

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет

вычислительной техники

Кафедра

«Вычислительная техника»

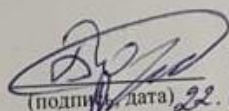
Специальность: 09.05.01 «Применение и эксплуатация
автоматизированных систем специального
назначения»
Специализация: «Эксплуатация вычислительных машин, комплексов,
систем и сетей специального назначения»

ДИПЛОМНЫЙ ПРОЕКТ

на тему

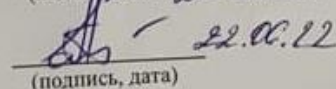
КОРПОРАТИВНЫЙ МЕССЕНДЖЕР ДЛЯ ПРИЕМА И ПЕРЕДАЧИ
ТЕКСТОВЫХ СООБЩЕНИЙ

Студент


(подпись, дата) 22.06.22.

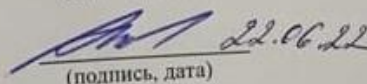
Бородин Сергей Александрович
(ФИО полностью)

Руководитель


(подпись, дата) 22.06.22

Бикташев Р.А.
(фамилия, инициалы)

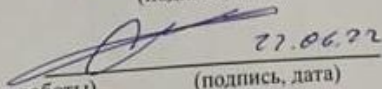
Нормоконтролёр


(подпись, дата) 22.06.22

Бычков А.С.
(фамилия, инициалы)

Рецензент

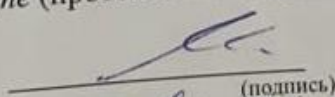
(должность, место работы)


(подпись, дата) 22.06.22

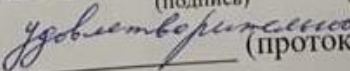
Печерский С.В.
(фамилия, инициалы)

Работа допущена к защите (протокол зас. кафедры от 22.06.22 г. № 13)

Заведующий кафедрой


(подпись)

М.А. Митрохин
(фамилия, инициалы)

Работа защищена с отметкой  (протокол заседания ГЭК от 24 июня 2022, № 10)

Секретарь ГЭК

Е.И. Гурин

Пенза 2022

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет
вычислительной техники

Кафедра
«Вычислительная техника»

«Утверждаю»
Заведующий кафедрой
М.А. Митрохин
«26» апреля 2022 г.

**ЗАДАНИЕ
НА ДИПЛОМНЫЙ ПРОЕКТ**

1. Студент Бородин Сергей Александрович гр. 17В02 факультета вычислительной техники.
2. Тема работы «Корпоративный мессенджер для приема и передачи текстовых сообщений»

Тема утверждена приказом ПГУ №0357/о от "26" апреля 2022 г.

3. Руководитель работы к.т.н., доцент кафедры ВТ ПГУ Бикташев Р.А.

4. Задание на работу (назначение разработки, исходные данные и т.п.)
Разработать систему, олицетворяющую корпоративный мессенджер для приема и передачи текстовых сообщений. Система должна иметь клиент-серверную, либо монолотную архитектуру. Программа должна сочетать в себе серверную часть для работы с логикой и клиентскую часть для взаимодействия с пользователями.

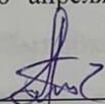
5. Перечень подлежащих разработке вопросов
Требуется произвести анализ предметной области, сформировать техническое задание, выбрать инструменты для разработки приложения, выполнить проектирование системы, разработать серверную и клиентскую часть системы. Серверная часть системы должна выполнять функции: обеспечивать добавление, изменение и удаление информации в базу; обеспечивать обмен сообщениями между пользователями; выполнять авторизацию в системе; сохранять параметры для ввода данных в систему. Клиентская часть необходима для взаимодействия пользователя с системой путем оперирования с UI системой. Также выполнить задание по БЖД и осуществить экономическое обоснование разработки системы.

6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1.	Анализ предметной области	8%	26.04.2022 – 29.04.2022	
2.	Составление технического задания	6%	30.04.2022 – 02.05.2022	
3.	Выбор инструментов разработки	8%	03.05.2022 – 06.05.2022	
4.	Проектирование системы	12%	07.05.2022 – 12.05.2022	
5.	Разработка серверной части системы	40%	13.05.2022 – 01.06.2022	
6.	Выполнение экономического обоснования	10%	02.06.2022 – 06.06.2022	
7.	Выполнение задания по БЖД	8%	07.06.2022 – 10.06.2022	
8.	Оформление пояснительной записки	8%	11.06.2022 – 14.06.2022	

Дата выдачи задания "26" апреля 2022 г.

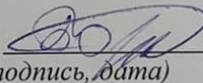
Руководитель работы



26.04.22

(подпись, дата)

Задание к исполнению принял студент



26.04.22

(подпись, дата)

Работу к защите допустить
Декан факультета



А.Р. Феодосов

(подпись, дата)

23.06.2022

Реферат

Пояснительная записка содержит 132 листа, 25 рисунков, 14 таблиц, 7 источников, 5 приложений.

КОРПОРАТИВНЫЙ МЕССЕНДЖЕР, БАЗА ДАННЫХ, ЧАТ, ФРЕЙМВОРК, *YII*, *HTML*, *CSS*, *JAVASCRIPT*, *PHP*.

Цель дипломного проектирования - разработка корпоративного мессенджера, необходимого для приема и передачи сообщений между пользователями.

Графическая часть дипломного проекта содержит следующие чертежи:

- схема данных – 1 лист формата A1;
- диаграмма классов – 1 лист формата A1;
- диаграммы *UML* – 3 листа формата A1.

Разработанная программа была реализована на языках: *PHP*, с использованием фреймворка *Yii*, *HTML*, *CSS*, *JavaScript*.

Для написания базы данных был использован язык *SQL*.

ПГУ 1.090501.02.001 ПЗ				
Изм.	Лист	№ докум.	Подпись	Дата
Разраб.		Бородин С.А.		
Провер.		Бикташев Р.А.		
Н. Контр.		Бычков А.С.		
Реценз.		Петровский С.		21.06
Утверд.		Митрохин М.А.		

Корпоративный мессенджер для приема и передачи текстовых сообщений Пояснительная записка		
Лит.	Лист	Листов
	2	132
ФВТ, каф. ВТ гр. 17В01		

Содержание

Перечень обозначений и сокращений	5
Введение	7
1 Корпоративный мессенджер для приема и передачи текстовых сообщений	9
1.1 Анализ предметной области	9
1.1.1 Формулировка поставленной задачи	9
1.1.2 Изучение предметной области	18
1.1.3 Обзор аналогичных решений	19
1.1.4 Функциональные характеристики	21
1.1.5 Функциональные требования	21
1.2 Проектирование мессенджера	28
1.2.1 Обоснование выбора технологий и средств разработки	28
1.2.2 Описание разработанного программного средства	40
2 Техничко-экономическое обоснование разработки программы	62
2.1 Организационно-правовая характеристика ПП	62
2.1 Планирование разработки	64
2.2 Описание бюджета на разработку проекта	75
3 Защита от ионизирующих излучений работников, обслуживающих автоматизированную систему	82
3.1 Основные характеристики ионизирующих излучений	82
3.2 Нормы радиационной безопасности	89
3.3 Защита от действия ионизирующих излучений	92
Заключение	98
Список использованных источников	101
Приложение А - Листинг программных модулей	102
Приложение Б - Экранные формы	115
Приложение В - Схема данных	123
Приложение Г - Диаграмма классов	125

					ПГУ 1.090501.02.001 ПЗ	Лист
						4
Изм.	Лист	№ докум.	Подпись	Дата		

Перечень обозначений и сокращений

CRUD - (англ. *create, read, update, delete* — «создание, чтение, обновление, удаление») сокращённое именованное 4-х базовых функций, используемых при работе с наборами данных

HTTP - протокол прикладного уровня передачи данных

MS - *Microsoft*

SQL - декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных

VBA - язык программирования, а также интегрированная среда разработки программного обеспечения, разрабатываемые корпорацией *Microsoft*

БД - база данных

ДОА - допустимые среднегодовые объемные активности

ДУА - среднегодовые удельные активности

ЖЦ - жизненный цикл

Зв - зиверт

Зв/год - зиверт в год

ИС - информационная система

кВт - киловатт

км/с – километр в секунду

КэВ - кило-электрон-Вольт

л/мин –литр в минуту

мЗв - миллизиверт

мкЗв -микрозиверт

мкР/с - микрорентген в секунду

мР/ч - миллирентгенах в час

МэВ - мега-электрон-вольт

ОС - операционная система

ПГП - пределы годового поступления

ПД - предел дозы

					ПГУ 1.090501.02.001 ПЗ	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

ПДД - предельно допустимая доза

ПК - персональный компьютер

ПО - программное обеспечение

РФ - Российская Федерация

См - сантиметр

СУБД - система управления базами данных

ТО - технологическое обеспечение

УСД - унифицированная система документации

ФОТ - фонд оплаты труда

эВ - электронвольт

юзабилити - способность продукта быть понимаемым, изучаемым, используемым и привлекательным для пользователя в заданных условиях

					<i>ПГУ 1.090501.02.001 ПЗ</i>	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

Введение

В первую очередь стоит заметить, что актуальность темы данного исследования обусловлена тем фактом, что по состоянию на сегодняшний день, большинство предприятий реализует довольно много самых разных технологических процессов. Сократить сроки их выполнения можно путем автоматизации данных технологических процессов.

В свою очередь, автоматизация технологических процессов положительно скажется на общей результативности деятельности всего производства в целом. Особое распространение, в наши дни, получило объектное моделирование. Это произошло благодаря стремительному развитию *IT*-сферы, а также появлению новых, инновационных алгоритмов и методик анализа.

Как следствие, повсеместно применяться стало объектное проектирование. Именно объектное проектирование и анализ занимают, по состоянию на сегодняшний день, первое место среди инновационных технологий создания виртуальных моделей систем.

Однако стоит заметить, что, несмотря на стремительное развитие объемного моделирования и проектирования, данная технология еще находится на стадии внедрения и совершенствования.

Уже сейчас повышенная сложность работ по проектированию обнаруживает нехватку классических методов структурного проектирования и анализа. В последнее время, к вопросу автоматизации в экономике стараются подойти непосредственно с точки зрения проектирования и анализа.

Таким образом, пытаются сделать максимально доступными и понятными решения задач, поставленных перед вопросом автоматизации в экономике. Ранее же вышеуказанные вопросы и вовсе не поддавались решению.

В применении объектного анализа системы непосредственно включены вопросы создания объектной модели, как основного составляющего объекта

					<i>ПГУ 1.090501.02.001 ПЗ</i>	Лист
						7
Изм.	Лист	№ докум.	Подпись	Дата		

компонентного проектирования.

Сейчас существует большое количество различных инструментальных средств проектирования. При выборе подходящего инструмента важно учесть будет ли программа распределенной или же требуется, чтобы она была локальной. Важно осознавать полностью тот факт, что современные средства разработки позволяют, с одинаковым успехом, создавать как локальные, так и распределенные программы. В данной работе проектируется и исследуется современное автоматизированное рабочее место [1].

При этом важно учесть тот факт, что такое автоматизированное рабочее место должно стать общей частью информационной системы. При этом, важно заложить основы и возможности выполнения так называемой распределённой работы. В целом, мы осуществляем разработку распределенной системы по эффективному управлению информационными базами данных.

При этом, к выполнению данной разработки можно подходить с двух разных сторон, а именно использовать такие подходы:

Привлечение к работам по разработке инновационных возможностей современных *Internet*-технологий.

Использование в своих изысканиях классических средств по стандартной технологии клиент-сервер.

При этом преимущество первого инновационного подхода заключается в отсутствии необходимости осуществлять работы по программированию клиентской стороны (характеризуется использованием обычного браузера, а также стандартных каналов передачи данных). Что касается второго, классического варианта, то его непосредственные преимущества заключаются в возможности успешной реализации основных бизнес-правил.

Данный дипломный проект посвящен проектированию корпоративного мессенджера.

1 Корпоративный мессенджер для приема и передачи текстовых сообщений

1.1 Анализ предметной области

1.1.1 Формулировка поставленной задачи

Автоматическая информационная система имеет свое информационное обеспечение. В качестве такого информационного обеспечения выступают базы данные и оперативная система. Автоматические информационные системы используют лексические и формальные базы, также в них используются отдельные языковые средства. Они необходимы для того, чтобы вводить данные, осуществлять поиск информации, обрабатывать информацию, а также выводить ее в таком виде, который будет удобен пользователю. Информационное обеспечение — это некий массив, состоящий из документов. Он может быть представлен в двух видах. Первый вид - это не форматированный вид. Второй вид - это форматированный вид. В состав информационного обеспечения также входят:

- классификаторы;
- словари;
- нормативная база.

Это все является неотъемлемой частью информационного обеспечения, они необходимы для того, чтобы автоматическая информационная система работала правильно и позволяла осуществлять все необходимые операции.

В тот момент, когда пользователь обращается к автоматической системе, используются определенные средства и определенные правила, которые необходимы для перевода естественного языка в этот язык, который используется в операционной системе.

Сейчас информационное обеспечение рассматривается как информационное и лингвистическое средства. Что входит в состав информационного обеспечения было рассмотрено ранее, то есть речь идет о

					ПГУ 1.090501.02.001 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		9

файлах и о данных. Непосредственно в состав лингвистического обеспечения входит так называемая форматная база, а также лексические средства языка.

Если говорить об автоматизированных системах, то в них в них чаще всего строятся математические модели и используются математические методы. При этом все автоматизированные системы должны соответствовать стандарту ГОСТ 24.03-90. Предусмотрена также возможность использовать различных прикладных программ. Само понятие информационное обеспечение включает в себя также документацию, которая относится к используемому программному обеспечению. Техническое обеспечение используется для того, чтобы обеспечить стабильную работу системы, оно в себя включает комплекс технических средств, которые позволяют это сделать.

Если рассматривать комплекс технических средств, то он включает в себя:

- средства, которые используются для сбора информации;
- средства, которые используются для передачи данных;
- средства, которые используются для контроля над правильностью информации;
- средства, которые используются для обработки и размножения имеющихся данных.

Ранее уже говорилось о том, что правовое обеспечение состоит в нормативно-правовых документах. Они используются для того чтобы обозначить права и обязанности всех сотрудников, которое осуществляет свою деятельность компании. В них также установлены те функции, которые выполняет данные сотрудники. Документы также используются для того, чтобы бы описать способы хранения информации, правила, по которым проводится проверка информации, правила по которым работает система и правила, по которым выполняются различные операции.

Особое внимание необходимо уделять организационно-методического обеспечению. Она выражается в общности документов, которые описывает структуру автоматизация. Тут стоит отметить, что структура необходимо для

					<i>ПГУ 1.090501.02.001 ПЗ</i>	Лист
						10
Изм.	Лист	№ докум.	Подпись	Дата		

того, чтобы контролировать и описывать очередность тех функций, которые должна выполнять система. Внимание стоит обращать на эргономическое обеспечение, которое состоит из методов и средств, делающих работу автоматизированной информационной системе более оптимальный и удобный.

То методологическое обеспечение, которое используется на сегодняшний день, состоит из средств, методов, а также советов и рекомендаций по использованию тех компонентов, которые включены в состав автоматизированной информационной системы. То есть можно говорить о том, что автоматизированная система - это некий комплекс, который состоит из программных и технологических решений, в них описаны действия, которые помогут более эффективно достичь той цели, которая перед ними ставится. В качестве функции, которые выполняет данная система можно отметить:

- информационные функции;
- управляющие функции;
- вспомогательные функции.

Информационные функции заключается в том, что автоматизированная информационная система дает возможность хранить данные, обрабатывать и собирать их. В качестве дополнительных функций, которые зачастую бывает очень полезным и можно отметить:

- оценка эффективности работы всей системы.
- контрольная работа система.
- возможность рассчитать основные параметры.
- дает возможность вносить изменения в основные параметры.
- выводит результат своей деятельности в таком виде, который будет удобен пользователю.

Так же есть возможность прогнозировать деятельность как отдельных частей системы, так и ее в целом.

Управляющие функции заключается в том, чтобы оказывать воздействие на все субъекты. То воздействие, которое оказывается должно быть логичным, адаптированным и программным. Она должно приводить к тому, что будут регулироваться все основные параметры системы. Можно выделить несколько режимов работы автоматизированных систем. Первый режим работы - это ручной режим. Второй режим работы - это режим диалога. Третий режим работы - это режим советчика [2].

На практике можно столкнуться с тем, что подсистемы информационной системы не работают не так, как необходимо, они не могут выполнять все функции эффективно и корректно. Если рассматривать функцию сбора данных, то она осуществляется следующим образом. Система копирует данных из тех каналов, которые ей предоставлены. Во время сбора данных используются автоматизированные способы обработки информации.

