	15 - 75	0,1	0,3	20,0 - 21,9	24,1 - 28,0	19,0 - 29,0

В 204 аудитории в холодный период температура поверхностей и температура воздуха составляет 22°С и 23°С, а влажность воздуха 45%. В теплый период температура поверхностей и температура воздуха составляет 24°С и 25°С соответственно. Сравнивая со значениями из таблицы, отклонений от норм не обнаружено.

В итоге температура воздуха и температура поверхностей составляют 22°C и 21°C влажности 45% в холодный период года; 24°C и 23°C при относительной влажности воздуха 50% в теплый период года, что соответствует нормам.

6.2.1.2 Отсутствие или недостаток естественного света и недостаточная освещенность рабочей зоны

Работая при освещении плохого качества или низких уровней, люди могут ощущать усталость глаз и переутомление, что приводит к снижению работоспособности. В ряде случаев это может привести к головным болям. Причинами во многих случаях являются слишком низкие уровни освещенности, слепящее действие источников света и соотношение яркостей, которое недостаточно хорошо сбалансировано на рабочих местах. Головные боли также могут быть вызваны пульсацией освещения, что в основном является результатом использования электромагнитных пускорегулирующих аппаратов (ПРА) для газоразрядных ламп, работающих на частоте 50 Гц.

В компьютерных комнатах должно быть, как естественное, так и искусственное освещение. Естественное освещение обеспечивается за счет оконных проемов, коэффициент искусственного освещения (КОЕ) которых должен быть не менее 1,2% в местах, где имеется снежный покров и не менее 1,5% на остальной территории. Свет из окна должен падать с левой стороны от пользователя. Естественное освещение в аудитории осуществляется через два оконных проема размером 1 на 1.35 метра в наружной стене. Нормируемые показатели естественного, искусственного и совмещенного освещения в соответствии с СанПиН 2.2.1/2.1.1.1278-03 указаны в таблице 26.

Таблица 26 – Нормируемые показатели естественного, искусственного и совмещенного освещения.

Помещения	Рабочая поверхность и плоскость нормирования КЕО и	Естественное освещение		Совмещенное освещение		Искусственное освещение			
		КЕО е н, %		КЕО е н, %		Освещенность, лк			
		При верхнем или комбинированном	При боковом освещении	При верхнем или комбинированном	При боковом освещении	При комбинированном освещении	При общем освещении	Показатель диск	Коэффициент пульс

	освещенности и высота плоскости и над полом, м	освещении	нии	освещении	нии	всего	от общего		омфорта, М, не более	ации освещенности, К _п , %, не более
1	2	3	4	5	6	7	8	9	10	11
Кабинеты, рабочие комнаты	Г – 0,8	3,0	1,0	1,8	0,6	400	200	300	40	15
Помещения для работы с дисплеями и видеотерминалами, залы ЭВМ	Г – 0,8 Экран монитора : В – 1,2	3,5 -	1,2 -	2,1 -	0,7 -	500 -	300 -	400 200	15 -	10

Для того чтобы производственное освещение в помещении со временем не ухудшалось, нужно не менее двух раз в год мыть стекла и светильники, а также следить за работой светильников и при необходимости менять вышедшие из строя лампы.

Когда естественного освещения недостаточно, необходимо использовать искусственное освещение. В качестве основных источников искусственного освещения используются лампы белого и дневного света ЛБ-20 и ЛД-20.

В помещении, в котором проводилась работа, используются потолочные светильники с люминесцентными лампами, которые расположены в ряд. В результате анализа освещенности рабочего места отклонений от норм выявлено не было. Уровень освещенности соответствует нормам в разные периоды светового дня.

Проведём расчет освещения производственного помещения.

Рассматриваемое помещение имеет светлый цвет потолков и стен, серое покрытие пола. Длина помещения (а) – 6 м., ширина (b) – 6 м., высота (h) – 2,8 м. В качестве источника света используются светильники, каждый из которых содержит по n=4 люминесцентные лампы мощностью 18 Вт.; общая яркость светового потока (Φ) 1150 Лм.

Помещение предназначено для работы за персональным компьютером, поэтому нормой освещенности (E) для него согласно СНиП 23-05-95 станет

200-300 Лк, рабочая плоскость стола находится на расстоянии (h_1) 0,8 м. над уровнем пола, коэффициент запаса (k_3) равняется 1,4, а коэффициенты отражения: для потолка – 0,7; для стен – 0,5; для пола – 0,3.

Сначала находим площадь помещения (S): $6*6=36 \text{ м}^2$.

Далее находим индекс помещения по следующей формуле:

$$\frac{S}{(h - h_1) * (a + b)} = \frac{36}{(2,8 - 0,8) * (6 + 6)} = 1,5$$

Теперь на основании показателей отражения поверхностей и высчитанного индекса можно из таблицы определить коэффициент использования ($k_{исп}$). В данном случае он равняется 64.

Определим необходимое количество светильников по формуле:

$$N = \frac{E * S * 100 * k_3}{U * n * \Phi} = \frac{300 * 36 * 100 * 1,4}{64 * 4 * 1150} = 1,5 \approx 6$$

В помещении, в котором проводилась работа, используются потолочные светильники с люминесцентными лампами, которые расположены в ряд. Проведенные расчеты показали, что минимальное число светильников должно быть равно 6. В результате анализа освещенности рабочего места отклонений от норм выявлено не было. Уровень освещенности соответствует нормам в разные периоды светового дня.

6.2.1.3 Превышение уровня шума

В учебной аудитории №204 исходя из-за большого количества оборудования, действует постоянный шум (ГОСТ 12.1.003–2014). Шум в помещении создается кондиционером и вентиляторами для охлаждения нагреваемых частей ЭВМ.

Уровень шума существенно влияет на качество выполняемой работы. Шум ухудшает условия труда, оказывая вредное действие на организм человека. Работающие в условиях длительного шумового воздействия

испытывают раздражительность, головные боли, головокружение, снижение памяти, повышенную утомляемость, понижение аппетита, боли в ушах и т. д.

Для снижения шума в офисных помещениях следует использовать для стен и потолков звукопоглощающую облицовку. Для снижения шума вентилятора следует выбирать агрегат с наименьшими удельными показателями звуковой мощности, а также ограничивать скорость движения воздуха в сетях величиной, обеспечивающей допустимый уровень шума.

6.2.1.4 Психофизиологические факторы

Во время долгой работы за ПК человек также может подвергаться воздействию психофизиологических факторов, таких как умственное перенапряжение, эмоциональные перегрузки и т.д.

Эмоциональные перегрузки изменяют функциональное состояние центральной нервной системы, что в итоге может негативно отразиться на состоянии организма человека в целом. Они могут быть вызваны необходимостью выполнения большого объема работы, конфликтными или стрессовыми ситуациями.

Умственное перенапряжение может наступать вследствие отсутствия необходимого времени на отдых после продолжительной работы. Оно может накапливаться и приводить к возникновению заболеваний.

Отличительными признаками монотонной работы служат однообразие рабочих действий, их многократное повторение и небольшая длительность. Монотонная работа отрицательно сказывается на эффективности производства: ухудшаются экономические показатели, повышается аварийность, травматизм, растет текучесть кадров.

Для снижения эмоциональных перегрузок и умственных перенапряжений предусмотрены перерывы в работе, возможность выбора удобного времени для выполнения работы. Для уменьшения рисков возникновения последствий от фиксированной рабочей позы установленное в аудитории оборудование имеет регулировки: стул регулируется по высоте и

наклону спинки, монитор позволяет подобрать наклон под индивидуальные особенности человека.

6.2.2 Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения

6.2.2.1 Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека;

Помещение, в котором расположены персональные компьютеры, относится к помещениям без повышенной опасности, потому что отсутствуют следующие факторы:

- высокая температура;
- токопроводящая пыль;
- токопроводящие полы;
- сырость;
- возможность одновременного прикосновения человека к имеющим

соединение с землёй металлоконструкциям зданий, технологическим аппаратам и механизмам, и металлическим корпусам электрооборудования.