Задача оператора заключается в том, чтобы осуществить первичный сбор данных, а также в последующем распределить все полученные данные. После того как вся информация собрана задача системы заключается в том, чтобы проанализировать всю имеющуюся информацию, также определить основные сущности. Тут стоит обратить внимание на то, что найденные сущности могут выступать в качестве шаблонов для различных таблиц, которые используются в базе данных. Только после того как сущности выделены и обработаны все данные переходят в систему для дальнейшей обработки и хранения. То есть можно сказать о том, что данная подсистема необходима для того, чтобы не только собирать данные, но и производить первичную их обработку, сортировку и распределение. Оператор должен осуществлять тесное взаимодействие с администратором базы данных. Это даст возможность заполнить все поля, и привести систему в соответствии с заранее оговоренного правилами. Также на данном этапе необходимо настроить базу данных, которая будет использоваться для хранения. В случае утери данных, необходимо выполнить процедуру их восстановления.

Управляемой системой СУБД является непосредственно тот стержень, которые есть у рассматриваемой под системы. Во время управления информацией руководство должно предусмотреть возможность того, что система будет собирать данные по тем каналом, которые ей предоставлены. Также необходимо предусмотреть возможность проведения анализа, перенос информации из одной участие базы данных в другую. Особенно это необходимо в том случае, если речь идёт о тех значениях, которые играет большую роль в управлении и в деятельности компании. После того как процесс сбора данных закончен, может быть передано специальным сотрудником для дальнейшей обработки[3].

Сам процесс управления компанией является очень сложной задачей. Выполнить его эффективно можно только в том случае, если у руководителя будут необходимые знания, если он будет пользоваться той инструкцией, которая у него есть.

Если говорить о получении результатов, то сам процесс состоит из несколько этапов. Давайте их отметим:

- на первом этапе происходит сбор информации;
- на втором этапе создаются модели процессов, которые будут использоваться при обработке данных;
- на третьем этапе происходит сам процесс моделирования результатов, также на данном этапе есть возможность прогнозировать события, которые произойдут в будущем.

Для того чтобы эффективно реализовать данную деятельность можно использовать информационные системы. Информационная система - это совокупность программных и технических средств, которая используется для сбора, обработки и хранения информации, оно дает возможность оптимизировать и сделать более эффективным те решения, которые будут приняты. Автоматизированные системы делают так, чтобы участие человека в некоторых процессах сводилась к минимуму.

Многие компании для того чтобы улучшить эффективность своей деятельности используют оптимизацию. Она помогает выбрать такой вариант управления, который будет самым эффективным. Информационная система принимает участие в решении тех задач, которые связаны с обработкой информации, или же с управлением тех процессов, которые есть в компании. В качестве основных проблем, которые помогают решить информационные системы можно отметить:

- принять такие решения, которые будут максимально эффективными.
- быстро собирать информацию, притом сбор информации может происходить в реальном времени.
- многовариантность расчетов;
- проводить более качественный анализ;
- реализация комплексного учета.

Те возможности, которые есть у современных технологий разнообразны. Они могут решать даже самые большие сложные задачи. Сегодня информационную систему можно рассматривать как некий объект, состоящий из разных элементов, необходимые для того, чтобы решить те задачи, которые перед ними ставятся.

Многие компании используют информационные автоматизированные системы для того, чтобы сделать процессы в компании не только более эффективными, но и уменьшить количество сотрудников в компании. Связано это с тем, что информационная система с лёгкостью справляется с такими задачами как сбор информации, обработка информации и хранения информации. Именно эти задачи является очень важными для многих компаний, поскольку они оказывают влияние на те решения, которые будут приняты. Если подходить к рассмотрению понятия информационная система в целом, то можно отметить, что это некий комплекс, который в своей деятельности используют разные средства, разные методы для обработки, сбора и хранения информации.

Сейчас говоря о информационной системе часто подразумевается, что её использование может быть только в том случае, если будет специальная техника, сейчас в качестве такой техники выступают ноутбуки и компьютеры. То есть тут можно говорить о том, что для реализации информационной системы нужно не только технология, но и человек, поскольку такая система не может работать полностью без участия человека. Этот фактор играет большую роль, поэтому его необходимо учитывать вовремя разработки информационной системы. Информационная система также зачастую используется для того, чтобы объединять разные потоки информации, которые могут поступать от разных объектов.

В состав современной информационной системы входят:

- аппаратная часть;
- источник информации;
- программная часть.

В качестве основных понятий экономической информационной системы можно отметить:

- интегрируемость;
- масштабируемость;
- управляемость;
- адаптивность;
- используемость (она необходима для того, чтобы реализовать возможности, которые заложены в саму систему);
- обоснованность;
- реактивность (очень важное качество, которое показывает насколько информационная система может правильно и быстро реагировать на изменения и на различные воздействия, которые на нее оказываются);
- безопасность, (она отвечает за то, чтобы все данные, находящиеся в информационной системе находились в безопасности, то есть были защищены от несанкционированного доступа).

На сегодняшний день можно выделить очень много разных экономических информационных систем. Некоторые из этих систем могут выполнять не одну, а две и более разных целей, при том делать это одновременно. Для того чтобы определить качество используемого экономической системы, необходимо объединить несколько свойств, которые будут показывать насколько система может удовлетворить потребности своего пользователя, насколько она соответствует всем действующим стандартам.

С экономической точки зрения необходимо оценивать стоимость разработки или создания такой системы, и определить насколько это выгодно. Во время определения стоимости необходимо учитывать:

- затраты, которые связаны с внедрением системы;
- затраты, которые компания потратит на тестирование;
- затраты на настройку системы;
- кроме того, большую роль играет и эксплуатация, а также те затраты, которые компания должна на нее потратить.

Тот результат, который компания получит в результате внедрения такой системы свою деятельность, также играет большое значение. В качестве основных показателей эксплуатационных возможностей системы можно отметить:

- насколько комфортно пользователю работает в данной системе;
- насколько системы надёжна и стабильна;
- возможность стабильной работы в сети;
- надлежащая степень автоматизации;
- наличие интуитивно понятного пользовательского интерфейса;
- наличие возможности надёжной и бесперебойной информационной защиты информационных данных и самой системы.

Если рассматривать все информационные системы, то их можно разделить на разные группы. Если в основу деления класть степень их

автоматизации, то можно выделить следующую группы информационных систем:

- ручные;
- автоматизированные;
- автоматические.

В качестве примера информационной системы автоматической, которую знают практически все можно провести поисковик Яндекс и Гугл. Эти системы разработана таким образом, чтобы собирать разные данные на различных ресурсах, при том они спроектированы так, чтобы человеческий фактор не мог оказать влияние на итоги поиска.

Если делить все информационные системы, существующие на сегодняшний день, с точки зрения особенностей их использования, то можно определить следующую группу систем:

- информационно-решающие;
- информационно-поисковые;
- информационно-аналитические;
- ситуационные центры.

При этом, ситуационные центры являются, по сути своей, ничем иным, как обобщенными информационно-аналитическими современными и весьма эффективными комплексами.

Все многообразие информационных систем, которые есть на сегодняшний день, можно разделить на группы с точки зрения той сферы, для которой не создаются. В этом случае можно выделить:

- информационные системы организованного управления;
- информационные системы управления техпроцессами;
- автоматические информационные системы научных исследований;
- информационные системы автоматизированного проектирования;
- автоматизированные системы обучения;
- интегрированные информационные системы.

Также во время определения функциональности информационной системы необходимо определить для чего оно создается, какие технические возможности у нее должны быть.

В рамках выполнения дипломного проекта поставлена задача разработать систему оперативного обмена служебными данными в виде текстовых файлов, файлов изображений, файлов других форматов и сообщений внутри определенной группы пользователей – корпоративного мессенджера. Корпоративный мессенджер разрабатывается для рабочих групп, одновременно выполняющих несколько проектов, в каждом из которых предполагается назначение ролей между одними и теми же исполнителями (пользователями). Организация работы с мессенджером предполагает, что в рабочей группе имеются специалисты, выполняющие конкретные задачи (соответствуют роли управляющий, аудитор, писатель и т.п.) в конкретном проекте. При этом в другом проекте этот же сотрудник может выполнять задачу, соответствующую другой роли. Данный продукт ориентирован на организацию удаленной работы закрытых групп пользователей.

1.1.2 Изучение предметной области

Разработка внедряется в сеть интернет, размещается на веб-сервере. Для описания предметной области (организации удаленной коммуникации в закрытых группах) можно рассмотреть компанию *Alconost*, в которой спустя 10 лет трудится более 300 человек со всего мира. Компания создала постоянно обновляющийся информационный ресурс, где собраны и систематизированы знания компании, правила, нужные ссылки и предыдущие наработки, что существенно ускорило обучение новых сотрудников и оптимизировало работу в целом. Все документы и файлы компании хранились в *Google Drive*, это позволяет получить доступ к документам с любого устройства, удобно согласовать тексты и осуществлять совместные правки. Для компании важна максимальная автоматизация рутины, упрощение бизнес-процессов и возможность работать в любое время и в любом месте.

Таким образом, для того, чтобы сделать работу пользователей более удобной, необходимо использовать такое приложение, которое бы позволяло общаться и обмениваться данными между группами пользователей. У каждой группы пользователей должны быть свои права и назначенные компетенции.

1.1.3 Обзор аналогичных решений

- *Staply* - мессенджер с проектным функционалом. Позволяет: составить иерархию групп; создавать задачи, задавать сроки, важность и ответственность в рабочих группах; осуществлять контроль над группами;

- *Slack* - корпоративный мессенджер с поддержкой интеграции с десятками сторонних сервисов. Объединяет в одном окне обсуждения в общих темах (каналах), приватных группах и личных сообщениях. Позволяет осуществлять видеозвонки; хранить сообщения без доступа к сети;

- *HipChat* - представляет собой групповой чат, видеочат для команд и компаний с возможностями обмена файлами и демонстрацией экрана;

- Битрикс24 - система управления внутренним информационным ресурсом компании для коллективной работы над задачами, проектами и документами, для эффективных внутренних коммуникаций. Позволяет контролировать, делегировать, оценивать задачи; создавать резервные копии в нескольких местах; работать в нескольких комнатах одновременно;

- *Skype* - программное обеспечение с закрытым кодом, обеспечивающее текстовую, голосовую и видеосвязь через Интернет между компьютерами;

- *Facebook* - социальная сеть для общения с друзьями, коллегами, одноклассниками и с теми, кто живет, работает и учится неподалеку;

- Вконтакте - универсальное средство для общения, которым ежедневно пользуются десятки миллионов человек;

- *Telegram* - кроссплатформенный мессенджер для смартфонов и других устройств, позволяющий обмениваться текстовыми сообщениями и медиафайлами различных форматов. Поддерживает ботов. Предоставляет: доступ к смайлам и стикерам, возможность настройки оформления;

- *ICQ* - централизованная служба мгновенного обмена сообщениями сети Интернет.

На основе функций аналогов и поставленных задач можно выделить следующие критерии:

- интеграция со сторонними сервисами;
- создание иерархии групп;
- распределение управляющим участников по ролям;
- возможность контроля управляющим над группами;
- резервное копирование в нескольких местах;
- работа в нескольких комнатах одновременно;
- настраиваемое оформление;
- инструменты;
- смайлы и стикеры;
- офлайн-сообщения.

Сравнение решений приведено в таблице 1.

Таблица 1 Сравнение существующих решений

Аналог Критерий	<i>Staply</i>	<i>Slack</i>	<i>HipChat</i>	<i>Битрикс24</i>	<i>Skype</i>	<i>Facebook</i>	Вконтакте	<i>Telegram</i>	<i>ICQ</i>	Примечание
Интеграция со сторонними сервисами	-	<i>Dropbox</i> , <i>Google Drive</i> , <i>GitHub</i> , <i>Google Docs</i> , <i>Google Hangouts</i> , <i>Twitter</i> и другие.	<i>GitHub</i> , <i>MailChimp</i> , <i>Heroku</i> и другие.	<i>MS SharePoint</i> , <i>MS Exchange Server</i> , <i>MS Outlook</i> , Продукты <i>Apple</i> и <i>Google</i> .	-	-	-	-	-	Позволяет пользователям отслеживать прогресс в различных проектах с помощью одной платформы
Создание иерархии групп	+	-	-	-	-	-	-	-	-	Позволяет равномерно распределить задачи
Распределение управляющим участников по ролям	-	-	-	-	-	-	-	-	-	Позволяет эффективно распределить задачи
Возможность контроля управляющим над проектом	+	+	+	+	+/-	+	+	-	-	Обеспечивает контролирование выполнения задач

Аналог Критерий	Staply	Slack	HipChat	Битрикс24	Skype	Facebook	Вконтакте	Telegram	ICQ	Примечание
Резервное копирование в нескольких местах	+/-	-	-	+	-	-	-	-	-	Обеспечивает сохранность данных
Работа в нескольких комнатах одновременно	+	-	-	+	+/-	+	+	+	+	Позволяет решать несколько задач одновременно
Настраиваемое оформление	-	-	-	-	-	-	-	+	+	Позволяет индивидуализировать интерфейс под пользователя или организацию
Инструменты	Блокнот, Задачи	-	-	Ежедневник, планировщик событий, календарь, график отсутствия, собрания и планерки	-	Создание мероприятий, Календарь	Создание мероприятий	-	-	Позволяют оперативно выполнять задачи

1.1.4 Функциональные характеристики

Разрабатываемое приложение будет позволять осуществлять обмен данными между зарегистрированными пользователями, обмениваться заданиями и файлами, объединять пользователей в рабочие группы и администрировать работу системы.

1.1.5 Функциональные требования

Система должна восстанавливать свое функционирование при корректном перезапуске аппаратных средств. В случае аварии необходимо обеспечить сохранность всех данных системы. Для этого должна быть предусмотрена возможность организации автоматического и (или) ручного резервного копирования данных Системы средствами системного и базового программного обеспечения (ОС, СУБД), входящего в состав программно-технического комплекса Заказчика.

Техническая помощь - это сложный метод обработки технической информации, предназначенной для работы информационной системы, а также соответствующей документации по этим средствам и техническим процессам.

Аппаратное обеспечение, коммуникация, программное обеспечение - это аппаратное развитие, неравномерное, пошаговое. Развитие компьютерных

технологий ускоряется. Производительность компьютера удваивается каждые четыре года.

Все компьютерные технологии можно разделить на разные группы. Если в основе деления лежит тип пользовательского интерфейса, то можно выделить: договорные компьютерные технологии, сетевые компьютерные технологии и пакетные компьютерные технологии.

Технологии, которые используются для управления информацией на сегодняшний день, очень разнообразны. Они отличаются по тем функциям, которые они могут выполнять, по составу. Вычислительная технология выступает в качестве средства организационных и коммуникационных технологий.

Коммуникационные технологии используются для того, чтобы передавать информацию. Они состоят из функциональной работы и самопроизвольной работы с компьютером и оборудованием.

Компьютер выступает в качестве инструмента, который используется для выполнения сложных технических задач, он помогает хранить данные, также он используется для интеграции различных технологий, которые направлены на работу с информационными ресурсами.

Организационная технология используется тогда, когда необходимо выполнить разные мероприятия, которые связаны с технологией хранения, с использованием и передачей информации. Также данная технология используется в том случае, если необходима поддержка информации [4].

В нашем случае процесс автоматизации используется для того, чтобы в автоматическом режиме совершать основные процессы, которые проходят в регистрации. Процесс автоматизации позволит распределять услуги между кабинетами:

- персональные компьютеры, которыми не пользуются менеджеры компании;
- соединительные линии, которые присутствуют в локальной

вычислительной сети;

- сервера.

Для того чтобы оценить персональный компьютер, необходимо его оценить по таким критериям:

- какой объем имеет жесткий диск;
- качество работы и быстродействия процессора;
- какой объём оперативной памяти установлен;
- с какой скоростью работает видеокарта.

Во время подготовки проекта и на начальном этапе большую роль играет определение того варианта, каким образом будет проходить автоматизация. Если выбор будет неправильный, то система не будет работать корректно, от нее не будет эффекта. В самом плохом случае неправильный выбор системы может привести к тому, что производство полностью остановится.

Существуют принципы, на которых должна основываться система автоматизации. К ним относятся:

- цель системы заключается в том, чтобы определить сферы, в которых будут внедряться процессы автоматизации, выделить последовательность и структуру, по которым данный процесс будет происходить;

- методы реализации системы автоматизации. В данном случае речь идет о том, в каком процессе или по какому направлению будет использоваться система автоматизация;

- техническая политика описывает количество тех внутренних стандартов и тех методик, которые могут быть использованы во время становления системы автоматизации;

- какие существуют ограничения. В данном случае речь идет о денежных или временных ограничениях;

- внедрение процесса контроля изменениями в планах.

Автоматизация проводится для того, чтобы решить некоторые задачи, стоящие перед бизнесом. Сам процесс заключается так же в разработке тех

путей, которые позволят достичь необходимого результата. Во время подготовки плана необходимо учесть ряд факторов, в качестве основных факторов можно отметить:

- насколько изменится в среднем производственный цикл;
- какой средний жизненный цикл продукции, которая производит компанию;
- для чего проводится процесс автоматизации, какие дальнейшие планы компании, связанные с развитием;
- соответствуют ли сотрудники занимаемой должности, необходима ли корректировка их функциональной части.

Часто автоматизацию рассматривают в качестве метода, который дает возможность достичь бизнес-целей компании. Но иногда данного процесса недостаточно. Процесс автоматизации в разных компаниях оказывает разное влияние на деятельность. Это связано с целями компаний, с их направлением работы, их миссии и так далее.

Так же существуют и ограничения, которые необходимо принимать во внимание во время определения стратегии. В качестве основных ограничений выступают:

- ограничения аппаратные;
- временные ограничения. Установка системы занимает определенное время, которое необходимо учесть при разработке системы. Так же для того чтобы сотрудники компании могли работать с новой системой, необходимо провести обучение;
- финансовые ограничения. Любой проект имеет свою стоимость, то есть то, сколько компания готова потратить для его реализации;
- человеческий фактор. В данном случае речь идет о том, как работники относятся к новому процессу, какие нюансы существуют с точки зрения законодательства.

На сегодняшний день выделяют несколько стратегий, которые могут

использоваться в процессе оптимизации. В первом случае то программное обеспечение, которое имеется в компании, автоматизируется под определенный бизнес-процесс. Во втором случае речь идет о том, что изменяется сам процесс и автоматизирует для того, чтобы получить более простую структуру. На выбор стратегии, которая будет использоваться в каждом конкретном случае, играет влияние разные факторы, в том числе экономические факторы или перспективы работы компании.

Далее предлагаю рассмотреть разные варианты автоматизации, которые можно использовать в компании:

- кусочная автоматизация. Данный вид автоматизация многие считают неэффективным, с точки зрения инвестирования денежных средств в развитие компании. Данный вид автоматизации отличается хаотичностью процессов, то есть при нем стратегический план не разрабатывается. Все решения, которые принимаются в данном случае, принимаются на основе различных предпочтений, в зависимости от уровня знаний, в зависимости от доступности оборудования, от скидок, которые предоставляются на технику и так далее;

- автоматизация по направлениям. В данном случае автоматизация затрагивает отдельные направления в деятельности компании. Такую методику зачастую используют в системах класса *ERP, MRPII*;

- автоматизация по участкам. В данном случае речь идет о том, что процесс автоматизации происходит не на всем производстве, а на его отдельных компонентах;

- полноценная автоматизация управления компанией. В данном случае автоматизация затрагивает все процессы и все подсистемы компании. Если рассматривать с точки зрения технического обеспечения, то процесс автоматизации затрагивает связь компании с внешней средой. Если рассматривать с точки зрения информационного обеспечения, то используется базы данных, которыми управляет СУБД.

Рассмотрев имеющиеся варианты и имеющиеся стратегии, можно

					ПГУ 1.090501.02.001 ПЗ	Лист
						25
Изм.	Лист	№ докум.	Подпись	Дата		

определить, что в нашем случае самым эффективным будет автоматизация по направлениям. Это связано с тем, что автоматизировать необходимо только одно направление в деятельности компании.

Сам процесс автоматизации будет выглядеть следующим образом:

- вначале необходимо изучить принципы, по которым работает компания;
- изучить варианты, по которым компания может развиваться в дальнейшем;
- определить какие стратегические свойства имеются в системе учета;
- определить какие функциональные возможности есть у имеющихся систем учета;
- изучить какие варианты систем могут использоваться;
- определить каким образом будет происходить процесс автоматизации;
- определите структуру данного процесса;
- подготовить бизнес-план в соответствии с которым будет внедрен процесс автоматизации в компании.

Информационное обеспечение - это совокупность информации, информационных массивов и систем документации.

Информационное обеспечение состоит из двух компонентов. Первый компонент - это классификаторы технико-экономической информации и документы. Второй компонент - это экранные формы и макеты для ввода первичных данных. Для обеспечения эффективного поиска информации, необходимо сделать так, чтобы информация вся была представлена в цифровом виде. Для этого необходимо вначале классифицировать информацию, а далее закодировать её. При этом используются различные классификаторы.

Классификатор - это некий документ, который используется для того чтобы описать экономическую информацию в ЭИС. Данный документ состоит из названий объектов, кодовых обозначений, классификационных

группировок.

Классификаторов на сегодняшний день можно выделить очень много. Если рассматривать классификаторы с точки зрения их сферы деятельности, то можно выделить следующие группы:

- отраслевые;
- международные;
- локальные;
- общегосударственные.

В качестве основного компонента информационного обеспечения выступают системы документации. Они используются во время управления объектом. В данном случае в качестве документов выступают разные сведения, которые помогают решать экономические задачи. Данные сведения находятся на носителе.

Внутримашинный ввод-вывод - это представление данных на машиночитаемом носителе в виде специально организованных массивов (файлов), базы данных и их связей данных. Эта информация, которая встроена поддерживает под систему и создают некую среду в которой работают сотрудники. Данная среда используется сотрудниками для выполнения своих обязанностей. Каждый объект имеет определённый набор атрибутов и свойств [5].

На сегодняшний день есть определенные требования, которые предъявляются к поддержке данных. В качестве основных требований можно отметить:

- та информация, которая получена, может быть использована для кодирования информации в классе;
- поддержка должна быть такой, чтобы она могла поддерживать все функции объекта;
- для того чтобы кодировать входную и выходную информацию используется самый высокий уровень управления. Использовать данный

уровень управления необходимо в классе;

- документы должны соответствовать всем стандартам, которые приняты в компании;
- системы, взаимодействующие с усовершенствованными системами, должны обеспечивать совместимость с информационной поддержкой;
- данные, которые расположены в документах, то есть структура документа, графики и содержания необходимо согласовывать с заказчиком.

1.2 Проектирование мессенджера

1.2.1 Обоснование выбора технологий и средств разработки

Современный инструментарий инновационных реляционных систем по непосредственному управлению базами данных, буквально с каждым днём, становится все более эффективным и совершенным. В нем регулярно появляются и новые функциональные возможности, и сервисы. Реляционная модель данных, по состоянию на сегодняшний день, при выполнении построения баз данных выполняет роль базового инструмента. Её эффективность объясняется, в первую очередь, наглядностью хранения данных. Данная система представлена в виде логичных и наглядно понятных для конечного пользователя таблиц. Все это стало непосредственной причиной появления большого количества реляционных СУБД, которые между собой отличаются по многим, совершенно разным, параметрам и выбор нужного СУБД сейчас представляет, поэтому определённую сложность и не так-то прост, как это может показаться на первый взгляд.

Стоит подробнее остановиться на наиболее популярных и повсеместно распространённых СУБД.

Oracle Database. Данная СУБД является объектно-реляционной и именно она стала коммерческой системой, которая поддерживает популярный и весьма распространённый SQL-язык. Всего сейчас в прямой доступности находится шесть СУБД:

					ПГУ 1.090501.02.001 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		28

- *Enterprise Edition*. По сути своей является максимальной полноценной версией, не имеющей каких-либо ограничений. В первую очередь данная версия предназначена для корпоративного использования на крупных предприятиях. Конечные пользователи получают довольно эффективный инструмент, используя который можно быстро и с наименьшими трудовыми затратами оптимизировать сервер как функционально, так и архитектурно;

- *Standard Edition*. Стандартная версия имеет определённые ограничения по количеству имеющихся процессорных разъемов. Как правило, их количество не превышает четырёх штук. Используется данная редакция в компаниях и предприятиях среднего размера. Данная редакция, в основном, используется во встраиваемых и мобильных устройствах. Также данная версия успешно применяется на малых предприятиях;

- *Express Edition (XE)*. Является вполне доступной СУБД, в которой используется только один процессор, а также имеются в наличии существенные ограничения по объёмам - ОЗУ (1 Гб), БД (11 Гб).

Стоит заметить, что каждая из вышеперечисленных версий СУБД имеет сходный исходный код, а их функциональность идентична.

Oracle Database весьма популярна в наши дни, она является кроссплатформенным программным обеспечением и большая её часть написана на языке программирования C. Что касается оставшейся части, то она отведена под переделку. Всего переделке подлежит до двадцати процентов кода и его можно переделать под нужную платформу. Стоит понимать, что все создано на языках, которые являются машинно-зависимыми.

Microsoft Access. Данная реляционная СУБД имеет широкое распространение и создана она корпорацией Майкрософт. Благодаря встроенному в неё языку *Basic for Applications (VBA)*, пользователь имеет возможность для непосредственной работы с базами данных, а также для создания приложения в *Access*. С этой целью используется структурированный язык *SQL*, а также макрокоманды.

MS Access весьма удобна в использовании, и она считается особым файло-серверным СУБД. В качестве движка используется *Access Database Engine* либо же *Microsoft Jet 4.0* (вид используемого движка, в первую очередь, зависит от ревизии СУБД).

MS Access также можно вполне успешно совмещать с внешними СУБД. В частности, ее можно успешно совмещать с *MySQL*, *Firebird*, а также *Oracle*.

Несмотря на то, что существуют версии с открытыми программными кодами, для большей эффективности лучше всего приобрести официальную лицензию.

При недостатке знаний персонал способствует дополнительному обучению сотрудников компании, которые по долгу своих рабочих обязанностей должны будут тесно взаимодействовать с новым программным комплексом. Как правило, подобное обучение осуществляется в максимально сжатые сроки и весьма эффективно. После обучения, сотрудники компании либо производственного предприятия смогут максимально тесно взаимодействовать с новым программным комплексом, работая с базами данных. Как результат, эффективность и продуктивность их труда значительно повышается, а работа с базами данных автоматизируется и сроки необходимых на ее успешное осуществление значительно сокращаются. Стоит отметить, что настольная версия СУБД имеет удобный русифицированный интерфейс, а также весьма хорошую СЗИ.

MS SQL Server. Данная СУБД также была популярной и её актуальность в наши дни нисколько не уменьшилась. Данный программный продукт является результатом совместной работы *Sybase*, *Ashton-Tate*, а также *Microsoft*. Стоит обратить внимание на тот факт, что *MS SQL Server* является СУБД клиент-серверной архитектуры. Обновлений данного программного обеспечения, по состоянию на сегодняшний день, выпущено огромное количество и все они включают в себя следующие компоненты:

- основной компонент ядра СУБД (*Database Engine*);

- полный набор технологий для репликации, непосредственно с базой данных;

- наглядный вывод, в удобно воспринимаемой форме, отчетов;

- всесторонний анализ имеющихся данных.

Версия ПО *Microsoft SQL Server Express Edition* имеет ограниченную функциональность и находится она в открытом доступе. Стоит заметить, что данная версия используется, в основном, только в небольших компаниях либо на компактных производствах. При этом *Microsoft SQL Server* полностью совместим с базами данных других форматов. В частности, программное обеспечение *Microsoft SQL Server* совместимо с:

- *Oracle*;

- *DB2*;

- *Sybase*;

- *Microsoft Access*.

Отличается данная версия СУБД простым и понятным доступом к анализируемым данным. Данные при необходимости можно надёжно шифровать, а также ограничивать к ним доступ со стороны несанкционированных лиц, не имеющих разрешения на осуществление такого доступа.

Sybase Adaptive Server Enterprise. Данная версия реляционной СУБД, разработана толковыми сотрудниками фирмы SAP.

Первые версии были разработаны в тесном сотрудничестве с сотрудниками и специалистами *Microsoft SQL Server*. Ее успешно эксплуатируют при необходимости осуществлять постоянное тесное взаимодействие с большим количеством пользователей, а также подвергать анализу и обработке большие информационные объёмы.

От программного обеспечения посторонних производителей данная СУБД не зависит, что также добавляет ей привлекательности в глазах специалистов, использующих её в своей работе.

Также весьма популярной является *SAP Sybase Adaptive Server Enterprise Cluster Edition*, которая характеризуется уникальной отказоустойчивостью.

Она обеспечивается путём разбивания на отдельные кластеры, а также перевода работника и пользователей с кластера в котором произошёл сбой на рабочий узел.

В случае произошедшего сбоя операция выполняется вторично. При этом происходит переход на рабочий узел.

ЛИНТЕР. Данная СУБД была разработана нашими соотечественниками, сотрудниками компании РЕЛЭК.

Она относится к типу кроссплатформенного программного обеспечения и поддерживает данный программный комплекс практически все, существующие на день сегодняшний, операционные системы.

В данной СУБД отечественного производства язык запросов отвечает полностью всем требованиям *SQL:2003*.

По состоянию на день сегодняшний существуют и успешно применяются следующие версии данной СУБД:

- ЛИНТЕР *Multiversion*;
- ЛИНТЕР Бастион;
- ЛИНТЕР *Realtime*;
- ЛИНТЕР Стандарт.

MySQL является весьма эффективной и распространенной, универсальной СУБД.

Она хоть и имеет лицензированную версию, но можно скачать и полностью бесплатную версию данного ПО. Платная версия поставляется от *MySQL AB*. СУБД имеет открытый код.

Самая первая версия СУБД была издана в далеком 1995 году. На данный момент разработка и всесторонняя поддержка данного программного обеспечения осуществляется компанией *Oracle*. ПО используется непосредственно для осуществления управления малых, а также средних

систем. Стоит заметить, что в версиях выше *MySQL 3.22* все без исключения ограничения с таблиц были сняты.

Также имеется весьма удобный и практичный визуальный интерфейс. Он позволяет значительно упростить процесс работы с базами данных данного программного обеспечения, а именно с БД *MySQL* — *PhpMyAdmin*. СУБД *MySQL* соответствует стандарту языка *SQL*, начиная с версии 5.0. Основным языком разработки является распространенный *C/C++*.

IBM Database 2 от компании *IBM* представляет собой современную и весьма эффективную систему управления реляционными базами данных. Имеет ограничения на количество используемых процессоров. Начиная с 2000-х годов имеется в доступе бесплатная версия ПО. В данной версии имеются ограничения на количество используемой памяти и процессорных ядер. Также не имеет своей системы по осуществлению аутентификации пользователей.

Для этой цели используются средства самой операционной системы либо возможности серверов безопасности. Авторизация пользователей осуществляется при помощи *DB2*, непосредственно после аутентификации. При этом Прекомпилятор является главным элементом *DB2*. Кроме этого, к списку основных элементов *DB2* относятся:

- супервизор стадии исполнения (*Runtime supervisor*);
- генератор основных планов прикладных задач (*Bind*);
- программа текущего управления всеми хранимыми информационными данными (*Stored data manager*).

Firebird это ни что иное, как сервер баз данных и используется он непосредственно для обработки весьма внушительных баз данных. Основано данное ПО на *InterBase 6.0* от компании *Borland*.

При этом ПО *PostgreSQL* основан на распространенном языке программирования *SQL* и соответствует стандарту *SQL:2011*.

PostgreSQL распространяется свободно на бесплатной основе. Также

имеются такие два платных расширения данного ПО:

- *Postgres Plus Advanced Server*;
- *Postgres Plus*.

Они обеспечивают полную совместимость с *Oracle Database*. Стоит заметить, что в *PostgreSQL* используется непосредственно клиент-серверная модель данных. Также поддерживается работа с массивными объёмами баз данных. При этом можно не только использовать разные типы данных, но и создавать свои собственные.

PostgreSQL отличается высокой гибкостью. Можно выполнять хранимые процедуры на следующих языках программирования:

- *Java*;
- *Perl*;
- *Python*;
- *Ruby*;
- *C/C++*.

Имеются как стандартные библиотеки, так и различные встроенные функции. Имеется лицензия *BSD (Berkeley Software Distribution license)*, которая предоставляет возможность модифицировать СУБД и свободно его распространять. Таким образом, пользователи могут использовать *PostgreSQL* непосредственно как платформу для разработки своих собственных приложений, которые используют реляционную СУБД.

Interbase относится к кроссплатформенным системам управления основными реляционными базами данных. По состоянию на сегодняшний день её продвижением занимается компания *Embarcadero Technologies*. Стоит отметить, что ряд разработок по данной СУБД легли в основу общедоступной СУБД, известной под названием *Firebird*.

Использоваться данная программная система может как на домашнем персональном компьютере, так и на крупных серверах производственных компаний. Весьма устойчиво данное ПО к перебоям в электроснабжении.