Мероприятия, направленные на предотвращение возможности поражения электрическим током, включают в себя следующее:

- при выполнении монтажных работ необходимо использовать только исправно работающий инструмент, аттестованный службой КИПиА;
- заземление корпусов приборов и инструментов, которое поможет защитить от поражения электрическим током, который может возникнуть между корпусом приборов и инструментом при пробое сетевого напряжения на корпус;
- запрет на выполнение работ на задней панели при включенном сетевом напряжении;
- выполнение работ по устранению неисправностей должно производиться компетентными людьми.

- нужно постоянно наблюдать за исправностью электропроводки и в случае обнаружения неисправностей незамедлительно принимать действия по их устранению.

Перед началом работы необходимо проверить, чтобы не было свешивающихся со стола или висящих под столом проводов электропитания, убедиться в целостности вилки и провода электропитания, в отсутствии видимых повреждений аппаратуры.

Токи статического электричества, которые могут возникнуть в процессе работы компьютера на корпусах системного блока, монитора и клавиатуры, могут провоцировать разряды при прикосновении к этим элементам, которые не представляют опасности для человека, но могут привести к поломке компьютера. Для уменьшения величин токов статического электричества применяются нейтрализаторы, увлажнение воздуха, используются покрытия полов с антистатической пропиткой.

6.3 Экологическая безопасность

Охрана окружающей среды заключается в устранении отходов жизнедеятельности человека и бытового мусора. Если персональные компьютеры теряют свою работоспособность, их списывают и отправляют на специализированный склад, на котором уже принимаются меры по утилизации техники и комплектующих.

Рассмотрим загрязнение литосферы в результате исследовательской деятельности бытовым мусором, на примере люминесцентных ламп. Их эксплуатация требует осторожности и четкого выполнения инструкции по обращению с данным отходом (код отхода 35330100 13 01 1, класс опасности – 1). В данной лампе содержится опасное вещество ртуть в газообразном состоянии. При не правильной утилизации, лампа может разбиться и пары ртути могут попасть в окружающую среду. Вдыхание паров ртути может привести к тяжелому повреждению здоровья.

При перегорании ртутьсодержащей лампы (выходе из строя) её замену осуществляет лицо, ответственное за сбор и хранение ламп (обученное по электробезопасности и правилам обращения с отходом). Отработанные люминесцентные лампы сдаются только на полигон токсичных отходов для захоронения. Запрещается сваливать отработанные люминесцентные лампы с мусором.

Бытовой мусор помещений организаций несортированный, образованный в результате деятельности работников предприятия (код отхода 91200400 01 00 4). Агрегатное состояние отхода твердое; основные компоненты: бумага и древесина, металлы, пластмассы и др. Для сбора мусора рабочее место оснащается урной. При заполнении урны, мусор выносится в контейнер бытовых отходов. Предприятие заключает договор с коммунальным хозяйством по вывозу и размещению мусора на организованных свалках.

Объекты исследования, а именно электронные медицинские справки и разрабатываемая МИС, не могут угрожать загрязнению окружающей среды т.к. не являются материальными.

6.4 Безопасность в чрезвычайных ситуациях

6.4.1 Анализ вероятных ЧС, которые может инициировать объект исследований

Разрабатываемый объект не может привести к ЧС, т.к. не является материальным.

6.4.2 Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований

Чрезвычайными ситуациями в подобных помещениях могут быть пожары. Основы пожарной безопасности определены по ГОСТ 12.1.004-91 и ГОСТ 12.1.010-76.

Рабочее место, где будет использоваться данная разработка, относится к категории В.

Причинами пожара могут быть: токи короткого замыкания, электрические перегрузки, выделение тепла, искрение в местах плохих контактов при соединении проводов, курение в неположенных местах.

6.4.3 Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС

Пожарная безопасность объекта должна обеспечиваться системами предотвращения пожара и противопожарной защиты, в том числе организационно-техническими мероприятиями.

Должны быть приняты следующие меры противопожарной безопасности:

- обеспечение эффективного удаления дыма, т.к. в помещениях, имеющих оргтехнику, содержится большое количество пластиковых веществ, выделяющих при горении летучие ядовитые вещества и едкий дым;
- обеспечение правильных путей эвакуации;
- наличие огнетушителей и пожарной сигнализации;
- соблюдение всех противопожарных требований к системам отопления и кондиционирования воздуха.

Помещение (КЦ НИ ТПУ) оборудовано пожарными извещателями, которые позволяют оповестить дежурный персонал о пожаре. В качестве пожарных извещателей в помещении устанавливаются дымовые фотоэлектрические извещатели типа ИДФ-1 или ДИП-1.

Выведение людей из зоны пожара должно производиться по плану эвакуации.

Установлены правила поведения людей, порядок и последовательность действий в условиях чрезвычайной ситуации по п. 3.14 ГОСТ Р 12.2.143-2009.

Согласно Правилам пожарной безопасности, в Российской Федерации (ПОСТАНОВЛЕНИЕ ПРАВИТЕЛЬСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ

от 25 апреля 2012 года N 390 О противопожарном режиме (с изменениями на 7 марта 2019 года)) в зданиях и сооружениях (кроме жилых домов) при

единовременном нахождении на этаже более 10 человек должны быть разработаны и на видных местах вывешены планы (схемы) эвакуации людей в случае пожара.

Необходимыми действиями в результате возникшей ЧС и мерами по ликвидации её последствий являются:

1. Передать сигнал «Тревога» голосом, задействовать систему оповещения людей о пожаре.
2. Сообщить по телефону 01, с сотового 010 адрес объекта, место возникновения пожара, свою фамилию. Сообщить по телефону 03, с сотового 030 адрес объекта, что случилось, информацию о пострадавших, свою фамилию, оказать помощь пострадавшим.
3. Открыть все эвакуационные выходы, направить людей к эвакуационным выходам согласно знакам направления движения.
4. Отключить от электропитания оборудование, механизмы и т.п., обесточить помещение.
5. По возможности принять меры по тушению пожара используя средства противопожарной защиты.
6. По возможности предотвратить развитие аварии, обозначить место аварии.

6.5 Выводы по разделу

В данном разделе были рассмотрены основные аспекты производственной, экологической и техногенной безопасности. В рамках производственной безопасности были изучены микроклимат производственного помещения, который включает в себя анализ освещенности, шума и электромагнитных полей, и психофизиологические факторы; а также выполнен расчет минимального количества светильников, необходимых для обеспечения необходимого уровня освещенности. Экологическая безопасность сводится к утилизации вредных отходов производства. Техногенная

безопасность заключается в проведении необходимых мер по предотвращению возникновения чрезвычайных ситуаций.

В результате анализа всех факторов, рассматриваемое помещение является полностью безопасным для работы и соответствует нормативам.

ЗАКЛЮЧЕНИЕ

В результате выполнения дипломной работы была сделана программа, которая производит извлечение ключевых слов и их значения предикторов и размерности из медицинских справок результатов ОАК. Обработанные данные сохраняются в БД МИС. Также был разработан основной web-интерфейс для медицинской информационной системы.

Было проведено тестирование разработанного парсера медсправок на контрольных примерах. В результате тестирования была выявлена, приблизительная точность программы по обработке медсправок равна 56,76%. Данный результат показывает, что необходимо еще дорабатывать данное web-приложение.

Также в ходе выполнения работы:

1 Была разработана БД для МИС, которая включает в себя девять таблиц;

2 Была разработана МИС которая включает в себя следующие функции:

- функция «управления данными пациентами пациентов»;
- функция «визуализация анализов результатов»;
- также был разработан профиль врача.

СПИСОК ЛИТЕРАТУРЫ

1. Когаловский М. Р. Перспективные технологии информационных систем. — М.: ДМК Пресс; Компания АйТи, 2003. — 288 с.
2. Лейкоцитарная формула [Электронный ресурс]. – Режим доступа: <https://www.cmd-online.ru/vracham/spravochnik-vracha/leykotsitarnaya-formula/> (дата обращения: 08.03.2019).
3. Python. [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/Python> (дата обращения: 23.04.2019).
4. Язык программирования Python 3 [Электронный ресурс]. URL: <https://pythonworld.ru/> (дата обращения: 27.04.2019).
5. Регулярные выражения [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Регулярные_выражения (дата обращения: 11.05.2019).
6. Википедия. Свободная энциклопедия [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/База_данных(дата обращения: 11.05.2019).
7. Карпова Т. Базы данных. Модели, разработка, реализация. – Спб, 2001.
8. Университет «ИНТУИТ» [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru/studies/courses/42/42/lecture/27193?page=3> (дата обращения: 12.05.2019).
9. Сайт «Первые шаги» [Электронный ресурс]. – Режим доступа: www.firststeps.ru/msoffice/access/r.php?5 (дата обращения: 13.05.2019)
10. Дронов В. А. Django: практика создания Web-сайтов на Python. — М.: БХВ-Петербург, 2016. — 530 с.
11. Анализ на лейкоцитарный индекс интоксикации – норма и отклонения [Электронный ресурс]. – Режим доступа: <https://vseotravleniya.ru/voprosy/indeks-intoksikatsii.html> (дата обращения: 08.03.2019).

12. Принципы оценки эффективности [Электронный ресурс]. – Режим доступа: https://yamedik.org/gematologiya/immunolog_klin_z/pryncipy_ocenky_effektivnosti/ (дата обращения: 08.04.2019).
13. Студопедия [Электронный ресурс]. – Режим доступа: <https://studopedia.info/8-87112.html> (дата обращения: 12.04.2019).
14. Bootstrap [Электронный ресурс]. – Режим доступа: <https://bootstrap-4.ru/> (дата обращения: 13.04.2019).
15. Web-creator [Электронный ресурс]. – Режим доступа: <https://web-creator.ru/articles/django> (дата обращения: 24.04.2019).
16. Википедия. Свободная энциклопедия [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/MariaDB> (дата обращения: 29.04.2019).
17. Data-Driven Documents [Электронный ресурс]. – Режим доступа: <https://d3js.org/> (дата обращения: 08.03.2019).
18. Информационные системы в медицине: Учебное пособие / Н.В.Абрамов, Н.В.Мотовилов, Н.Д. Наумов, С.Н. Черкасов. - Нижневартовск: Изд-во Нижневарт. гуманит. ун-та, 2008. - 171 с.
19. JavaScript: Web API чтение текста и распознавание голоса [Электронный ресурс] / Блог о программировании. – URL: <https://archakov.im/post/javascript-web-api-recognition-and-speech-text.html> (дата обращения: 9.12.2018).
20. Что такое SSL и зачем он нужен [Электронный ресурс] / Colocation, аренда стоек, размещение серверов, сервер хостинг – URL: <http://www.colocat.ru/texts/ssl.html> (дата обращения: 9.12.2018).
21. Django allauth Авторизация через социальную сеть [Электронный ресурс]. – <https://vivazzi.pro/it/allauth/> (дата обращения: 19.05.2019).
22. Яблчанский Н.И., Пилипенко В.А., Кондратенко П.Г. Индекс сдвига лейкоцитов крови как маркер реактивности организма при остром воспалении // Лаб. дело. — 1983. — № 1. — С. 60—61.

23. Богданов Д. Д. Анализ текстовых данных, с целью выделения значимых фрагментов об анализах пациентов / Д. Д. Богданов; науч. рук. С. В. Аксёнов // Молодежь и современные информационные технологии: сборник трудов XVI Международной научно-практической конференции студентов, аспирантов и молодых учёных, 03-07 декабря 2018 г., г. Томск. — Томск: Изд-во ТПУ, 2019. — [С. 30-31].

24. Богданов Д. Д. Анализ текстовых данных, с целью выделения значимых фрагментов в результатах анализов пациентов / Д. Д. Богданов; науч. рук. С. В. Аксёнов // Международная молодежная научная школа «Междисциплинарные исследования: новые форматы и направления в политэтнокультурном образовательном пространстве», 17-20 декабря 2018 г., г. Томск.

25. Богданов Д. Д. Разработка программы голосового ввода в виде web-приложения для эффективного заполнения медицинских карточек пациентов / Д. Д. Богданов; науч. рук. С. В. Аксёнов // XVI Международная конференция студентов, аспирантов и молодых учёных, «Перспективы развития фундаментальных наук» 23-26 апреля 2019 г., г. Томск.

**TECHNIQUE DEVELOPMENT TO PROCESS THE BLOOD TEST
RESULTS FOR GENERAL HEALTH ASSESSMENT**

Студент:

Группа	ФИО	Подпись	Дата
8KM71	Богданов Данила Динарович		

Руководитель ВКР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Аксёнов Сергей Владимирович	к.т.н.		

Консультант – лингвист отделения ИЯ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Диденко Анастасия Владимировна	к.ф.н		

Before developing software, it must be selected a language and development environment. Finally, it the Python programming language and development environment PyCharm, was selected.

1.1 Analysis of key data from medical certificate

Software development needs extraction key data used in medical certificates. That is, we need to create database for all types of cells and their dimensions. Different medical institutions use different designations of cell types and their dimensions in medical certificates. For example, a cell type "гемоглобин" can be written both in Russian and in English "hemoglobin" or may be specified in the abbreviation Hgb. It is also the case with dimensions. For example, hemoglobin dimension may be indicated in г/л, in Gm/100ml etc. Finally, it was obtained an excel file contains all designations of all the cell types from all available medical certificates (Fig. 1).

	B	C	D	E	F	G	H	I	J	K	L	M	N
34	Средняя концентрация гемоглобина в эритроците	MCHC	Средняя концентра	Ср. конц. Hb в эр	mean corpuscular	26-37	г/дл	г/%	g/dl	g/dL	g%, Gms/dl, Gms/%		
35	Ширина распределения эритроцитов по объему, коэффициент	RDW-CV	Ширина распредел	Расп. эрит. по V, RDWc		11,5-14,5	redcelldis %						
36	Ширина распределения объема эритроцитов, стандартное отклонение	RDW-SD	Ширина распредел	Расп. эрит. по V, станд откл		37-54	redcelldis fl	fl	fl	fl			
37	Нормобласты, ядросодержащие эритроциты	nRBC, nucleated red cells, NRCs				0	%						
38	Нормобласты, ядросодержащие эритроциты	nRBC, nucleated red cells, NRCs				0	абс						
39	Нормобласты, ядросодержащие эритроциты	nRBC, nucleated red cells, NRCs				0	кл/100 лейкоц.						
40													
41	ВИДЫ ЛЕЙКОЦИТОВ										Для 3-дифф.	Для 5-дифф.	Микроскопия
42	Относительные значения	%											
43	Абсолютные значения	#	abs	абс									
44	Относительные значения												
45	Миелобласты	M_BL	Myeloblast			0	%						
46	Промиелоциты	PROM	Promyelocytes			0	%						
47	Миелоциты	MIEL	Myelocytes, myelos			0	%						
48	Метамиелоциты	META	Metamyelocytes, ju	Юные		0	%						
49	Палочкоядерные нейтрофилы	BAN	Band, stabs, bands	Wand-nuclear ne	Палочкоядерные	0-5	%						
50	Сегментоядерные нейтрофилы	SEG	Segmented neutrophils, segs, polymo	Сегментоядерные		40-70	%						
51	Эозинофилы	EO	Eosinophils	EOS, Eos, Eo, eos		0-5	%						
52	Базофилы	BASO	Basophils	BA, BAS, BASO, basos		0-0,5	%						
53	Моноциты	MONO	Monocytes	MO, MON, monos		1-8	%						
54	Лимфоциты	LYMPH	Lymphocytes	LY, LYM, LYMP, LYMF, lymphs		19-37	%						
55	Плазматические клетки	PLASM	Плазматocytes	Plasma cells		0-1	%						
56	Пролимфоциты	Pro_LY	Prolymphocytes			0	%						

Figure 40 – Excel file with key medical data

1.2 Creating files for store key data

Creating data is best stored in separate files. Because if we create an array and write there all our key data, the file size **main_lib.py** will increase several times. It is the program will run slower, and it will be harder for the developer to edit and add new data to the array.