Также примечательно, что *Interbase* полностью поддерживает максимальное большинство самых популярных языков программирования. В частности, поддерживаются следующие языки программирования:

- *Java*;
- *C*;
- *C++*;
- *NET*;
- *Delphi*;
- *PHP*;
- *Ruby*.

При этом, версия ПО 2009 года использует метод симметричного блочного шифрования *AES/DES*, а также поддерживает юникод. Также имеется удобная возможность арендовать внешние вычислительные мощности.

Программное обеспечение можно успешно использовать в течении ограниченного временного промежутка. После этого требуется приобрести лицензию данного ПО.

SQLite довольно популярная СУБД, которая относится к кроссплатформенному программному обеспечению. Языком исходного кода является СИ. Примечательно, что исходный код данной версии находится в открытом доступе. Главная особенность рассматриваемой СУБД - эффективная библиотека, работающая в одном с приложением процессе. Упомянутая выше модель данных в значительной степени упрощает программу, а также уменьшает время отклика. Даная консольная утилита даёт возможность осуществлять работу с файлом базы данных, непосредственно из командной строки. Данные могут вноситься в любой столбик таблицы независимо от его типа. Таким образом, программное обеспечение *SQLite* позволяет значительно повысить точность и эффективность выполнения своей работы специалистами и сократить временные сроки на их осуществление.

В этом разделе приведем сравнительный анализ, а также укажем характеристики основных языков программирования и эффективных технологий создания функциональных веб-проектов:

- *PHP* является довольно распространенным гипертекстовым препроцессором. Он представляет собой современный язык программирования, используемый для осуществления генерации страниц ресурсов сети, которые непосредственно ориентируются на гипертекстовую разметку. Данный язык характеризуется весьма высокой скоростью обработки, и он дает уникальную возможность подключать сторонние модули. Также данный язык дает возможность осуществлять продуктивную работу с криптографической защитой, базами данных, графикой и документацией. Стоит заметить, что расширения и подключаемые модули для языка *PHP* существуют для всех типов операционных систем;

- *ASP* является современной многофункциональной технологией, которая представляет собой отдельные страницы на сервере. При этом, данные страницы при их вызове способны выполнять ряд определенных действий. Таким образом, при помощи данной технологии можно создавать страницы-приложения, интегрированные с различными информационными системами и базами данных;

- *ASP.NET* является своеобразным расширением технологии *ASP*, продолжающим ее функциональность. Поддерживает гораздо более расширенную функциональность по сравнению с *Microsoft ASP*. Стоит заметить, что данная программная среда стала одной из частей популярного проекта по развитию Интернета – платформы *.NET Framework*. Она является программным компонентом, выпущенном компанией *Microsoft*;

- *Perl* ещё не так давно позиционировался для эффективного создания отчетов на базе собранных из хранилища данных. В связи с этим *Perl* использовался для работы с большими базами данных. Также он успешно применялся для создания сайтов с интегрированными в них базами данных.

Также Перл поддерживает много инструментов, необходимых для работы непосредственно с текстом;

- *Python / Zope* являются довольно распространенными в наши дни языками высокого уровня. При этом, программные приложения под непосредственным управлением *Python* характеризуются высокой скоростью исполнения. *Python* повсеместно профильные специалисты используют для разработки приложений для математических расчетов либо же осуществления работы с текстом;

- *JSP* это технология, которая представляет собой страницы, непосредственно написанные на распространенном языке *Java* и выполняемые на стороне сервера. При использовании *Jasper* в самом начале создается высоко оптимизированный байтовый код. После этого он передается *Java*-машине для его непосредственного выполнения;

- *HTML*, по сути, является основой современного интернета в том виде, в каком мы привыкли его видеть и использовать. Разработчик с его помощью может применять в своей работе самые разные фреймы. Также он позволяет предоставлять в браузере любые данные в виде графических изображения. При этом, готовая страница представляет собой набор папок, в которых расположен текст, а также осуществляется подключение прочих сторонних компонентов;

- *XHTML* является, по сути, гипертекстовой разметкой. Однако он основан не на тегах, а на особых иерархиях *XML*. Специфика *XHTML* позволяет его использовать непосредственно на различных мобильных устройствах;

- *XSL* это набор специальных пользовательских стилей. Однако он, по сути своей, имеет другую реализацию по сравнению с распространённой *CSS*. В частности, в нем непосредственно используется *XML* и именно это и является основным отличием *XSL*. Стоит заметить, что использование *XML* кардинально уменьшает общее количество допущений при разработке

профильными специалистами непосредственно самого стиля. При этом, на этапе написания документ можно всегда, при необходимости, проверить на отсутствие ошибок и различных неточностей. Таким образом, можно максимально оперативно обнаружить все ошибки еще на стадии проектирования;

- *JavaScript* также широко используется профильными специалистами в наши дни при создании разных сайтов и сетевых порталов. Он относится непосредственно к прототипно-ориентированному языку программирования. Также, по сути, он является неким диалектом языка *ECMAScript*. В подавляющем большинстве случаев он используется в роли встраиваемого языка, предоставляющего большой функционал и ориентированного для осуществления доступа непосредственно к различным объектам программных приложений. Также он повсеместно используется в распространенных браузерах как сценарный язык, позволяющий успешно реализовывать сценарии для непосредственной реализации интерактивных веб-страниц. Стоит отметить, что указанная технология используется в последнее время для осуществления корректировки специалистами работы органов управления ресурса и осуществления контроля за действиями браузера. *JavaScript* используют также в сфере рекламы, для создания эффективных баннеров, а также в современной контекстной рекламе, которую не обнаруживают поисковики;

- *SSI* непосредственно с английского языка переводится как включение от сервера. При помощи него специалисты осуществляют перенос отдельных функций либо повторяющихся кодов, исполняемых для непосредственной подзагрузки в выбранный документ по мере необходимости. Позволяет успешно заменять каскадные стили и значительно облегчает работу профильных специалистов по разработке сайтов;

- *AJAX* является весьма распространённой в наши дни среди профильных специалистов технологией, представляющей собой асинхронный

JavaScript, а также *XML*. Они при работе пользователей на страницах ресурса в интернете позволяют выполнять оперативный обмен информацией без обновления непосредственно самой страницы. Эта возможность весьма удобная и повсеместно ее используют все ведущие профильные специалисты. Благодаря данной технологии пользователь, может оставаясь на одной странице осуществлять просмотр фото, читать полезные и интересные для него статьи, а также просматривать сообщения. Таким образом, навигация по сайту значительно облегчается и упрощается. Однако, справедливости ради стоит отметить также следующие минусы данной технологии:

- 1) работа данной технологии становится видимой при падении скорости канала;
- 2) оказывается негативное влияние на рейтинг сайта;
- 3) технология *AJAX* потребляет определенное количество ресурсов пользовательского персонального компьютера. Из-за этого, при просматривании портала с насыщенной графикой и мультимедиа он может подтормаживать и даже подвисать при недостаточной мощности ресурсов персонального компьютера пользователя;

- *XML* является весьма популярной иерархической технологией, предназначенной для хранения данных непосредственно в текстовом виде. Стоит заметить, что непосредственно написание скриптов на *XML* вполне можно отнести к объектно-ориентированному программированию.

В заключении стоит отметить, что, рассмотрев все положительные стороны языков программирования, как и имеющиеся минусы было принято окончательное решение.

Непосредственно для разметки страниц будущего сайта будет использоваться язык *HTML*, *CSS* будет использоваться для оформления стиля страницы, *JavaScript*, а также базирующаяся непосредственно на нем технология *AJAX* будет использована для быстрой работы ресурса и проверки переданных данных на их корректность и отсутствие ошибок. Также будет

использоваться при выполнении работ по разработке сайта язык *PHP* как ключевой технологии всего проекта, так как именно он предназначен для создания подобных разработок. Язык *PHP* является простым и его преимуществом можно назвать также возможность просматривать результаты работы на своем персональном компьютере в автономном режиме без подключения к удаленному серверу.

1.2.2 Описание разработанного программного средства

Внутримашинный анализ - это совокупность операций, которые выполняются в диалоговом режиме при помощи персонального компьютера. То есть можно говорить о том, что технологические процессы являются определенными технологическими операциями, которые выражены в взаимодействии пользователя с персональным компьютером. Сама работа происходит в определенной последовательности, которая заранее известна.

Пользователь, который осуществляет данные операции должен иметь определенный интерфейс и определенный доступ. Пользователь может выполнять только те действия, права на которые ему предоставлены. После того как пользователь проходит авторизацию, перед ним открываются перечень тех возможностей и тех прав, которые у него есть. Из предложенного списка может выбрать те данные, которые ему необходимы для дальнейшей работы. После чего в выбранном справочнике выбираются таблицы, которые будут использоваться в дальнейшей обработке.

Процесс программирования — это такой процесс, который заключается в разработке программы. Независимо от метода исследования сам процесс будет состоять из нескольких шагов, а именно:

- на первом этапе необходимо определить, какие требования предъявляются к программе;
- далее необходимо определить каким образом будут реализованы те алгоритмы, которые описаны в самой задаче;
- на следующем этапе необходимо определить набор команд, которые

помогут достичь поставленной цели;

- далее необходимо проверить программу на работоспособность и отладить ее работу;

- на последнем этапе проводится тестирование разработанной программы.

Во время разработки алгоритма необходимо выбирать такие действия, которые помогут достичь поставленной цели. Существует такие задачи, которые могут быть решены разными способами. В этом случае необходимо использовать отдельные критерии, к примеру, быстрота работы алгоритма, а далее подбирать уже такие решения, которые будут максимально к нему приближены. После этого можно писать подробное описание работы алгоритма.

В процессе отладки программы необходимо найти различные ошибки, которые были допущены и исправить их. Все возможные ошибки можно разделить на две группы. Первая группа ошибок - это семантические ошибки. Сюда относятся такие ошибки, которые связаны с неправильным написанием на языке программирования. Вторая группа ошибок - это синтаксические ошибки. Сюда относятся такие ошибки, которые связаны с неправильным построением алгоритма или самой математической модели [6].

Для того чтобы проверить программу на корректность необходимо провести ее тестирование. Тестирование заключается в том, чтобы вести такие исходные данные, после прохождения которых через программу исход известен. Для того чтобы сравнить программу на корректность необходимо использовать метод сравнения. В данном случае речь идет о сравнении известных данных, с теми данными, которые были получены во время тестирования.

Тестирование является очень важным процессом. Во время тестирования можно выявить смысловые ошибки. Также проследить за тем, насколько корректно оператор выполняет свою работу, насколько значимые

переменные и значимые элементы массива используется в программе, отследить правильность работы, а также правильность обращения к индексированию элементов.

Во время тестирования также можно проверить насколько корректно и правильно работает программа при условии максимального количества данных. Также можно проверить реакцию программу на введение неверных данных.

Программист должен заранее продумать последовательность действий, которые смогут привести к тому результату, который ему необходим.

Алгоритмом называется четкое и предписание, которое определяет процесс перехода от начальных данных, к определенному итогу.

Само предписание также стоит расценивать как некий алгоритм. В качестве основных свойств данного алгоритма можно отметить:

- определенность;
- универсальность;
- продуктивность.

В том программном обеспечении, которое создается, необходимо реализовать два алгоритма:

- первый алгоритм будет отвечать за взаимодействия со справочником;
- второй алгоритм будет отвечать за нахождение отчётов.

В этой базе данных, которая используется для того, чтобы ввести данные используется специальная форма. На данной форме расположены поля, каждому полю присваивают за определенный атрибут.

Технологическое обеспечение (ТО) системы отвечает делению ИС на подсистемы по этапам технологической обработки типов данных.

Анализ данных проходит в несколько этапов:

На первом этапе происходит ввод данных пациентов, здесь же необходимо предусмотреть возможность редактировать имеющихся данных. На втором этапе необходимо сохранить все данные, которые получены. На

третьем этапе происходит сортировка данных, на этом идет процесс обработки имеющейся информации, ее сортировка, и поиск. На последнем этапе полученные данные передаются. Эти данные используются для создания отчетов.

Описанные выше этапы могут использоваться при работе со следующими типами данных:

- при составлении отчетов;
- при записи информации о клиентах;
- при обработке информации о записях клиентов на прием;
- при подготовке документов.

Новые технологии необходимы для того, чтобы была возможность один раз вводить данные о пациенте, в дальнейшем эти данные менять, редактировать, переименовывать. Эти данные можно будет получать в различных формах, в различных документах и отчетах. Появится возможность сводить в единую структуру необходимые данные.

В качестве запросов, которые должен реализовать проект можно отметить:

- запись имеющихся данных;
- обработка тех данных, которые есть в системе;
- передавать данные, которые есть в системе.

Далее предлагаю отметить те операции, которые должны быть реализованы. Данная операция затрагивает начальные данные:

- где будут находиться начальные данные. В данном случае речь идет о приеме такого параметра, который отвечает за процессы и объекты компании. На данном этапе вводятся начальные данные, они берутся со слов заказчика. На основе этих данных в дальнейшем необходимо подсчитать все те параметры, которые необходимы. То есть подсчитать количество, стоимость и так далее;

- запись первичных данных. Данный процесс заключается в том, что все

числовые параметры должны быть перенесены на любой носитель. Для того чтобы решить эту задачу сотрудник регистратуры должен вводить данные в те формы, которые для него представляются-

- сбор данных. В данном случае речь идет об операциях, которые осуществляют работу с данными. Эти операции связаны с набором документов, набором файлов, которые в дальнейшем будут обрабатываться или отправляться. Реализовать эту задачу можно как самостоятельно или же централизованно. Во втором случае речь идет о том, что данные записываются тогда, когда они поступают на источник. Во время поиска начальных данных необходимо подготовить документы, подготовить всю информацию, которая используется во время расчётов. Данный этап очень важен, поэтому ему нужно уделить особое внимание;

- отправка данных на обработку. Данная процедура является типичной. Операции, которые связаны со сбором и регистрацией данных должны выполняться за одним рабочим местом.

Для того чтобы организовать данные процесс необходимо использовать несколько операций, они применяются в строгой последовательности. Необходимо использовать свои методики, которые будут отвечать за обработку данных. В данном случае также могут быть использованы свои инструменты

В качестве примера режима обработки данных, который будет реализован в рассматриваемой базе данных, можно отметить:

- интерактивный. Для его реализации используются прикладные программы, он используется во время ввода информации, во время обработки информации;

- пакетный. Он используется для решения таких задач, которые связаны с небольшой необходимостью вмешиваться в процесс при решении задачи, при этом он имеет возможность работать с большим объемом данных;

- совокупный.

Он используется для решения таких задач, которые связаны с небольшой необходимостью вмешиваться в процесс при решении задачи, при этом он имеет возможность работать с большим объёмом данных. Во время такой обработки данных влияние человеческого фактора на процесс решения задачи практически отсутствует. Человек нужен только для того, чтобы ввести начальные данные. Поэтому такой процесс может проходить практически без участия пользователя. Для решения более сложных задач можно использовать параллельную обработку данных. Данный процесс имеет тесную связь с оформлением документов в привычном бумажном виде [7].

При работе в диалоговом режиме пользователь активно участвует в процессе. Отличительной особенностью диалогового режима является то, что в него данные могут записываться тогда, когда они необходимы.

Учитывая нюансы создаваемой проблемной области при решении задачи, определение такого режима обработки диктуется рядом особенностей:

- те, кто отправляет данные имеют тесную связь с информационной системой, но при этом могут быть использованы и бумажные технологии для передачи данных;

- информация анализируется вовремя ее поступление, но при этом отчеты готовятся только тогда, когда вся необходимая информация поступит;

- процесс обработки данных происходит в определенной последовательности, но при этом человек может вмешаться в данную последовательность.

Можно выделить два направления передачи итоговых данных. В первом случае полученные данные передаются на экран монитора. Во втором случае те данные, которые получены, передаются для печати.

Во всех этих случаях передача данных на базу не исключено. Те данные, которые получаются во время обработки данных, могут быть сохранены в самой базе. В дальнейшем они могут использоваться для того, чтобы решить аналогичные или похожие задачи.

Рассматривая информационное обеспечение под системы необходимо рассматривать то, какую информационную модель выбрал персонал компании для своей работы. Чаще всего выделяют два вида информационной модели. Первый вид - это немашинное обеспечение. Второй вид - это внутримашинное обеспечение.

Само информационное обеспечение состоит из следующих компонентов:

- систем кодирования;
- систем определения;
- систем обобщения;
- информационной базы;

К немашинному обеспечению относятся документы. Все документы, которые есть в компании, можно разделить на выходные документы и на входные документы. В качестве примера входных документов можно привести налоговую отчетность, декларации и так далее. Выходные документы выступают в качестве вторичных документов.