The software uses “blood” folder to store the data. This folder is in the directory **\content\data**. This file tree contains a meaning.

Content folder is the folder where all text medical certificates are stored, as well as key data. Data folder is the folder where folder names, specific key medical data can be stored (For example, the blood folder stores all key blood data, etc.).

This implementation of the file tree is due to the fact that the data is stored on a web server in a clear structured manner. This means that any developer who will work with this program can quickly find the files need.

It was decided to separate cell types and dimensions from each other and store in separate **dat** files. The files are called **measureBlood.dat** and **varBlood.dat** (Fig. 2).

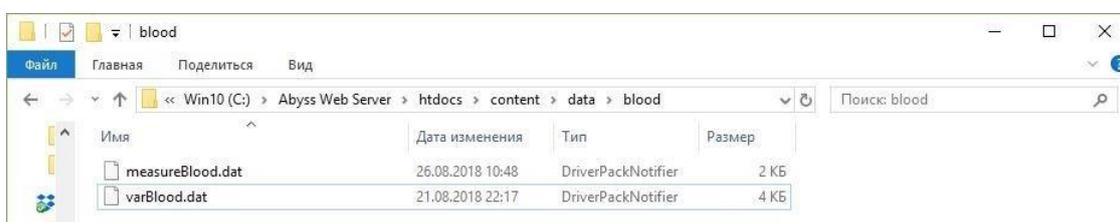


Figure 41– File storage structure in the folder «*blood*»

The use of the .dat extension was due to two reasons. First, the .dat extension bears the meaning. This extension indicates that this file contains some necessary data that is used in the program. Secondly, this extension prevents an inexperienced user from accidentally changing data.

The structure of storing key data inside files with the .dat extension is explained as follows: each key value is distinguished by square brackets and begins with a new line. This is done so that in the future the regular expression can easily pull out all the values and write them into an array. Also, this record structure allows the developer to easily edit and add new values. (Fig. 3, Fig. 4).

```

1 [лейкоциты]
2 [общее количество лейкоцитов]
3 [wbc]
4 [wbcs]
5 [leucocytes]
6 [whitebloodcell]
7 [лимфоциты]
8 [абсолютное значение популяции ср клеток]
9 [гранулоциты]
10 [эритроциты]
11 [общее количество эритроцитов (муж)]
12 [общее количество эритроцитов (жен)]
13 [redbloodcells]
14 [rbc]
15 [rbcs]
16 [eritrocites]
17 [гемоглобин]
18 [гемоглобин (муж)]
19 [гемоглобин (жен)]
20 [гемоглобин (муж)]
21 [гемоглобин (жен)]
22 [hb]
23 [среднее содержание hgb в эритроците]
24 [hgb]
25 [hemoglobin]
26 [гематокрит]
27 [гематокрит (муж)]
28 [ht]
29 [hct]
30 [hematocrit]
31 [гематокрит (жен)]
32 [средний объем эритроцита]
33 [средний объем эритроцитов]
34 [mcv]
35 [ср. объем эритроц.]
36 [сред. объем эритроц.]

```

Figure 42– The storage structure of key data in the file *varBlood.dat*

```

1 [x10?121л]
2 [г/л]
3 [Г/Л]
4 [10^12/л]
5 [x10*9/л]
6 [x10?121л] # русская x
7 [x10?121/л] # русская x
8 [x10*121л] # русская x
9 [x10*121/л] # русская x
10 [x10?121л] # английская x
11 [x10?121/л] # английская x
12 [x10*121л] # английская x
13 [x10*121/л] # английская x
14 [10^9/л]
15 [x10*9/л] # английская x
16 [мм/ч]
17 [пг/кл]
18 [мм/час]
19 [фентолистр]
20 [пикограмм]
21 [абсолютное количество]
22 [относительное количество]
23 [г/л]
24 [г%]
25 [10*12/л]
26 [10*9/л]
27 [мм/ч]
28 [фл]
29 [пг]
30 [nr] # RP иногда неправильно переводит пг
31 [отн]
32 [г/дл]
33 [г/дл] # RP иногда неправильно переводит г/дл
34 [10*12/л]
35 [10*9/л]
36 [fl]

```

Figure 43– The storage structure of key data in the file *measureBlood.dat*

Table 1 presents the detailed ratio of blood cell types to dimensions.

Table 27 – The ratio of blood cell types to dimensions

Cell type	measure
Hemoglobin	g/l
Total red blood cell count	$10^{12}/l$
Total white blood cell count	$10^9/l$
Erythrocyte sedimentation rate	mm/h
Hematocrit	%
Reticulocyte	%
Myeloblasts	%
Promyeloblasts	%
Myelocytes	%
Metamyelocytes	%
Band neutrophils	%
Segmental neutrophils	%
Eosinophils	%
Basophils	%
Monocytes	%
Lymphocytes	%
Plasma cells	%
Prolymphocytes	%
Lymphoblasts	%

1.3 Regular expressions

In order to successfully process and retrieve the necessary data from medical references, it need to use a tool such as regular expressions.

Regular expressions is a formal search language and implementation of manipulations with substrings in the text, based on the use of metacharacters. The search uses a pattern string consisting of characters, metacharacters and specifies the search rule. For text manipulations, the replacement string is additionally specified, which can also contain special characters.

1.3.1 Regular Expression Features

Regular expressions are used by some text editors and utilities for searching and substituting text. For example, using regular expressions, it can set patterns that allow to:

- find all sequences of "cat" characters in any context;
- find the separate word "cat" and replace it with "dog";
- remove from the text all sentences in which the word cat is mentioned.

Regular expressions allow to define much more complex for search or replace patterns.

The result of working with a regular expression can be:

- checking the presence of the desired sample in a given text;
- definition of a substring of text that matches the pattern;
- definition of groups of characters corresponding to individual parts of the sample.

If a regular expression is used to replace text, then the result of the work will be a new text string, which is the source text from which the found substrings are deleted and the replacement strings are substituted new strings.

1.4 Loading key data into an array

The program must have loaded key data (from files with the .dat extension) in the array. In the future, using this array, the program will conduct a comparative

analysis of key data with a medical certificate and also pull out the necessary data using regular expressions.

The create two functions **open_read_varBlood()** and **open_read_measureBlood()**.

The **open_read_varBlood()** function completely reads the **varBlood.dat** file and then using the template (**r '\ [(\ w \ s \ () \ . \ - \ \% \ #] +) \]'**) loads the necessary data into the **varBlood** array. This function loads blood cell types.

The **open_read_measureBlood()** function completely reads the **measureBlood.dat** file and then using the template (**r '\ [(\ w \ s \ () \ . \ - \ \% \ # \ / \ ^ \ * \ = \ : \ ? \ +] +) \]'**) loads the necessary data into the **measureBlood** array. This function loads blood cell types and blood dimensions.

1.5 Extracting necessary data from medical certificate

After loading the key data into the arrays **measureBlood**, **varBlood** and also loading the original certificate into the **content** variable, it must extract the necessary data from the content and write it into the matches **matches_list** and **res**.

Function **extract_data_from_blood()** using a regular expression (**r '\ [r \ n] ([^ \ d \ r \ n] + ?) \ s * ? [r \ n] (\ d * [.,] \ d + \ | \ d +)'**) retrieves from the medical certificate the names of blood types, their numerical values and writes the data into an array of **matches_list**.

The **extract_measure_blood()** function performs a comparative analysis of medical certificate with the **measureBlood** array. If the function finds a match between the array and the certificate, then the function writes the matching dimension to the **res** array.

The **concatenation_result()** function combines the **matches_list** and **res** arrays into a single unit. After connecting the arrays, the function saves the obtained general certificate data into two separate files, **result.txt** and **result.csv** (Fig. 5, Fig. 6).

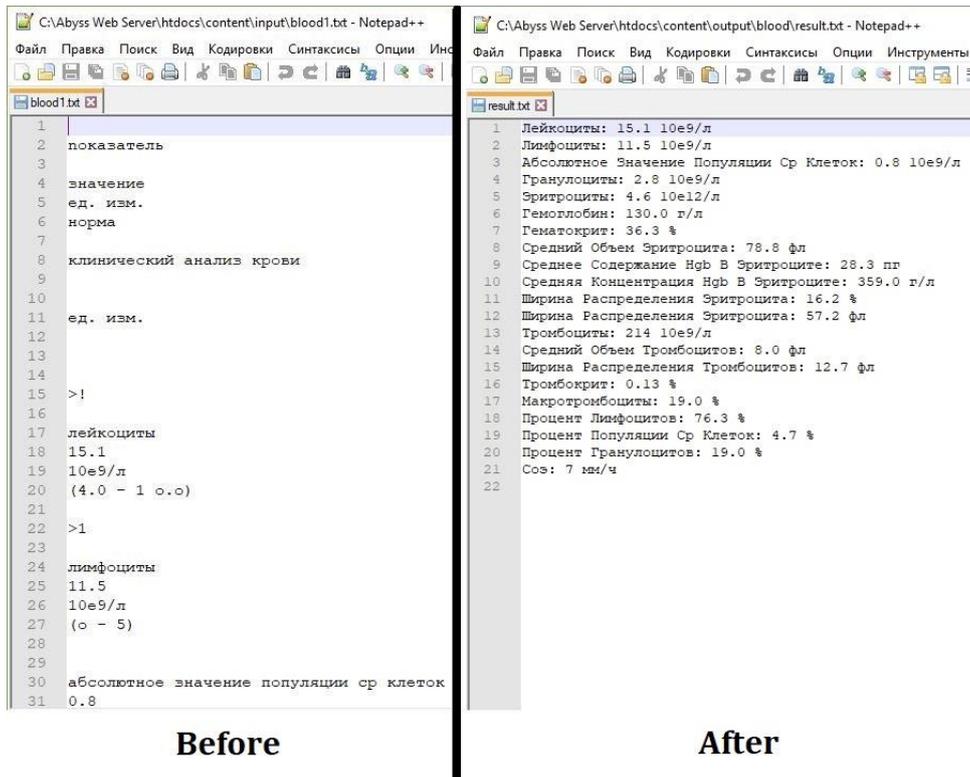


Figure 44– The result of processing the medical electronic certificate in txt format

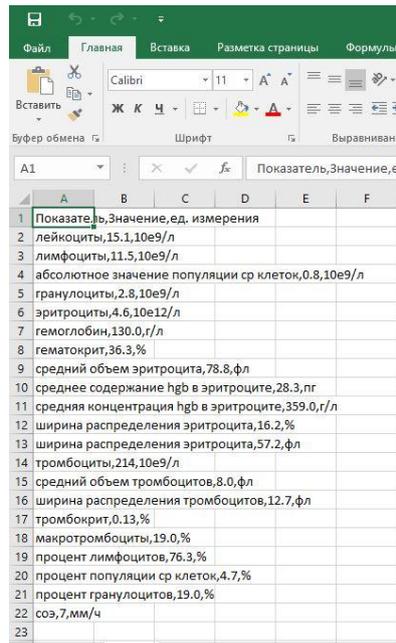


Figure 45– The result of processing a medical electronic certificate in csv format

1.6 Additional scripts in JavaScript

To completely eliminate the random factor of uploading incorrect data to a web server, it must be creating additional scripts in JavaScript. These scripts check client-side downloaded data for some criteria. If the loaded data matches these

criteria, then data is sent to a web server. If the downloaded data does not match at least one criteria, then the browser warns the user that he or she has loaded incorrect data and does not send data to the web server.

1.6.1 Script 1

If the user forgot to upload a text file and tries to submit an empty form, then the program will display a modal window in which it says that the user must select the file to be downloaded. Figure 7 shows the execution of the **empty_form()** function.

```
function empty_form(){ // функция проверяет на наличие пустой формы
    var txt = document.getElementById('file-chooser').value;
    if (txt == '') {
        alert('Выберите файл для загрузки');
        return false;
    }
    return true;
}
```

Function script **empty_form()**

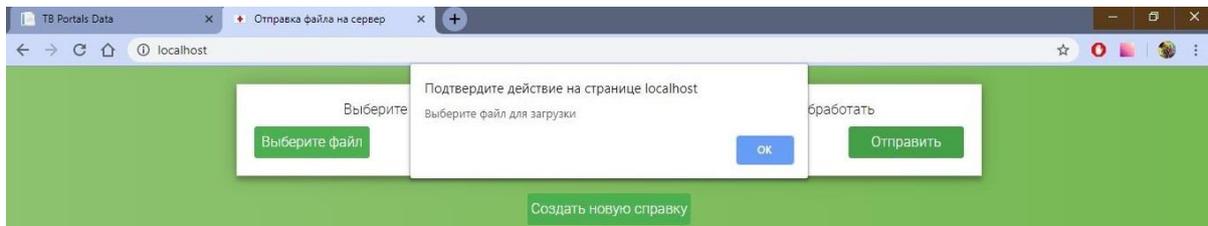


Figure 46– The result work of the **empty_form()** function

1.6.2 Script 2

If the user downloads another file (not a text file) and tries to send the file to a web server, then the program will display a modal window in which it says that the user must select a file with the extension .txt. Figure 8 shows the execution of the **validateTxt()** function.

```
function validateTxt(){ // функция проверяет на наличие расширения .txt
    var uploader = document.getElementById("file-chooser");
```

```

    var fileUrl = uploader.value;

    var parts, ext = ( parts = fileUrl.split("/").pop().split(".") ).length
    > 1 ? parts.pop() : "";

    if (ext !== 'txt'){ //очищает заполненное поле ввода типа file, если
загружен файл не с расширением .txt

        alert( "ОШИБКА. Файл должен быть с расширением .txt Например:
Имя_файла.txt ");

        $("#form-upload")[0].reset();

    }
}

```

Function script **validateTxt()**

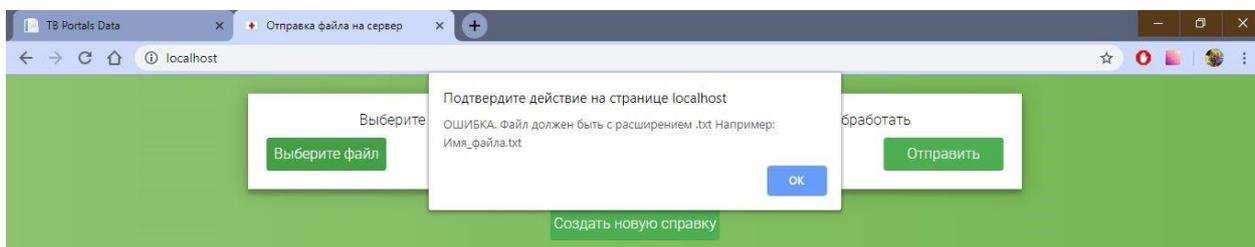


Figure 47– The result work of the **validateTxt()** function

1.6.3 Script 3

If a user uploads an empty text file and tries to send the file to a web server, then the program displays a modal window, which says that the user must load a non-empty text file. Figure 9 shows the execution of the **empty_file()** function.

```

function empty_file() { // функция проверяет на пустоту текстового файла

    var fileInput = $("#file-chooser")[0];

    var size = Number(fileInput.files[0].size); // Size returned in bytes.

    if (size == 0) {

        alert("ОШИБКА. Вы пытаетесь загрузить пустой txt файл. Пожалуйста,
загрузите текстовый файл медицинской справки");

        $("#form-upload")[0].reset();

        return false;

    }

    return true;

}