Внутримашинное информационное обеспечение представлено в виде файлов или различных массивов. Оно используют информационную среду, благодаря которой сотрудники компании могут выполнять свои обязанности. Данная область состоит из различных объектов, а также из тех взаимосвязи, которые формируются между данными объектами. Каждый объект имеет свои характеристики и свои свойства.

Существуют определенные требования, которые предъявляются к информационному обеспечению. В качестве основных требований можно отметить:

- используемая операционная система должна иметь возможность поддерживать все функции, которые необходимы компании;
- в тот момент, когда происходит кодировка данных, необходимо предусмотреть возможность использования подтвержденных ID;

- во время командировки необходимо использовать классификаторы такого же уровня, на котором оно используется;

- необходимо использовать такое информационное обеспечение, которая будет полностью совместима с той системой, которая используется в компании;

- формы документов, которые используются в компании должны соответствовать принятым стандартам;

- визуальные формы документов должны соответствовать описанию рабочего места сотрудника, который заполняет данный документ;

- сокращения, который используется в документе должны быть утверждены заранее;

- необходимо обеспечить проверку данных, которую поступают в информационную систему.

В качестве основы информационного обеспечения выступает база информации, в которой хранится и обрабатывается информация. Что будет входить в состав данной базы зависит от того, какие параметры системы выбрано, для чего создаётся данная база, каким образом будут контролироваться работа база. Сама базу необходимо составлять так, чтобы была возможность увеличить на те функции, которые поддерживает база данных.

Рассматривая составляющие базы данных, стоит отметить, что она должна включать в себя:

- различные справочники работа со справочниками осуществляется различными программами;

- таблицы, загрузка данных в таблицы проходит с использованием системы, которая основывается на работе определенного алгоритма. Управлять данной системой может администратор;

- таблиц сохраненной и текущей информации, доступ к которым реализуется на основе ПО системы с указанными правами пользователей.

Необходимо также предусмотреть и обмен данными между разными частями системы.

В разрабатываемом проекте используются такие классификаторы, которые принято у заказчика. В нашем случае речь идет об общих государственных классификаторах, а также о локальных классификаторах. Тут стоит обратить внимание на то, что государственные классификаторы должны соответствовать тем стандартам, которые приняты в стране. Определённые требования предъявляются и к локальным классификаторам. Рассматривая справочники можно отметить, что их всего есть два вида. Первый вид справочников - это линейные справочники. Второй вид справочников - это иерархические справочники.

Также стоит обращать внимание на те требования, которые предъявляются к классификации и процессу кодирования данных. Все системы классификации, которые есть на сегодняшний день, можно разделить на две группы. Первая группа - это многоаспектные классификаторы. Вторая группа - это иерархические классификаторы. Иерархические системы классификации изначально разбивают имеющиеся множество на более мелкие подмножества, после чего они формируют между ними взаимосвязи. Если рассматривать многоаспектные системы классификаторов, то можно отметить, что они используют сразу несколько признаков.

В нашем случае нет необходимости использовать многоаспектные системы классификаторов, поскольку не нужно классифицировать данные сразу по нескольким признакам. Это говорит о том, что в качестве самой эффективной системы будет выступать иерархическая система.

Унифицированная система документации (УСД) выступает в качестве некоего комплекса, которое объединяет в себе различные документы. В данную систему также должны быть включены те требования и те правила, которые предъявляются к данным.

Если рассматривать процесс реального проектирования, то он состоит из

					<i>ПГУ 1.090501.02.001 ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		<i>48</i>

разных частей. Вначале необходимо разработать информационную систему, после этого можно переходить к созданию программных модулей. Данный подход к проектированию системы является дорогим и требует много времени. Поэтому используют его не всегда. Используют его только в том случае если:

- когда необходимо создать полностью новую информационную систему;
- когда необходимо создать доступных прототипов информационной системы;
- когда необходимо разработать такие информационные системы, которые обладают минимальным количеством функций.

Сейчас все чаще используется электронная формула документооборота. Это некий шаблон, который пользователь должен самостоятельно заполнить. Такие формы состоят из кнопок, командных форум, отличительной особенностью их является то, что они могут быть использованы при разных входных данных. Тут стоит обратить внимание на тот, что использовать данные технологии можно только в том случае, если будут использованы узконаправленное программное обеспечение. Далее предлагаю отметить какие формы может включать в себя такая система:

- формы, которые используются для сбора данных, также такие формы, которые позволяют переносить данные из базы данных для дальнейшей их обработки;
- такие формы, которые позволяют собирать данные не только внутри компании, но и за ее пределами.

В качестве информационной базы выступают такие данные, которые организованы определенным способом. Храниться такие данные в памяти компьютера в виде файлов. Эти файлы в дальнейшем могут быть использованы для различных процессов при решении задач.

На сегодняшний день можно выделить разные способы, благодаря

которым можно реализовать информационную базу. Первый способ заключается в том, что локальные файлы группируются. Второй способ заключается в том, что формируется некая информационная база, которую контролируют СУБД.

Локальные файлы, которые существуют, поддерживают оперативную обработку данных. Тут можно столкнуться с тем, что данные могут дублировать. Это делает доступ к данным файлом более сложным, а в некоторых случаях невозможным. Чтобы решить данную проблему можно хранить данные в приложении, которое позволит быстро передавать данные и импортировать в необходимую информацию.

В качестве работы данной системы можно отметить, что необходимо осуществлять повышенный контроль данным, которое вводится. Кроме того, необходимо обеспечить разделение доступа к разным данным между разными пользователями.

В нашем случае самая эффективная будет интегрированная база данных. Это связано с тем, что в компании большой документооборот, при том данные, которые использует компания своей деятельности, много раз могут корректироваться, зачастую они дублируются.

Есть несколько вариантов, которые компания может использовать при организации данных в своей системе базы данных.

Взаимосвязь в иерархической модели реализована совокупностью структурных данных, где тип сегмента потомка связывается лишь с одним типом сегмента предка. Взаимосвязи, которое формируется в иерархической модели разделены на структурные данные. При этом определенный тип сегмента должен быть связан с одним типом сегмента потомка. Такая модель имеет свои отличительные особенности, в качестве одной из отличительной особенности можно отметить то, что не существует явного разделения логических и физических параметров в данной системе. Если пользователь будет вводить неявные запросы, в этом случае будет необходимо перестроить

работу самой базы данных.

Если рассматривать его модель, то в ней можно наблюдать отношения «один ко многим» или же «один к одному».

При использовании реляционной модели результат представлен в виде таблиц. Все те отношения, которые возникают в данной модели, имеют свой ID. Это дает возможность свести к минимуму повторение данных. Отличительной особенностью такой модели является то, что она очень простая в использовании, при том внести изменения в неё достаточно просто. Для этого не требуется много затрат. На сегодняшний день можно выделить два подхода, которые используются при проектировании реляционной базы данных:

Все имеющиеся данные говорят о том, что в нашем случае будет более эффективным использовать именно реляционную модель базы данных.

При разработке информационных систем необходимо использовать структурный подход. Он заключается в том, что все функции должны быть разбиты. То есть система имеет определённый набор функций, которые в свою очередь должны быть разбиты на более мелкие функции в соответствии с теми задачами, которые стоят перед системой. Сам процесс проектирования информационной системы проходит в несколько этапов. Давайте их рассмотрим:

первый этап - это концептуальное проектирование. На данном этапе собирается вся необходимая информация. После того как информация собрана она анализируется. Необходимо определить предметная область, выявить фрагменты, которые должны присутствовать в базе данных, определить объекты базы данных, также какие связи будут формироваться между ними;

второй этап - это логическое проектирование. В данном случае речь идёт о формировании и проектировании тех требования, которые необходимо реализовать в базе данных. Результатом данного этапа выступает спроектированная база данных, которая будет включать в себя специальные

программы;

третий этап - это физическое проектирование. На данном этапе определяются особенности, в котором будут храниться информация, определяются те методы, на основе которых будут предоставлены доступы к данным для разных пользователей.

В качестве основных элементов в модели выступают:

- сущности;
- свойства;
- атрибуты;
- связи.

Сущность - это все объекты, которые присутствуют в базе данных. То есть сущность - это объекты, которые хранятся в базе данных. Также необходимо обращать внимание на логическую структуру. Она описывает состав базы данных, её типа, а также те связи, которые в ней формируются.

Построение инфологической модели - это сложный процесс, который состоит из нескольких этапов:

- на первом этапе необходимо определить сущностью;
- на втором этапе необходимо определить, какие взаимосвязи должны формироваться между сущностями;
- на третьем этапе необходимо сформировать задание для альтернативных и первичных ключей;
- на следующем этапе необходимо выявить те атрибуты, которые будут характерны для сущностей;
- на последнем этапе переводится модель к тому и вводу, в котором она должна быть.

Логический уровень представления модели - это абстрактное представление данных, где данные представлены так, как они выглядят в реальном мире. Логическая модель базы данных представлена на рисунке 1. Описание базы данных приведено в таблице 2.

Тип модуля	Название таблицы	Описание
Таблицы данных	<i>chat</i>	Чаты пользователя
	<i>chat_message</i>	Сообщения чата
	<i>chat_user</i>	Состав чата
	<i>project</i>	Проекты
	<i>project_user</i>	Состав участников проекта
	<i>workgroup</i>	Рабочие группы проекта
	<i>workgroup_document</i>	Документы рабочей группы
	<i>workgroup_message</i>	Сообщения рабочей группы
	<i>workgroup_task</i>	Задачи рабочей группы
	<i>workgroup_user</i>	Состав рабочей группы

Для каждой таблицы базы данных создается модель (*Model* из *Model-View-Controller*). Все модели нашего проекта располагаются в директории «*basic\models*». Описание соответствия моделей таблица базы данных приведено в таблице 3.

Таблица 3 - Описание моделей для таблиц

Тип модуля	Название таблицы	Название модели
Справочники	<i>project_status</i>	<i>ProjectStatus</i>
	<i>task_status</i>	<i>TaskStatus</i>
	<i>user_role</i>	<i>UserRole</i>
	<i>workgroup_status</i>	<i>WorkgroupStatus</i>
	<i>user</i>	<i>User</i>
	<i>user_type</i>	<i>UserType</i>
	<i>user_function</i>	<i>UserFunction</i>
Таблицы данных	<i>chat</i>	<i>Chat</i>
	<i>chat_message</i>	<i>ChatMessage</i>
	<i>chat_user</i>	<i>ChatUser</i>
	<i>project</i>	<i>Project</i>
	<i>project_user</i>	<i>ProjectUser</i>
	<i>workgroup</i>	<i>Workgroup</i>
	<i>workgroup_document</i>	<i>WorkgroupDocument</i>
	<i>workgroup_message</i>	<i>WorkgroupMessage</i>
	<i>workgroup_task</i>	<i>WorkgroupTask</i>
	<i>workgroup_user</i>	<i>WorkgroupUser</i>

Большинство контроллеров нашего проекта (кроме контроллера *ClassifierController*) реализует *CRUD* - (англ. *create, read, update, delete* — «создание, чтение, обновление, удаление») сокращённое именование 4-х базовых функций, используемых при работе с наборами данных.

Консоль управления *Yii2* позволяет для конкретной модели выполнить генерацию *CRUD*. Генератор *CRUD* создает контроллер, набор действий

контроллера («Index», «View», «Create», «Update», «Delete») и ряд видов («View») соответствующих действиям.

Контроллеры нашего проекта находятся в папке «*basic\controllers*».

Представления, соответствующие действиям контроллеров, располагаются в папке «*basic\views*». Описание базовых функций контроллера приведено в таблице 4. Описание контроллеров приведено в таблице 5.

Таблица 4 - Описание базовых действий контроллеров

Название действия	Описание действия
<i>Index</i>	Метод просмотра списка всех записей
<i>View</i>	Метод просмотра записи
<i>Create</i>	Метод добавления записи
<i>Update</i>	Метод редактирования записи
<i>Delete</i>	Метод удаления записи

Таблица 5 - Описание контроллеров системы

Тип модуля	Название модели	Название контроллера	Действия контроллера
Системные	Использует различные модели (модели фреймворка Yii 2.0)	<i>SiteController</i>	<i>Index</i> (главная страница системы), <i>Login</i> (вход), <i>Logout</i> (выход), <i>Signup</i> (регистрация), <i>RequestPasswordReset</i> (отправка пользователю на email ссылки для сброса пароля), <i>ResetPassword</i> (метод ввода нового пароля)
Справочники	<i>ProjectStatus</i>	<i>ProjectStatusController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>TaskStatus</i>	<i>TaskStatusController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>UserRole</i>	<i>UserRoleController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>WorkgroupStatus</i>	<i>WorkgroupStatusController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>User</i>	<i>UserController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>UserType</i>	<i>UserTypeController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>UserFunction</i>	<i>UserFunctionController</i>	<i>Index, View, Create, Update, Delete</i>
Данные	<i>Chat</i>	<i>ChatController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>ChatMessage</i>	<i>ChatMessageController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>ChatUser</i>	<i>ChatUserController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>Project</i>	<i>ProjectController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>ProjectUser</i>	<i>ProjectUserController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>Workgroup</i>	<i>WorkgroupController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>WorkgroupDocument</i>	<i>WorkgroupDocumentController</i>	<i>Index, View, Create, Update, Delete, Download</i>
	<i>WorkgroupMessage</i>	<i>WorkgroupMessageController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>WorkgroupTask</i>	<i>WorkgroupTaskController</i>	<i>Index, View, Create, Update, Delete</i>
	<i>WorkgroupUser</i>	<i>WorkgroupUserController</i>	<i>Index, View, Create, Update, Delete</i>

Работа пользователя с системой осуществляется через обращения к действиям контроллеров. Пример обращения пользователя к действию «view» (просмотр) контроллера «project» (проект):

http://localhost:7081/basic/web/project/view/1

Т.е. в данном случае обращение пользователя к действию системы осуществляется через *GET* запрос протокола *HTTP*. Результатом данного действия является *html*-код, который генерирует сервер на запрос пользователя.

Диаграмма классов системы представлена на рисунке 2.

Спроектировать интерфейс пользователя является очень важным процессом при создании цифровых продуктов. Он может заключаться в разработке десктопных приложений или же разработкой веб-сайта. Данная стадия играет большую роль в жизни компании, она оказывает непосредственный эффект на деятельность компании. То есть от разработки данного интерфейса может зависеть развитие всего продукта на рынке.

Сейчас есть много разных программистов. Но мало кто может похвастаться успехами в данной сфере. Те интерфейсы, которые есть сейчас зачастую до конца не продуманны, в них можно найти много разных мелочей, которые необходимо дорабатывать. Именно мелочи могут приносить неудобство пользователям, и как следствие привести к тому, что компания потеряет клиентов. Поэтому данную проблему можно назвать актуальной. Ее необходимо решать.

Сейчас есть много разных программистов. Но мало кто может похвастаться успехами в данной сфере. Те интерфейсы, которые есть сейчас зачастую до конца не продуманны, в них можно найти много разных мелочей, которые необходимо дорабатывать. Именно мелочи могут приносить неудобство пользователям, и как следствие привести к тому, что компания потеряет клиентов. Поэтому данную проблему можно назвать актуальной. Ее необходимо решать.