```

Function script **empty_file()**

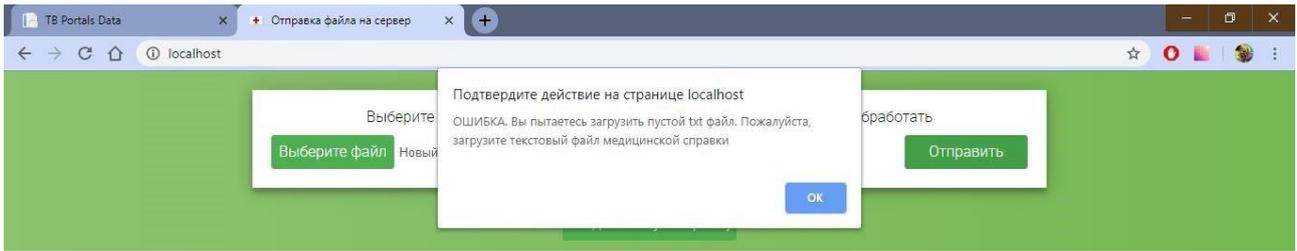


Figure 48– The result work of the `empty_file()` function

Листинг 1. Файл file-upload.js

```

function validateTxt(){ // функция проверяет на наличие расширения .txt
    var uploader = document.getElementById("file-chooser");
    var fileUrl = uploader.value;
    var parts, ext = ( parts = fileUrl.split("/").pop().split(".") ).length >
1 ? parts.pop() : "";
    if (ext !== 'txt'){ //очищает заполненное поле ввода типа file, если
загружен файл не с расширением .txt
        alert( "ОШИБКА. Файл должен быть с расширением .txt Например:
Имя_файла.txt ");
        $("#form-upload")[0].reset();
    }
}

function empty_form(){ // функция проверяет на наличие пустой формы
    var txt = document.getElementById('file-chooser').value;
    if (txt == '') {
        alert('Выберите файл для загрузки');
        return false;
    }
    return true;
}

function empty_file() { // функция проверяет на пустоту текстового файла
    var fileInput = $("#file-chooser")[0];
    var size = Number(fileInput.files[0].size); // Size returned in bytes.
    if (size == 0) {
        alert("ОШИБКА. Вы пытаетесь загрузить пустой txt файл. Пожалуйста, загрузите
текстовый файл медицинской справки");
        $("#form-upload")[0].reset();
        return false;
    }
    return true;
}

// скрипт занимается дублированием пути из Input File в только что добавленное
текстовое поле.
$(document).ready( function() {
    $(".file-upload input[type=file]").change(function(){
        var filename = $(this).val().replace(/.*\\/, "");
        $("#filename").val(filename);
    });
});

```

Листинг 2. Файл main_lib.py

```

#*****
# Файл:      main_lib.py
# Автор:     (с) Данила Богданов
# Дата:      08.05.2018 - создание версии 2.0 web-версия
# Версия:    2.0
# Главная библиотека проекта
#*****

import re

path_varBlood = '../content/data/blood/varBlood.dat'
path_varPee = '../content/data/pee/varPee.dat'
path_measureBlood = '../content/data/blood/measureBlood.dat'

```

```

path_measurePee = '../content/data/blood/measurePee.dat'
path = '../content/input/blood1.txt'
bloodValue = '../content/output/blood/bloodValue.txt'
measure = '../content/output/blood/measure.txt'
res = '../content/output/blood/result.txt'

varBlood = []
measureBlood = []
varPee = []

def open_read_varBlood(): # функция загружает файл varBlood.dat
    file = open(path_varBlood, 'r')
    content = file.read()
    file.close()
    temp = re.findall(r'\[[([\w\s\()\.\-%\#\|\^*\=\:\?+\]]+)\]', content)
    return temp
varBlood = open_read_varBlood()

def open_read_measureBlood(): # функция загружает файл measureBlood.dat
    file = open(path_measureBlood, 'r')
    content = file.read()
    file.close()
    temp = re.findall(r'\[[([\w\s\()\.\-%\#\|\^*\=\:\?+\]]+)\]', content)
    return temp
measureBlood = open_read_measureBlood()

def open_read_varPee(): # функция загружает файл varPee.dat
    file = open(path_varPee, 'r')
    content = file.read()
    file.close()
    temp = re.findall(r'\[[([\w\s\()\.\-%\#\|\^*\=\:\?+\]]+)\]', content)
    return temp
varPee = open_read_varPee()

n = len(varBlood) # определение размера массива blood
m = len(varPee) # определение размера массива pee

def convert_to_lower(path): # подготавливаем файл к чтению
    file = open(path, 'r')
    content = file.read().lower()
    file.close()

    file = open(path, 'w')
    file.write(content)
    file.close()

    print('<p>файл успешно конвертирован и сохранен в файле blood1.txt')
    return content

def open_read(path): # функция открывает и считывает файл
    file = open(path, 'r')
    content = file.read()
    file.close()
    #print(content)
    return content

def save_result(matches_list): # функция сохраняет извлеченные данные
    with open(bloodValue, 'w') as result:
        for name, number in matches_list:
            result.write('{}: {} \n'.format(name.title(), number))
    print('<p>Данные сохранены в файле bloodValue.txt')
    return matches_list

```

```

def concatenation_result(): # функция объединяет и сохраняет общие данные
справки в отдельном файле
    l1 = []
    l2 = []
    with open(bloodValue) as f1:
        for x in f1:
            l1.append(x.rstrip())
    with open(measure) as f2:
        for y in f2:
            l2.append(y)
    with open(res, 'w') as fw:
        for one, two in zip(l1, l2):
            fw.write('{} {}'.format(one, two))
    print('<p>Файлы успешно объединены и сохранены в файле result.txt')
    return 0

def blood_doc(count_blood): # функция классифицирует справку об анализе крови

print('<p>#####')
print('<br>detected анализ крови')

print('<br>#####')
for i in range(n):
    with open(path) as content:
        if varBlood[i] in content.read():
            count_blood += 1
            print('<br>обнаружено слово ' + varBlood[i])
        #else:
            #print('Данное слово '+varBlood[i]+' не обнаружено')
print(' ')
print('<p>найденно совпадений:', count_blood)
print(' ')
return count_blood # возвращает кол-во найденных слов

def pee_doc(count_pee): # функция классифицирует справку об анализе мочи

print('<p>#####')
print('<br>detected анализ мочи')

print('<br>#####')
for i in range(m):
    with open(path) as content:
        if varPee[i] in content.read():
            count_pee += 1
            print('<br>обнаружено слово ' + varPee[i])
        #else:
            #print('Данное слово '+pee[i]+' не обнаружено')
print(' ')
print('<p>найденно совпадений:', count_pee)
print(' ')
return count_pee # возвращает кол-во найденных слов

def who_is_win(count_blood, count_pee): # функция классифицирует вид справки
inf = None
if count_blood > count_pee:
    print('<p>Данный документ является справкой об анализе крови<p>')
    inf = 1
else:

```

```

        print('<p>Данный документ является справкой об анализе мочи<p>')
        inf = 0
    return inf

def extract_data_from_blood(content): # извлечение необходимых данных из файла
в списке
    matches_list =
re.findall(r'[\r\n] ([^\d\r\n]+?)\s*?[\r\n] (\d*[\.,]\d+|\d+)', content,
flags=re.M|re.S)
    for x in matches_list:
        print('<p>{}: {}'.format(x[0], x[1])) # выводим отформатированный
результат в консоль
    return matches_list

def extract_measure_blood(): # функция извлекает размерности для крови
with open(path) as f_in:
    with open(measure, 'w') as f_out:
        for line in f_in:
            # Для удаления справа пустых символов: ' ', '\n', '\r', и т.п.
            line = line.rstrip()
            # Если размерность есть в списке
            if line in measureBlood:
                res = f_out.write('{}\n'.format(line))
                print('<p>{}\n'.format(line))
print('<p>Данные сохранены в файле measure.txt')
return res

#####

count_blood = 0 # счетчик для крови
count_pee = 0 # счетчик для мочи

# content = open_read(path)
# print(content)
# count_blood = blood_doc(count_blood)
# count_pee = pee_doc(count_pee)
# inf = who_is_win(count_blood, count_pee)
# matches_list = extract_data_from_blood(content)
# res_measure = extract_measure_blood()
# save_result(matches_list)
# res_measure = extract_measure_blood()
# concatenation_result()

```

Листинг 3. Файл main_prog.py

```

*****
# Файл:      main_prog.py
# Автор:     (с) Данила Богданов
# Дата:      05.02.2018 - создание главной программы web-версия
# Версия:    2.0
# Главная программа проекта
*****

import main_lib
import re
import cgi # http://lectureswww.readthedocs.io/5.web.server/cgi.html
import os
import sys
import codecs

sys.stdout = codecs.getwriter("utf-8")(sys.stdout.detach())

data = cgi.FieldStorage()

```

```

upload = data[ 'filename' ] # к переменной присваиваем значение полного пути к
загруженному файлу
filename = os.path.basename( upload.filename ) # к переменной присваиваем
краткое имя файла

def uploadFile(filename):
    path = '../content/input/'+filename
    with open( path, 'wb' ) as copy:
        copy.write( upload.file.read() ) # запишем копию загруженного файла
на веб-сервер

    print( 'Content-type:text/html\r\n\r\n' )
    print( '<!DOCTYPE HTML>' )
    print( '<html lang="en">' )
    print( '<head>' )
    print( '<meta charset="UTF-8">' )
    print( '<title>Python Response</title>' )
    print( '</head>' )
    print( '<body>' )
    print( '<h1>Файл Загружен:' , filename , '</h1>' )
    print( '<a href=" ../index.html">Back</a>' )
    print( '</body>' )
    print( '</html>' )
    return path

VarBlood = main_lib.open_read_varBlood() # загружаем файл varBlood.dat
measureBlood = main_lib.open_read_measureBlood() # загружаем файл
measureBlood.dat
VarPee = main_lib.open_read_varPee() # загружаем файл varPee.dat

path = uploadFile(filename)

count_blood = 0 # счетчик для крови
count_pee = 0 # счетчик для мочи

#content = open_read(path) # открываем и считываем файл

content = main_lib.convert_to_lower(path) # конвертируем файл
content = main_lib.open_read(path) # открываем файл

count_blood = main_lib.blood_doc(count_blood) # поиск ключевых слов из массива
varBlood
count_pee = main_lib.pee_doc(count_pee) # поиск ключевых слов из массива varPee

inf = main_lib.who_is_win(count_blood, count_pee) # определяем победителя

matches_list = main_lib.extract_data_from_blood(content) # извлекаем необходимые
данные из файла в списки

main_lib.save_result(matches_list) # сохраняем полученные результаты
res_measure = main_lib.extract_measure_blood() # извлекаем размерности для крови

main_lib.concatenation_result() # объединяем и сохраняем общие данные справки в
отдельный файл

```

Листинг 4. Файл profilePage/models.py

```

from django.db import models
from django.contrib.auth.models import User
from django.db.models.signals import post_save
from django.dispatch import receiver

```

```

class profileDoctors(models.Model):
    SEX_FEMALE = 0
    SEX_MALE = 1
    SEX_CHOICES = (
        (SEX_FEMALE, ('Женщина')),
        (SEX_MALE, ('Мужчина'))
    )
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    middle_name = models.CharField(max_length=50, null=True, blank=True)
    email = models.EmailField(max_length=50, null=True, blank=True)
    bio = models.TextField(max_length=800, blank=True, null=True,)
    birth_date = models.DateField(null=True, blank=True)
    mobile_phone = models.CharField(max_length=30, null=True, blank=True)
    work_phone = models.CharField(max_length=30, null=True, blank=True)
    home_phone = models.CharField(max_length=30, null=True, blank=True)
    home_address = models.CharField(max_length=30, null=True, blank=True)
    avatar = models.ImageField(upload_to='avatar_img/doctors', blank=True,
null=True, default=None, max_length=255)
    sex = models.BooleanField(choices=SEX_CHOICES, blank=True, null=True,)
    education = models.CharField(max_length=80, null=True, blank=True)
    specialization = models.CharField(max_length=50, null=True, blank=True)
    work_experience = models.DateField(blank=True, null=True,
auto_now_add=True, auto_now=False)

    def __str__(self):
        return "id: %s | %s" % (self.id, self.user)

    class Meta:
        verbose_name = 'Профиль'
        verbose_name_plural = 'Профили'

    #Метод который перехватывает сигнал
    @receiver(post_save, sender=User)
    def create_or_update_user_profile(sender, instance, created, **kwargs):
        if created:
            profileDoctors.objects.create(user=instance)
        else:
            instance.profiledoctors.save()

```

Листинг 5. Файл patientsList/models.py

```

from django.db import models
from profilePage.models import profileDoctors

class patientsData(models.Model):
    SEX_FEMALE = 0
    SEX_MALE = 1
    SEX_CHOICES = (
        (SEX_MALE, ('Мужчина')),
        (SEX_FEMALE, ('Женщина'))
    )
    first_name = models.CharField(max_length=50, null=True, blank=False)
    middle_name = models.CharField(max_length=50, null=True, blank=True)
    last_name = models.CharField(max_length=50, null=True, blank=False)
    email = models.EmailField(max_length=50, null=True, blank=True)
    birth_date = models.DateField(null=True, blank=True)
    sex = models.BooleanField(choices=SEX_CHOICES, blank=True, null=True)
    mobile_phone = models.CharField(max_length=30, null=True, blank=True)
    home_phone = models.CharField(max_length=30, null=True, blank=True)
    home_address = models.CharField(max_length=30, null=True, blank=True)
    avatar = models.ImageField(upload_to='avatar_img/patients', blank=True,
null=True, default=None, max_length=255)