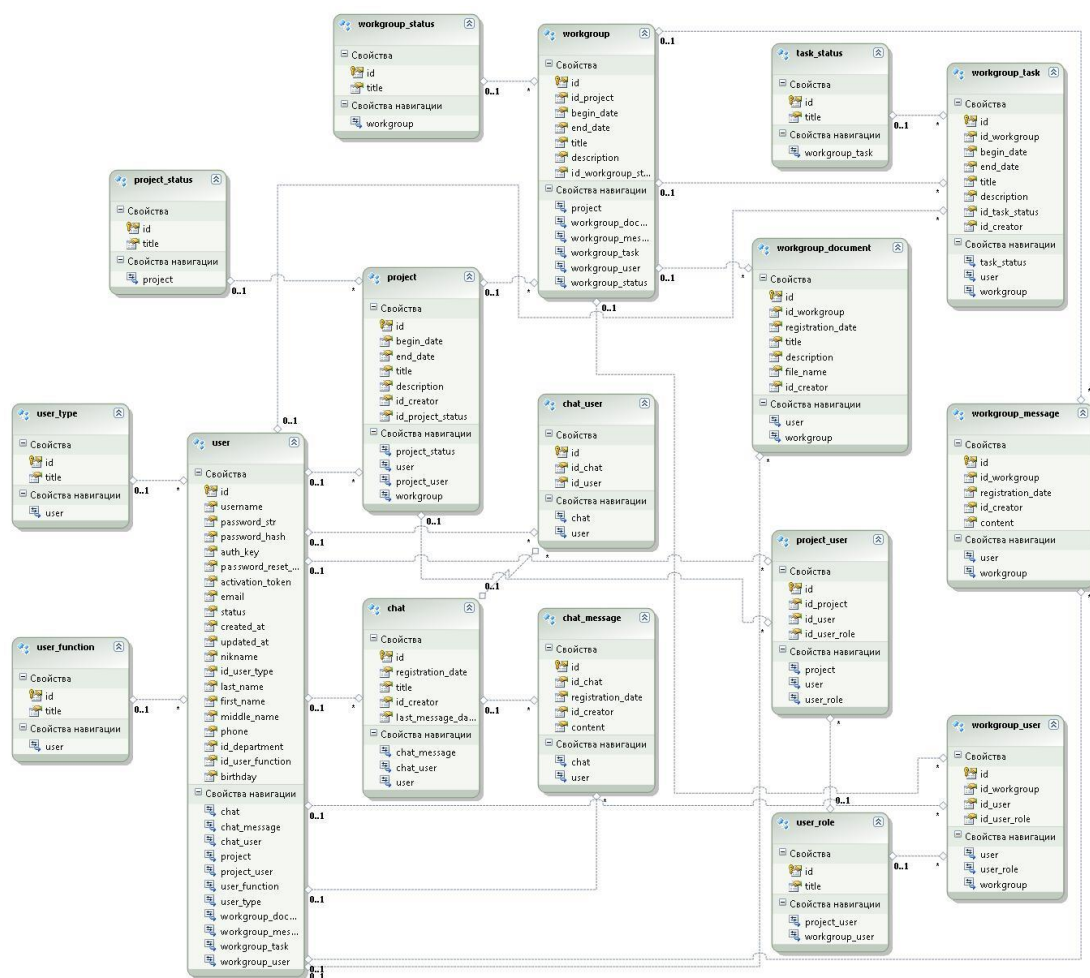


Рисунок 2 - Диаграмма классов моделей

В качестве основной задачи, которая стоит во время подготовки интерфейса можно отметить то, что необходимо разработать такое программное обеспечение, которое будет понятным и логичным для пользователей. Разработанный интерфейс должен быть не только понятным, но и удобным для пользователей, он должен вызывать доверие. Это даст возможность не только качественно обслуживать имеющихся клиентов, но и привлекать новых. Разработанное программное обеспечение должно помогать пользователю. Оно должно делать работу пользователя более удобной. Оно должно экономить время клиента.

Подготовка программного обеспечения является очень сложной задачей. Этот процесс занимает много времени и проходит в несколько этапов. В начале необходимо выявить, что от интерфейса хочет сам заказчик.

Необходимо описать проект и составить бизнес план.

Бизнес-план — это некий брифинг. Для его составления необходимо собрать много данных, составить конкретное техническое задание и описать требования, которые предъявляются к будущему интерфейсу.

Так же бриф должен содержать в себе информацию о целевой категории, о потенциальных клиентов. В нем отражается рабочая область, поэтому он играет очень большую роль, и оказывает непосредственное влияние на успех всего проекта в целом.

На следующем этапе необходимо более четко и основательно изучить целевую аудиторию. Большое внимание необходимо уделить описанию ЦА для того программного обеспечения, которое создается. Обратить внимание необходимо и на различные нюансы, которые помогут решить задачу более эффективно как для заказчика, так и для клиента. Полученные данные смогут создать удобный интерфейс, при этом можно будет учесть предпочтения не только заказчика, но и самих пользователей.

После того как собраны все необходимые данные можно переходить к следующему этапу, то есть к непосредственной разработке. В качестве основного параметра, который оказывает влияние на взаимодействие пользователя, выступают опыт взаимодействия.

Опыт взаимодействия является ощущением, которое возникает у человека при взаимодействии непосредственными объектами мира. Опыт взаимодействия показывает, какие продукты, в какой период времени оказывают влияние на эмоциональные параметры потенциальных клиентов, на их психологические параметры. Если данный опыт будет принимать положительные значения, то потенциальные клиенты будут более лояльно относиться к компании, будет советовать компанию, тем самым приобрела компании становится больше, и компания может развиваться.

Существуют определенные качества, которые необходимо учитывать и использовать во время создания программного обеспечения:

- юзабилити. Здесь речь идет о том, что интерфейс должен быть удобным пользователю, он должен давать возможность пользователю решать те задачи, которые перед ним стоят. Правильное юзабилити будет в том случае, если будет правильно сформирована архитектура, если интерфейс будет логичным, понятным пользователю, если контент будет читаемым, если он будет призывать потенциального пользователя совершить определенные действия;

- ценность. На сегодняшний день существует много программного обеспечения, но при этом ценность данного программного обеспечения неизвестна. Чаще всего это связано с тем, что функция такого программного обеспечения не соответствует потребностям пользователей. Ценным будет тот продукт, который максимально удовлетворяет потребности пользователя. При этом необходимо уделять внимание не только внешнему виду, но и функциональным возможностям;

- доступность. В данном случае речь идет о том, что программное обеспечение должно быть доступным для пользователей. Часто данное свойство оказывает влияния на то, будут ли пользователи покупать разработанное программное обеспечение. Неправильно разработанное программное обеспечение приводит к потерям клиентов, а значит снижает прибыль компании. Поэтому данному свойству необходимо уделять особое внимание, необходимо заранее определить возможные проблемы, с которым придется столкнуться во время реализации программного обеспечения, а также найти пути решения найденных проблем;

- привлекательность. Данный аспект оказывает влияние на эмоциональную часть взаимодействия. Сегодня есть много продуктов, которые интересны пользователям, даже несмотря на то, что в них есть много недостатков. Объясняется такой интерес тем, что они привлекательны. В качестве примера можно привести видеоигры. Фанаты таких игр будут тратить много времени на игру, которая отличается своей графикой, в которой есть

интересный сюжет и имеется спецэффекты. При этом зачастую и неважно насколько это доступно, насколько это правильно организовано, какая ценность у данного продукта.

Далее необходимо определить, каким функционалом будет обладать разрабатываемое программное обеспечение. Для этого необходимо составить пользовательские истории, а также описать путешествия пользователя.

Пользовательская история - это такой метод, который позволяет описать те требования, предъявляемые к информационной системе. Они могут выступать как в качестве нескольких подготовленных предложений, также могут быть описаны на повседневном языке или на языке пользователя.

Если говорить другими словами то пользовательская история - это те намерения, которые необходимо реализовать. Сюда входят сценарии, функциональные параметры и другие параметры будущей системы.

Разработанная история выступает в качестве системы для дальнейшей разработки, ей пользуются разработчики. Использование данного подхода является очень удобным и очень информативным не только для разработчиков, но и для самих заказчиков.

Данные требования должны быть реализованы. Это связано с тем, что новое программное обеспечение и его интерфейс должны отмечать всем требованиям, которые предъявляются со стороны пользователя. То есть можно говорить о том, что новый интерфейс должен учитывать:

- стилевые особенности пользователя;
- когнитивные особенности пользователя;
- эргономичные особенности пользователя.

Оттого насколько данные особенности будут учитывать интересы пользователей будет зависеть то, насколько эффективно сможет работать новая система.

В ряде случаев можно использовать знакомые пользователю понятия. Тогда пользователь на интуитивном уровне будет делать определенный

выбор. Знакомые понятия могут быть использованы в разных элементах интерфейса. В данном случае речи идет о стереотипном восприятии.

Но такого подхода недостаточно при создании эффективного интерфейса. Зачастую необходимо использовать более глубокий подход. Необходимо использовать такие задачи инженерной технологии как:

- проводить анализ когнитивно-стилевых характеристик пользователя в системе «человек-компьютер», изучение структуры и классификация деятельности пользователя;

- изучать то, как пользователь воспринимает информацию;

- использовать принцип, который часто называют «человеко-компьютерного» взаимодействия;

- изучать как психологические факторы влияют на принцип «человек-компьютер»;

- проектирование, создание и исследование моделей «человеко-компьютерного» взаимодействия.

2 Технико-экономическое обоснование разработки программы

2.1 Организационно-правовая характеристика ПП

Оптимальное решение разработчик находит, применяя ряд критериев. Что касается проверки программного приложения, то оно заключается в обнаружении им последующем устранении ошибок и всех неточностей. Сами ошибки бывают как семантическими, так и синтаксическими. При этом, синтаксические ошибки, которые непосредственно связаны с неверным написанием операторов языка программирования, находятся исключительно только на этапе трансляции программы. В свою очередь семантические ошибки чаще всего связаны с неправильным построением алгоритмической и математической модели в процессе разработки.

Подобные ошибки обнаруживаются не сразу, так как их наличие не мешает программной системе функционировать и даже выдавать результаты. Тестирование дает возможность доказать непосредственно факт наличия определенной смысловой ошибки. Для определения смысловых ошибок используют профильные специалисты специальные отладчики. При тестировании системы важно проверить насколько адекватно и точно работает программа при условии большого количества входных наборов данных.

Алгоритмизация является главным шагом всего процесса программирования. При этом, основная задача программиста заключается непосредственно в продумывании точной и верной последовательности действий, выполняя которые можно быстро получить задуманный результат. Алгоритм представляет собой максимально четкое предписание, которое непосредственно отражает процедуру перехода от начальных данных к финалу. В частности, предписание можно с полным правом считать алгоритмом, в том случае, если оно имеет следующие три свойства:

- максимальная определенность;

					ПГУ 1.090501.02.001 ПЗ	Лист
						62
Изм.	Лист	№ докум.	Подпись	Дата		

- универсальность;
- продуктивность.

Непосредственно в разрабатываемом программном обеспечении нужно обязательно внедрить следующие алгоритмы:

- алгоритм обнаружения отчетных данных;
- алгоритм, обеспечивающий должное взаимодействие со справочниками.

Непосредственно, исходя из используемой модели базы данных и для непосредственного ввода данных, используют особые поля и формы, специально для этого предназначенные. В каждое из полей форм переносится определенный атрибут.

При составлении плана тестирования модулей программы были разработаны наборы тестовых данных в виде функциональных пар. Функциональная пара – это определённый тип результата работы модуля, соответствующий конкретным входным данным. В таблице 6 приведен перечень некоторых тестовых данных, рассмотренных во время тестирования программы.

Таблица 6 - Набор тестовых данных

Входные данные	Ожидаемый результат	Полученный результат	Меры
Ввод данных о враче	Корректный ввод и сохранение данных	Корректный ввод Отсутствует подтверждение сохранения данных	Добавлено подтверждение об успешном сохранении данных
Последняя резервная копия базы	Корректная Загрузка данных	Загружается предыдущая копия базы	Исправлено условие сверки копий баз
Параметры для удаления данных	Удаление сведений и вывод сообщения об успешном удалении	Удаление сведений и вывод сообщения об успешном удалении. Возможно ошибочное удаление данных	Программа сработала корректно. Добавлено диалоговое окно подтверждения запроса на удаление
Параметры поиска данных о враче	Результат поиска, полностью отвечающий запросу	В результате поиска представлены данные обо всех однофамильцах	Добавлены условия проверки соответствия полей имени и отчества

Входные данные	Ожидаемый результат	Полученный результат	Меры
Запрос на предоставление сводной таблицы по услугам	Получение требуемых данных	Получение требуемых данных. Форма представления отчета неудобна для восприятия	Программа сработала корректно. Доработано представление данных в <i>Excel</i>
Ввод данных об	Корректный ввод данных, удачная синхронизация сведений	Корректный ввод данных, некорректная синхронизация (при не заполненном разделе практика - синхронизация невозможна)	Реализован вывод сообщения о причинах ошибки или невозможности синхронизации данных

После многократного проведения автономного тестирования модулей и интегрального тестирования всей программы, с исправлением выявленных в результате каждого тестового прогона ошибок, можно сделать вывод, что программа работает корректно.

2.1 Планирование разработки

Разработка информационной подсистемы - это трудоемкий процесс, включающий в себя множество этапов: разработка технического задания, планирование производства, проектирование системы, разработку основных программных модулей, тестирование системы, отладку и внедрение созданного программного продукта.

Проводить анализ выбранной информационной системы, необходимо опираться на разные показатели. Необходимо определять эффективность работы новой системы, определять насколько она оправдана с экономической точки зрения. Насколько выбранная система полезна можно оценивать по-разному, но самой эффективной оценкой будет та оценка, которая выражена в денежном эквиваленте. То есть необходимо оценивать доходы компании до внедрения данной системы, и после того когда система была введена.

Для того чтобы оценить эффективность информационной системы можно использовать метод обобщенной оценки информационной системы.

Такой метод состоит из использования неформальных и формальных процедур.

Компания, для того чтобы внедрить информационную систему, должна потратить немало средств. Процесс изучения эффективности информационной системы - это не только изучение экономических и технических навыков, но и оценка того, как новые технологии помогут развиваться компании, какому результату они приведут.

На сегодняшний день на рынке наблюдается высокая конкуренция. Поэтому необходимо в первую очередь обращать внимание на то, на сколько информационная система результативна, какой экономический эффект она принесет компании. Также стоит обращать внимание и на окупаемость. Многие компании не могут оценить качество выбранной информационной системы, поскольку на процессе внедрения допускаются очень важные ошибки.

В первую очередь необходимо дать оценку новому проекту, эксперты, которые принимают участие в создании и поддержке информационной системы, должны контролировать систему, они должны оценивать какие пути для развития есть у компании.

Оценка результативности работы информационной системы должна проходить в несколько этапов. Давайте их отметим:

- на первом этапе необходимо дать классическую оценку, проанализировать расходы и доходы компании на внедрение системы;
- на втором этапе необходимо проанализировать, стоимость владения данной системы;
- на третьем этапе необходимо проанализировать информационную систему с точки зрения инвестиций;
- на четвертом этапе необходимо определить показатели, которые будут говорить о экономической стороне системы.

Оценку необходимо проводить не в общем, а для каждого принципа

отдельно. То есть регионального принципа, технического принципа и для денежного принципа необходимо проводить разные оценки. Во время оценки необходимо:

- оценивать эффективность выбранной системы, и длительность её использования;
- описывать те последствия, которые будет испытывать на себе компания после введения системы;
- анализировать каким образом инфляция повлияет на деятельность системы;
- анализировать денежные потоки компании;
- описание проекта в рамках всего ЖЦ;
- анализировать то время, которое компания потратит на реализацию проекта;
- проанализировать какие доходы компании получат от проекта, какие расходы она должна будет принести;
- дать описание проекта, описать из чего он состоит;
- описать аналогичные проекты, выделить тот, который будет самым эффективным;
- провести многоуровневую оценку;
- определить влияние проекта на компанию с финансовой точки зрения;
- определить какие сторонние факторы могут оказать влияние на деятельность системы.

Те данные, которые будут получены в итоге пробного анализа эффективности проекта, должны использоваться во время разработки информационной системы. Но тут также необходимо учитывать и эффективность информационной системы. Для этого необходимо ее оценить по следующим показателям:

- какую совокупную прибыль может принести проект;
- какой совокупный доход может принести проект;

- процент доходности;
- проанализировать какие денежные вложения необходимы для реализации проекта;
- за какой период времени проект окупится.

Все это говорит о том, что процедура анализа эффективности очень сложная, она неоднозначная. Для разных компаний, для разных случаев можно использовать свой подход. В некоторых случаях стоит помнить, что минимальное влияние человека может положительно сказаться на работе всей системы.

Наша система создается для того, чтобы использовать ее в продаже они в работе какой-либо компании. Это необходимо учесть. Во время анализа необходимо сравнить данные, которые связаны с внедрением предложенной системы в деятельность компании. То есть в данном случае необходимо рассматривать два параметра. Первый параметр - это падение стоимости. Второй параметр - это снижение трудозатрат.

Разработка программного обеспечения - это некоторая деятельность, во время которой используются различные технические достижения. Для осуществления такой деятельности необходимы определенные знания. Существуют разные факторы, которые приводят к тому, что разработка программного обеспечения содержит в себе элементы неопределенности. Часто такую неопределенность расценивают как риски. Успех проекта зависит от того какую степень имеют разные риски, каким образом и насколько сильно они могут влиять на деятельность проекта. Необходимо не только выявлять риски, но и научиться ими управлять.

Многие проекты, которые используют программное обеспечение, создается для того, чтобы отличаться, чтобы получить новые функции, чтобы использовать свои возможности более эффективно. Проекты, которые связаны с разработкой программного обеспечения, всегда сопровождаются с различными рисками.

Риски в данном случае встречается довольно часто, поэтому большое внимание необходимо уделять работе с ними. Необходимо не только выявлять возможные риски, но и максимально смягчать их влияние на проект. Это связано с тем, что риски могут угрожать успеху проектов. Если проект имеет временные или финансовые ограничения, то в таких проектах особое внимание необходимо уделять снижению рисков на деятельность компании.

Во время разработки проекта необходимо учитывать риски, которые могут возникать на разных этапах внедрения проекта. Далее предлагаю описать основные риски и те шаги, которые помогут снизить их влияние на систему.

Этап проектирования проекта:

- риски, которые имеют связь с работниками компании. В качестве примера таких рисков можно отметить:

- 1) сотрудники компании не имеют необходимого опыта;
- 2) мотивация сотрудников компании очень слабая;
- 3) сотрудники компании не понимают, для чего происходит данный процесс, какие цели он преследует;
- 4) коммуникации среди сотрудников в компании не настроены;

- к способам, которые помогут свести к минимуму влияние данных рисков можно отметить:

- 1) обучение новых сотрудников компании более опытными коллегами;
- 2) применение руководством компании различных мотивационных бонусов;
- 3) доведение до сотрудников компании четких должностных инструкций;

- вторая группа риска связана с тем, что будут приняты неправильные решения. В качестве основных рисков данного типа можно отметить:

- 1) неправильно оцененные ресурсы;

- 2) разработка функционала, который не нужен;
- 3) неправильный выбор технических средств;
- 4) неполное, неточное выполнение технического задания, которое было предоставлено заказчиком;

- для того чтобы минимизировать эти риски необходимо:

- 1) составить четкий план, по которому будет разрабатываться проект;
- 2) обеспечить наличие ресурсов, необходимых для реализации проекта;
- 3) установить границы проекта;
- 4) согласовывать принятые решения с заказчиком;

- риски, с которыми можно столкнуться во время планирования проектов. В качестве основных рисков можно отметить:

- 1) неправильно определенные сроки реализации;
- 2) не до конца разработанный план.

- для минимизации данных рисков, необходимо:

- 1) провести независимое исследование;
- 2) обеспечить доступ всех сотрудников компании к имеющейся документации по проекту;

- риски, которые связанные с сотрудниками компании. В качестве основных рисков можно отметить:

- 1) отсутствие достаточной мотивации сотрудников;
- 2) уход ценных сотрудников;
- 3) не понимание сотрудниками компании, с какой целью проводится данный процесс;
- 4) утечка корпоративных данных;
- 5) плохую организацию коммуникации внутри коллектива;

- для минимизации этих рисков, необходимо:

- 1) тщательно подбирать персонал;

2) организовать в компании систему мотивации;

3) проводить мероприятия по сплочению коллектива;

- риски, с которыми можно столкнуться в связи с возникновением сбоев в системе. В качестве основных рисков можно отметить:

1) некоторые компоненты системы не будут работать;

2) система работает неправильно;

- для того чтобы снизить данные риски, необходимо:

1) внимательно выбирать систему, которая будет использоваться;

2) периодически создавать резервные копии;

3) использовать проверенные системы;

4) анализировать и выделять слабые места системы;

- риски, связанные с сотрудниками. В качестве основных рисков можно отметить:

1) не налаженная связь между разработчиками и сотрудниками компании;

2) недостаточная квалификация сотрудников, то есть сотрудники компании не имеют тех знаний, которые необходимы для работы с системой;

- для минимизации обозначенных рисков необходимо:

1) проводить обучение персонала;

2) создать план, по которому будет проходить процесс автоматизации, и четко следовать ему;

3) рассказать сотрудникам о преимуществах новой системы.

- в качестве технических рисков можно отметить потерю данных. Для того чтобы снизить данные риски, можно прибегнуть к помощи профессионалов при разработке системы.

На этапе эксплуатации и сопровождения можно столкнуться со следующими рисками:

- технические риски. В качестве основных можно отметить:

					<i>ПГУ 1.090501.02.001 ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		<i>70</i>

- 1) система работает с ошибками;
- 2) некорректная работа старого функционала после обновления;
- 3) оборудование используется неправильно;

- для того чтобы снизить данные риски, необходимо:

- 1) выявлять и своевременно исправлять ошибки в системе;
- 2) вовремя обновлять документацию.

Программные средства вычислительной техники являются материальным объектом интеллектуальной и трудовой деятельности человека, состоящим из программной и документально оформленной частей, реализующих свои потребительские свойства и качества в составе функционирующих вычислительных систем или систем обработки данных и информации.

Проводить анализ выбранной информационной системы, необходимо опираться на разные показатели. Необходимо определять эффективность работы новой системы, определять насколько она оправдана с экономической точки зрения. Насколько выбранная система полезна можно оценивать по-разному, но самой эффективной оценкой будет та оценка, которая выражена в денежном эквиваленте. То есть необходимо оценивать доходы компании до внедрения данной системы, и после того когда система была введена.

Для того чтобы оценить эффективность информационной системы можно использовать метод обобщенной оценки информационной системы. Такой метод состоит из использования неформальных и формальных процедур.

Компания, для того чтобы внедрить информационную систему, должна потратить немало средств. Процесс изучения эффективности информационной системы - это не только изучение экономических и технических навыков, но и оценка того, как новые технологии помогут развиваться компании, какому результату они приведут.

На сегодняшний день на рынке наблюдается высокая конкуренция.

Поэтому необходимо в первую очередь обращать внимание на то, на сколько информационная система результативна, какой экономический эффект она принесет компании. Также стоит обращать внимание и на окупаемость. Многие компании не могут оценить качество выбранной информационной системы, поскольку на процессе внедрения допускаются очень важные ошибки.

В первую очередь необходимо дать оценку новому проекту, эксперты, которые принимают участие в создании и поддержке информационной системы, должны контролировать систему, они должны оценивать какие пути для развития есть у компании.

Оценка результативности работы информационной системы должна проходить в несколько этапов:

- на первом этапе необходимо дать классическую оценку, проанализировать расходы и доходы компании на внедрение системы;
- на втором этапе необходимо проанализировать, стоимость владения данной системы;
- на третьем этапе необходимо проанализировать информационную систему с точки зрения инвестиций;
- на четвертом этапе необходимо определить показатели, которые будут говорить о экономической стороне системы.

Оценку необходимо проводить не в общем, а для каждого принципа отдельно. То есть регионального принципа, технического принципа и для денежного принципа необходимо проводить разные оценки. Во время оценки необходимо:

- оценивать эффективность выбранной системы, и длительность её использования;
- описывать те последствия, которые будет испытывать на себе компания после введения системы;
- анализировать каким образом инфляция повлияет на деятельность

системы;

- анализировать денежные потоки компании;
- описание проекта в рамках всего ЖЦ;
- анализировать то время, которое компания потратит на реализацию проекта;
- проанализировать какие доходы компании получают от проекта, какие расходы она должна будет принести;
- дать описание проекта, описать из чего он состоит;
- описать аналогичные проекты, выделить тот, который будет самым эффективным;
- провести многоуровневую оценку;
- определить влияние проекта на компанию с финансовой точки зрения;
- определить какие сторонние факторы могут оказать влияние на деятельность системы.

Те данные, которые будут получены в итоге пробного анализа эффективности проекта, должны использоваться во время разработки информационной системы. Но тут также необходимо учитывать и эффективность информационной системы. Для этого необходимо ее оценить по следующим показателям:

- какую совокупную прибыль может принести проект;
- какой совокупный доход может принести проект;
- процент доходности;
- проанализировать какие денежные вложения необходимы для реализации проекта;
- за какой период времени проект окупится.

Все это говорит о том, что процедура анализа эффективности очень сложная, она неоднозначная. Для разных компаний, для разных случаев можно использовать свой подход. В некоторых случаях стоит помнить, что минимальное влияние человека может положительно сказаться на работе всей

					ПГУ 1.090501.02.001 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		73

системы.

Наша система создается для того, чтобы использовать ее в продаже они в работе какой-либо компании. Это необходимо учесть. Во время анализа необходимо сравнить данные, которые связаны с внедрением предложенной системы в деятельность компании. То есть в данном случае необходимо рассматривать два параметра. Первый параметр - это падение стоимости. Второй параметр - это снижение трудозатрат.

На основании проведения экспертных оценок с учетом состава коллектива разработчиков были определены их планируемые продолжительности.

В таблице 7 представлено участие специалистов в разработке ИС.

Таблица 7 - Наименование работ по этапам разработки ИС

Наименование работ	Обозначение	Кол-во дней
Формирование требований	$T_{сб}$	2
Разработка концепции	T_k	1,5
Разработка технического задания	$T_{тз}$	1,5
Разработка клиентской части	$T_{кл}$	10
Тестирование системы	T_t	2,5
Отладка системы	$T_{отл}$	2,5
Написание сопроводительной документации	$T_{док}$	2,5
Итого:		22,5

Помимо представленных задач, руководитель осуществляет контроль за ходом выполнения работ по проекту разработки ИС на всех ее этапах.

На таблице 8 представлено предполагаемое распределение участников по этапам разработки ИС.

Таблица 8 - Распределение участников по этапам разработки ИС

Обозначение значения	Кол-во дней	Задействованные сотрудники предприятия в проекте		
		Руководитель	Программист	Специалист
$T_{сб}$	2	2	-	1
T_k	1,5	1,5	-	1
$T_{тз}$	1,5	1	-	0,5

Обозначение значения	Кол-во дней	Задействованные сотрудники предприятия в проекте		
		Руководитель	Программист	Специалист
T _{кл}	2,5	-	10	-
T _т	2,5	1	0,5	1,5
T _{отл}	2,5	-	-	2,5
T _{док}	2	-	-	2,5
Итого:	22,5	5,5	10,5	9

Далее, на основании предполагаемой продолжительности этапов разработки ИС и состава задействованных на них исполнителей, были определены прогнозируемые затраты на оплату труда привлеченных специалистов.

2.2 Описание бюджета на разработку проекта

Проводить анализ выбранной информационной системы, необходимо опираться на разные показатели. Необходимо определять эффективность работы новой системы, определять насколько она оправдана с экономической точки зрения. Насколько выбранная система полезна можно оценивать по-разному, но самой эффективной оценкой будет та оценка, которая выражена в денежном эквиваленте. То есть необходимо оценивать доходы компании до внедрения данной системы, и после того когда система была введена.

Для того чтобы оценить эффективность информационной системы можно использовать метод обобщенной оценки информационной системы. Такой метод состоит из использования неформальных и формальных процедур.

Компания, для того чтобы внедрить информационную систему, должна потратить немало средств. Процесс изучения эффективности информационной системы - это не только изучение экономических и технических навыков, но и оценка того, как новые технологии помогут развиваться компании, какому результату они приведут.

На сегодняшний день на рынке наблюдается высокая конкуренция.

Поэтому необходимо в первую очередь обращать внимание на то, на сколько информационная система результативна, какой экономический эффект она принесет компании. Также стоит обращать внимание и на окупаемость. Многие компании не могут оценить качество выбранной информационной системы, поскольку на процессе внедрения допускаются очень важные ошибки.

В первую очередь необходимо дать оценку новому проекту, эксперты, которые принимают участие в создании и поддержке информационной системы, должны контролировать систему, они должны оценивать какие пути для развития есть у компании.

Оценка результативности работы информационной системы должна проходить в несколько этапов:

- на первом этапе необходимо дать классическую оценку, проанализировать расходы и доходы компании на внедрение системы;
- на втором этапе необходимо проанализировать, стоимость владения данной системы;
- на третьем этапе необходимо проанализировать информационную систему с точки зрения инвестиций;
- на четвертом этапе необходимо определить показатели, которые будут говорить о экономической стороне системы.

Оценку необходимо проводить не в общем, а для каждого принципа отдельно. То есть регионального принципа, технического принципа и для денежного принципа необходимо проводить разные оценки. Во время оценки необходимо:

- оценивать эффективность выбранной системы, и длительность её использования;
- описывать те последствия, которые будет испытывать на себе компания после введения системы;
- анализировать каким образом инфляция повлияет на деятельность

системы;

- анализировать денежные потоки компании;
- описание проекта в рамках всего ЖЦ;
- анализировать то время, которое компания потратит на реализацию проекта;
- проанализировать какие доходы компании получают от проекта, какие расходы она должна будет принести;
- дать описание проекта, описать из чего он состоит;
- описать аналогичные проекты, выделить тот, который будет самым эффективным;
- провести многоуровневую оценку;
- определить влияние проекта на компанию с финансовой точки зрения;
- определить какие сторонние факторы могут оказать влияние на деятельность системы.

Те данные, которые будут получены в итоге пробного анализа эффективности проекта, должны использоваться во время разработки информационной системы. Но тут также необходимо учитывать и эффективность информационной системы. Для этого необходимо ее оценить по следующим показателям:

- какую совокупную прибыль может принести проект;
- какой совокупный доход может принести проект;
- процент доходности;
- проанализировать какие денежные вложения необходимы для реализации проекта;
- за какой период времени проект окупится.

Все это говорит о том, что процедура анализа эффективности очень сложная, она неоднозначная. Для разных компаний, для разных случаев можно использовать свой подход. В некоторых случаях стоит помнить, что минимальное влияние человека может положительно сказаться на работе всей

					ПГУ 1.090501.02.001 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		77

системы.

Наша система создается для того, чтобы использовать ее в продаже они в работе какой-либо компании. Это необходимо учесть. Во время анализа необходимо сравнить данные, которые связаны с внедрением предложенной системы в деятельность компании. То есть в данном случае необходимо рассматривать два параметра. Первый параметр - это падение стоимости. Второй параметр - это снижение трудозатрат.

Затраты на разработку ИС определяются по следующим основным статьям:

- заработная плата разработчиков:

- 1) начисления на заработную плату (Отчисления в государственные внебюджетные фонды);
- 2) затраты на расходные материалы;
- 3) эксплуатационные расходы;
- 4) расходы на содержание помещений;
- 5) накладные расходы.

Создание ИС рассматривалось как работа, оплачиваемая по ставке 8-ми часового рабочего дня. В таблице 9 приведены сведения об окладе и стоимости дня работы каждого специалиста.

Таблица 9 - Оплата труда специалистов

Специалист	Оклад	Оплата за 8-ми часовой день
Руководитель	70 000, 00р	3 181,81р
Программист	35 000,00 р.	1 590,91р
Специалист	20 000,00р	909,09р.

На основании данных, представленных в таблицах 8 и 9, могут быть рассчитаны планируемые суммарные затраты на оплату труда специалистов, участвующих в разработке ИС. Соответствующие расчеты представлены в таблице 10.

Таблица 10 - Суммарные планируемые затраты на оплату труда

Специалист	Оплата за 8-ми часовой день	Кол-во рабочих дней	Затраты на оплату труда
Руководитель	3 181,81р.	5,5	17 500,00р.
Программист	1 590,91р.	10,5	16 704,55р.
Специалист	909,09р.	9	8 181,82р.
		Итого:	42 386,36р.

Согласно законодательству, учитывается еще одна статья «Дополнительная заработная плата», это процент от основной (42386 рублей). Норматив определяется на предприятии: 8-14%.

Дополнительная заработная плата равна 10% от основной, то есть 4238 рублей. Итого ФОТ составляет 46624 рубля.

Сумма отчислений может быть рассчитана на основании планируемой суммарной заработной платы специалистов:

$$O = S * r = (42386 + 4238) * 0,3 = 13987 \text{ рублей,} \quad (1)$$

где O итоговая сумма страховых взносов;

S - суммарная заработная плата сотрудников;

r - ставка страховых взносов.

Для выполнения работ по созданию ИС планируется приобрести расходные материалы, состав и стоимость которых представлены в таблице 11.

Таблица 11 - Затраты на приобретение расходных материалов

Наименование	Единица измерения	Кол-во	Цена	Итого
Бумага для принтера	упаковка	2	300,00р.	600,00р.
Картридж для принтера	штука	1	3 500,00р.	3 500,00р.
Канцелярский набор	штука	2	300,00р.	600,00р.
			Итого:	4 700,00р.

В процессе проектирования и разработки ИС использовался один ПК, потребляемая им мощность, тариф оплаты 1 кВт и расчет планируемых затрат на электроэнергию представлены в таблице 12.

Таблица 12 - Расчет прямых затрат на электроэнергию

Компьютер	Мощность, Вт	Кол-во дней	Кол-во часов	Кол-во кВт	Стоимость кВт	К оплате, руб
HP 8000 Elite CMT	500	50	200	100	3,80р.	380,00р.
					Итого:	380,00р.