```

```

def __str__(self):
    return "id: %s | %s %s %s" % (self.id, self.last_name, self.first_name,
self.middle_name)

class Meta:
    verbose_name = 'Пациент'
    verbose_name_plural = 'Пациенты'

class Care(models.Model):
    patient_id = models.ForeignKey(patientsData, on_delete=models.CASCADE)
    doctor_id = models.ForeignKey(profileDoctors, on_delete=models.CASCADE)
    inspect_date_time = models.DateTimeField(null=True, blank=True)
    complaints = models.CharField(max_length=300, null=True, blank=True)
    history_disease = models.CharField(max_length=300, null=True, blank=True)
    anamnesis_life = models.CharField(max_length=300, null=True, blank=True)
    anamnesis_vte = models.CharField(max_length=300, null=True, blank=True)
    objective_status = models.CharField(max_length=300, null=True, blank=True)
    local_status = models.CharField(max_length=300, null=True, blank=True)
    diagnosis_admission = models.CharField(max_length=300, null=True, blank=True)
    justif_diagnosis = models.CharField(max_length=300, null=True, blank=True)
    diagnosis = models.CharField(max_length=300, null=True, blank=True)

def __str__(self):
    return "id: %s" % (self.id)

class Meta:
    verbose_name = 'История болезни'
    verbose_name_plural = 'Истории болезней'

class medCertif(models.Model):
    care_id = models.ForeignKey(Care, on_delete=models.CASCADE, null=True,
blank=True)
    patient_id = models.ForeignKey(patientsData, on_delete=models.CASCADE)
    kalf_caliph_index = models.DecimalField(max_digits=6, decimal_places=3,
null=True, blank=True)
    shift_index = models.DecimalField(max_digits=6, decimal_places=3, null=True,
blank=True)
    allergy_index = models.DecimalField(max_digits=6, decimal_places=3,
null=True, blank=True)
    date_time_of_create = models.DateTimeField(null=True, blank=True)
    date_time_of_createE = models.DateTimeField(null=True, blank=True)

def __str__(self):
    return "id: %s" % (self.id)

class Meta:
    verbose_name = 'Медицинская справка'
    verbose_name_plural = 'Медицинские справки'

class typeOfCells(models.Model):
    name = models.CharField(max_length=30, null=True, blank=True)
    fullname = models.CharField(max_length=50, null=True, blank=True)
    description = models.CharField(max_length=300, null=True, blank=True)

def __str__(self):
    return "id: %s" % (self.id)

class Meta:
    verbose_name = 'Тип клетки'
    verbose_name_plural = 'Типы клеток'

```

```

class unitsOfMeasure(models.Model):
    name = models.CharField(max_length=30, null=True, blank=True)
    description = models.CharField(max_length=300, null=True, blank=True)

    def __str__(self):
        return "id: %s" % (self.id)

    class Meta:
        verbose_name = 'Размерность клетки'
        verbose_name_plural = 'Размерности клеток'

```

Листинг 6. SQL-запросы для создания таблицы resultmedcertif и norms

```

CREATE TABLE medinfodb.patientslist_resultmedcertif (
    id int(11) NOT NULL,
    value varchar(30) DEFAULT NULL,
    measure_id_id int(11) NOT NULL,
    type_id_id int(11) NOT NULL,
    PRIMARY KEY (id, type_id_id)
)
ENGINE = INNODB,
CHARACTER SET utf8,
COLLATE utf8_general_ci;

ALTER TABLE medinfodb.patientslist_resultmedcertif
ADD CONSTRAINT FK_patientslist_resultmedcertif_patientslist_medcertif_id FOREIGN
KEY (id)
REFERENCES medinfodb.patientslist_medcertif (id) ON DELETE NO ACTION ON UPDATE
NO ACTION;

ALTER TABLE medinfodb.patientslist_resultmedcertif
ADD CONSTRAINT patientsList_resultm_measure_id_id_e5434bff_fk_patientsL FOREIGN
KEY (measure_id_id)
REFERENCES medinfodb.patientslist_unitsofmeasure (id);

ALTER TABLE medinfodb.patientslist_resultmedcertif
ADD CONSTRAINT patientsList_resultm_type_id_id_4374f77b_fk_patientsL FOREIGN KEY
(type_id_id)
REFERENCES medinfodb.patientslist_typeofcells (id);

CREATE TABLE medinfodb.patientslist_norms (
    type_id int(11) NOT NULL,
    norms_male varchar(40) DEFAULT NULL,
    norms_female varchar(40) DEFAULT NULL,
    measure_id_id int(11) NOT NULL,
    PRIMARY KEY (type_id, measure_id_id)
)
ENGINE = INNODB,
CHARACTER SET utf8,
COLLATE utf8_general_ci;

ALTER TABLE medinfodb.patientslist_norms
ADD CONSTRAINT FK_patientslist_norms_patientslist_typeofcells_id FOREIGN KEY
(type_id)
REFERENCES medinfodb.patientslist_typeofcells (id) ON DELETE NO ACTION ON UPDATE
NO ACTION;

ALTER TABLE medinfodb.patientslist_norms
ADD CONSTRAINT patientsList_norms_measure_id_id_c14db7a4_fk_patientsL FOREIGN
KEY (measure_id_id)
REFERENCES medinfodb.patientslist_unitsofmeasure (id);

```

Листинг 7. Гипертекстовая разметка главной страницы index.html

```
{% extends "base.html" %}
{% load staticfiles%}
{# block head#}
{% block title %}Главная страница{% endblock %}

{#block body#}
{% block navbar %}
    {% include 'includes/navbar.html'%}
{% endblock %}
{% block content %}
    <!-- Services -->
    <div class="services">
        <div class="container">
            <div class="row">
                <div class="col text-center">
                    <div class="section_title">Преимущества</div>
                    <div class="section_subtitle">нашей медицинской
информационной системы</div>
                </div>
            </div>
            <div class="row icon_boxes_row">
                <!-- Icon Box -->
                <div class="col-xl-4 col-lg-6">
                    <div class="icon_box">
                        <div class="icon_box_title_container d-flex
flex-row align-items-center justify-content-start">
                            <div class="icon_box_icon"></div>
                            <div class="icon_box_title">Единая
БД</div>
                        </div>
                        <div class="icon_box_text">Lorem ipsum dolor
sit amet, consectetur adipiscing elit. Donec lorem maximus malesuada lorem
maximus mauris.</div>
                    </div>
                </div>
                <!-- Icon Box -->
                <div class="col-xl-4 col-lg-6">
                    <div class="icon_box">
                        <div class="icon_box_title_container d-flex
flex-row align-items-center justify-content-start">
                            <div class="icon_box_icon"></div>
                            <div class="icon_box_title">Обработка
справок ОАК</div>
                        </div>
                        <div class="icon_box_text">Lorem ipsum dolor
sit amet, consectetur adipiscing elit. Donec lorem maximus malesuada lorem
maximus mauris.</div>
                    </div>
                </div>
                <!-- Icon Box -->
                <div class="col-xl-4 col-lg-6">
                    <div class="icon_box">
                        <div class="icon_box_title_container d-flex
flex-row align-items-center justify-content-start">
                            <div class="icon_box_icon"></div>
                            <div
class="icon_box_title">Визуализация Данных</div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
```



```

</div>
<!-- Departments -->
<div class="departments">
  <div class="container">
    <div class="row">
      <div class="col text-center">
        <div class="section_title">Our Departments</div>
        <div class="section_subtitle">to choose from</div>
      </div>
    </div>
    <div class="row dept_row">
      <div class="col">
        <div class="dept_slider_container_outer">
          <div class="dept_slider_container">
            <!-- Slider -->
            <div class="owl-carousel owl-theme
dept_slider">
              <!-- Slide -->
              <div class="owl-item dept_item">
                <div
class="dept_image"></div>
                <div class="dept_content">
                  <div
class="dept_title">Neonatology</div>
                  <div
class="dept_link"><a href="#">Read More</a></div>
                </div>
              </div>
              <!-- Slide -->
              <div class="owl-item dept_item">
                <div
class="dept_image"></div>
                <div class="dept_content">
                  <div
class="dept_title">Dentistry</div>
                  <div
class="dept_link"><a href="#">Read More</a></div>
                </div>
              </div>
              <!-- Slide -->
              <div class="owl-item dept_item">
                <div
class="dept_image"></div>
                <div class="dept_content">
                  <div
class="dept_title">Orthopedics</div>
                  <div
class="dept_link"><a href="#">Read More</a></div>
                </div>
              </div>
              <!-- Slide -->
              <div class="owl-item dept_item">
                <div
class="dept_image"></div>
                <div class="dept_content">
                  <div
class="dept_title">Laboratory</div>
                  <div
class="dept_link"><a href="#">Read More</a></div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>

```