Величина амортизационных отчислений по i -му основному средству, используемому студентом при работе над ВКР, определяется по формуле:

$$A_{i\text{вкр}} = A_i * (T_{i\text{вкр}} / 12). \quad (2)$$

В данном случае используется только персональный компьютер стоимость 100 тысяч рублей, норма амортизации -1,11 (служба 9 лет) Амортизационные отчисления по нему тогда составляют 22, 5 дня, итого:

$$A_{i\text{вкр}} = 8148 \text{ рублей.} \quad (3)$$

Накладные расходы составляют 20% от ФОТ, итого:

$$46624 * 0,2 = 9324 \text{ рубля.} \quad (4)$$

В таблице 13 представлен прогноз затрат на выполнение разработки и структура затрат.

Таблица 13 - Структура планируемой себестоимости ИС

Статьи затрат	Сумма	% от общей суммы затрат
Основная заработная плата разработчиков	42 386,36р.	51
Дополнительная заработная плата	4238	5
Начисления на заработную плату	13987	17
Затраты на расходные материалы	4 700,00р.	6
Амортизационные отчисления	8148	10
Накладные расходы	9324	11
Итого:	.82783,36 рублей	

На рисунке 3 представлено процентное распределение затрат по статьям в структуре планируемой суммарной себестоимости разработки ИС.

Структура планируемой себестоимости ИС



Рисунок 3- Процентное распределение затрат по статьям в структуре

3 Защита от ионизирующих излучений работников, обслуживающих автоматизированную систему

3.1 Основные характеристики ионизирующих излучений

Ионизирующим излучением называют излучения, взаимодействие которых со средой приводит к образованию электрических зарядов различных знаков. Источники этих излучений широко используются в технике, химии, медицине, сельском хозяйстве и других областях, например, при измерении плотности почв, обнаружении течей в газопроводах, измерении толщины листов, труб и стержней, антистатической обработке тканей, полимеризации пластмасс, радиационной терапии злокачественных опухолей и др. Однако следует помнить, что источники ионизирующего излучения представляют существенную угрозу здоровью и жизни использующих их людей.

Существует два вида ионизирующих излучений:

- корпускулярное, состоящее из частиц с массой покоя, отличной от нуля (альфа- и бета-излучение и нейтронное излучение);
- электромагнитное (гамма-излучение и рентгеновское) с очень малой длиной волны.

Альфа-излучение представляет собой поток ядер гелия, обладающих большой скоростью. Эти ядра имеют массу 4 и заряд +2. Они образуются при радиоактивном распаде ядер или при ядерных реакциях. В настоящее время известно более 120 искусственных и естественных альфа-радиоактивных ядер, которые, испуская альфа-частицу, теряют 2 протона и 2 нейтрона.

Энергия альфа-частиц не превышает нескольких МэВ (мега-электрон-вольт). Излучаемые альфа-частицы движутся практически прямолинейно со скоростью примерно 20000 км/с.

Под длиной пробега частицы в воздухе или других средах принято называть наибольшее расстояние от источника излучения, при котором еще

можно обнаружить частицу до ее поглощения веществом. Длина пробега частицы зависит от заряда, массы, начальной энергии и среды, в которой происходит движение. С возрастанием начальной энергии частицы и уменьшением плотности среды длина пробега увеличивается. Если начальная энергия излучаемых частиц одинакова, то тяжелые частицы обладают меньшими скоростями, чем легкие. Если частицы движутся медленно, то их взаимодействие с атомами вещества среды более эффективно и частицы быстрее растрачивают имеющийся у них запас энергии.

Длина пробега альфа-частиц в воздухе обычно менее 10 см. За счет своей большой массы при взаимодействии с веществом альфа-частицы быстро теряют свою энергию. Это объясняет их низкую проникающую способность и высокую удельную ионизацию: при движении в воздушной среде альфа-частица на 1 см своего пути образует несколько десятков тысяч пар заряженных частиц - ионов.

Бета-излучение представляет собой поток электронов или позитронов, возникающих при радиоактивном распаде. В настоящее время известно около 900 бета - радиоактивных изотопов.

Масса бета - частиц в несколько десятков тысяч раз меньше массы альфа-частиц. В зависимости от природы источника бета - излучений скорость этих частиц может лежать в пределах 0,3 - 0,99 скорости света. Энергия бета-частиц не превышает нескольких МэВ, длина пробега в воздухе составляет приблизительно 1800 см., а в мягких тканях человеческого тела ~ 2,5 см. Проникающая способность бета-частиц выше, чем альфа-частиц (из-за меньших массы и заряда).

Нейтронное излучение представляет собой поток ядерных частиц, не имеющих электрического заряда. Масса нейтрона приблизительно в 4 раза меньше массы альфа-частиц. В зависимости от энергии различают медленные нейтроны (с энергией менее 1 КэВ (кило-электрон-Вольт) = 10^3 эВ), нейтроны промежуточных энергий (от 1 до 500 КэВ) и быстрые нейтроны (от 500 КэВ

до 20 МэВ). При неупругом взаимодействии нейтронов с ядрами атомов среды возникает вторичное излучение, состоящее из заряженных частиц и гамма - квантов (гамма-излучение). При упругих взаимодействиях нейтронов с ядрами может наблюдаться обычная ионизация вещества. Проникающая способность нейтронов зависит от их энергии, но она существенно выше, чем у альфа- или бета-частиц. Нейтронное излучение обладает высокой проникающей способностью и представляет для человека наибольшую опасность из всех видов корпускулярного излучения. Мощность нейтронного потока измеряется плотность потока нейтронов.

Гамма-излучение представляет собой электромагнитное излучение с высокой энергией и с малой длиной волны. Оно испускается при ядерных превращениях или взаимодействии частиц. Высокая энергия (0,01 - 3 МэВ) и малая длина волны обуславливает большую проникающую способность гамма-излучения. Гамма-лучи не отклоняются в электрических и магнитных полях. Это излучение обладает меньшей ионизирующей способностью, чем альфа- и бета-излучение.

Рентгеновское излучение может быть получено в специальных рентгеновских трубах, в ускорителях электронов, в среде, окружающей источник бета - излучения, и др. Рентгеновское излучение представляет собой один из видов электромагнитного излучения. Энергия его обычно не превышает 1 МэВ. Рентгеновское излучение, как и гамма-излучение, обладает малой ионизирующей способностью и большой глубиной проникновения.

Для характеристики воздействия ионизирующего излучения на вещество введено понятие дозы излучения. Дозой излучения - называется часть энергии, переданная излучением веществу и поглощенная им. Количественной характеристикой взаимодействия ионизирующего излучения и вещества, является поглощенная доза излучения (D), равная отношению средней энергии dE , переданной ионизирующим излучением веществу в элементарном объеме, к массе облученного вещества в этом объеме dm :

$$D = dE / dm. \quad (5)$$

До недавнего времени за количественную характеристику только рентгеновского и гамма-излучения, основанную на их ионизирующем действии, принималась экспозиционная доза X - отношение полного электрического заряда dQ ионов одного знака, возникающих в малом объеме сухого воздуха, к массе воздуха dm в этом объеме, т.е.:

$$X = dQ / dm. \quad (6)$$

Для оценки возможного ущерба здоровью при хроническом воздействии ионизирующего излучения произвольного состава введено понятие эквивалентной дозы (H). Эта величина определяется как произведение поглощенной дозы D на средний коэффициент качества излучения Q (безразмерный) в данной точке ткани человеческого тела, т.е.:

$$H = D \cdot Q. \quad (7)$$

Существует еще одна характеристика ионизирующего излучения - мощность дозы X (соответственно поглощенной, экспозиционной или эквивалентной) представляющая собой приращение дозы за малый промежуток времени dx , деленное на этот промежуток dt . Так, мощность экспозиционной дозы (x или w , Кл / кг · с) составит:

$$X = W = dx / dt. \quad (8)$$

Биологическое действие рассмотренных излучений на организм человека различно.

Альфа-частицы, проходя через вещество и сталкиваясь с атомами, ионизируют (заряжают) их, выбивая электроны. В редких случаях эти частицы поглощаются ядрами атомов, переводя их в состояние с большей энергией. Эта избыточная энергия способствует протеканию различных химических реакций, которые без облучения не идут или идут очень медленно. Альфа-излучение производит сильное действие на органические вещества, из которых состоит человеческий организм (жиры, белки и углеводы). На слизистых оболочках это излучение вызывает ожоги и другие воспалительные

процессы.

Под действием бета - излучений происходит радиолиз (разложение) воды, содержащейся в биологических тканях, с образованием водорода, кислорода, пероксида водорода H_2O_2 , заряженных частиц (ионов) $OH\cdot$ и $HO_2\cdot$. Продукты разложения воды обладают окислительными свойствами и вызывают разрушение многих органических веществ, из которых состоят ткани человеческого организма.

Действие гамма и рентгеновского излучений на биологические ткани обусловлено в основном образующимися свободными электронами. Нейтроны, проходя через вещество, производят в нем наиболее сильные изменения по сравнению с другими ионизирующими излучениями.

Таким образом, биологическое действие ионизирующих излучений сводится к изменению структуры или разрушению различных органических веществ (молекул), из которых состоит организм человека. Это приводит к нарушению биохимических процессов, протекающих в клетках, или даже к их гибели, в результате чего происходит поражение организма в целом.

Различают внешнее и внутреннее облучение организма. Под внешним облучением понимают воздействие на организм ионизирующих излучений от внешних по отношению к нему источников. Внутреннее облучение осуществляется радиоактивными веществами, попавшими внутрь организма через дыхательные органы, желудочно-кишечный тракт или через кожные покровы. Источники внешнего излучения - космические лучи, естественные радиоактивные источники, находящиеся в атмосфере, воде, почве, продуктах питания и др., источники альфа, бета, гамма, рентгеновского и нейтронного излучений, используемые в технике и медицине, ускорители заряженных частиц, ядерные реакторы (в том числе и аварии на ядерных реакторах) и ряд других.

Радиоактивные вещества, вызывающие внутреннее облучение организма, попадают в него при приеме пищи, курении, питье загрязненной

воды. Поступление радиоактивных веществ в человеческий организм через кожу происходит в редких случаях (если кожа имеет повреждения или открытые раны). Внутреннее облучение организма длится до тех пор, пока радиоактивное вещество не распадется или не будет выведено из организма в результате процессов физиологического обмена. Внутреннее облучение опасно тем, что вызывает длительно незаживающие язвы различных органов и злокачественные опухоли.

При работе с радиоактивными веществами значительному облучению подвергаются руки операторов. Под действием ионизирующих излучений развивается хроническое или острое (лучевой ожог) поражение кожи рук. Хроническое поражение характеризуется сухостью кожи, появлением на ней трещин, изъязвлением и другими симптомами. При остром поражении кистей рук возникают отеки, омертвление тканей, язвы, на месте образования которых возможно развитие злокачественных опухолей.

Под влиянием ионизирующих излучений у человека возникает лучевая болезнь. Различают три степени ее: первая (легкая), вторая и третья (тяжелая).

Симптомами лучевой болезни первой степени являются слабость, головные боли, нарушение сна и аппетита, которые усиливаются на второй стадии заболевания, но к ним дополнительно присоединяются нарушения в деятельности сердечно-сосудистой системы, изменяется обмен веществ и состав крови, происходит расстройство пищеварительных органов. На третьей стадии болезни наблюдаются кровоизлияния выпадение волос, нарушается деятельность центральной нервной системы и половых желез. У людей, перенесших лучевую болезнь, повышается вероятность развития злокачественных опухолей и заболеваний кроветворных органов. Лучевая болезнь в острой (тяжелой) форме развивается в результате облучения организма большими дозами ионизирующих излучений за короткий промежуток времени. Опасно воздействие на организм человека и малых доз радиации, так как при этом могут произойти нарушение наследственной

информации человеческого организма, возникнуть мутации.

Низкий уровень развития легкой формы лучевой болезни возникает при эквивалентной дозе облучения приблизительно 1 Зв, тяжелая форма лучевой болезни, при которой погибает половина всех облученных, наступает при эквивалентной дозе облучения 4,5 Зв. 100%-ный смертельный исход лучевой болезни соответствует эквивалентной дозе облучения 5,5-7,0 Зв.

В настоящее время разработан ряд химических препаратов (протекторов), существенно снижающих негативный эффект воздействия ионизирующего излучения на организм человека.

В России предельно допустимые уровни ионизирующего облучения и принципы радиационной безопасности регламентируются «Нормами радиационной безопасности» НРБ-99/2009. В соответствии с этими нормативными документами нормы облучения установлены для следующих трех категорий лиц:

- категория А - персонал, постоянно или временно работающий с источниками ионизирующих излучений;
- категория Б - ограниченная часть населения, которая по условиям размещения рабочих мест или по условиям проживания может подвергаться воздействию источников излучения;
- категория В - население страны, республики, края и области.

Для лиц категории А основным дозовым пределом является индивидуальная эквивалентная доза внешнего и внутреннего излучения за год (Зв/год) в зависимости от радиочувствительности органов (критические органы). Это предельно допустимая доза (ПДД) - наибольшее значение индивидуальной эквивалентной дозы за год, которое при равномерном воздействии в течение 50 лет не вызовет в состоянии здоровья персонала неблагоприятных изменений, обнаруживаемых современными методами.

Для персонала категории А индивидуальная эквивалентная доза (H , Зв), накопленная в критическом органе за время T (лет) с начала

профессиональной работы, не должна превышать значения, определяемого по формуле:

$$H = \text{ПДД} \cdot T. \quad (9)$$

Кроме того, доза, накопленная к 30 годам, не должна превышать 12 ПДД.

Для категории Б установлен предел дозы за год (ПД, Зв/год), под которым понимают наибольшее среднее значение индивидуальной эквивалентной дозы за календарный год у критической группы лиц, при котором равномерное облучение в течении 70 лет не может вызвать в состоянии здоровья неблагоприятных изменений, обнаруживаемых современными методами. В таблице 14 приведены основные дозовые пределы внешнего и внутреннего облучений в зависимости от радиочувствительности органов.

Таблица 14 - Основные значения дозовых пределов внешнего и внутреннего облучений

Группа критических органов	Органы и ткани человеческого организма	ПДД для категории А, Зв/год	ПДД для категории Б, Зв/год
1	Все тело, гонады (половые органы), красный костный мозг	0,05	0,005
2	Любой отдельный орган, кроме гонад, красного костного мозга, костной ткани, щитовидной железы, кожи, кистей, предплечий, лодыжек и стоп	0,15	0,015
3	Костная ткань, щитовидная железа, кожный покров, кисти, предплечья, лодыжки и стопы	0,30	0,03

3.2 Нормы радиационной безопасности

Нормы радиационной безопасности (НРБ-99/2009) применяются для обеспечения безопасности человека в условиях воздействия на него ионизирующего излучения искусственного или природного происхождения.

Требования и нормативы, установленные НРБ-99/2009, являются обязательными для всех юридических лиц независимо от их подчиненности и формы собственности, в результате деятельности которых возможно облучение людей, а также для местных распорядительных и исполнительных органов, граждан РФ, иностранных граждан и лиц без гражданства, проживающих на территории РФ.

Нормы радиационной безопасности относятся только к ионизирующему излучению. В них учтено, что ионизирующее излучение является одним из множества источников риска для здоровья человека и что риски, связанные с воздействием излучения, не должны соотноситься только с выгодами от его использования. Их следует сопоставлять с рисками нерадиационного происхождения.

Нормы распространяются на следующие виды воздействия ионизирующего излучения на человека:

- в условиях нормальной эксплуатации техногенных источников излучения;
- в результате радиационной аварии;
- от природных источников излучения;
- при медицинском облучении.

Требования по обеспечению радиационной безопасности сформулированы для каждого вида облучения. Суммарная доза от всех видов облучения используется для оценки радиационной обстановки и ожидаемых медицинских последствий, а также для обоснования защитных мероприятий и оценки их эффективности.

Требования НРБ-99/2009 не распространяются на источники излучения, создающие при любых условиях обращения с ними:

- индивидуальную годовую эффективную дозу не более 10 мкЗв;
- индивидуальную годовую эквивалентную дозу в коже не более 50 мЗв и в хрусталике не более 15 мЗв;

- коллективную годовую эффективную дозу не более 1 чел. Зв (либо при коллективной дозе более 1 чел. Зв когда оценка по принципу оптимизации показывает нецелесообразность ее снижения).

Требования НРБ-99/2009 не распространяются также на космическое излучение на поверхности Земли и внутреннее облучение человека, создаваемое природным калием, на которые практически невозможно влиять.

Перечень и порядок освобождения источников ионизирующего излучения от радиационного контроля устанавливаются санитарными правилами.

Для категорий облучаемых лиц устанавливаются три класса нормативов:

- основные пределы доз (ПД);

- допустимые уровни монофакторного воздействия (для одного радионуклида, пути поступления или одного вида внешнего облучения), являющиеся производными от основных пределов доз: пределы годового поступления (ПГП), допустимые среднегодовые объемные активности (ДОВА), среднегодовые удельные активности (ДУА) и др.;

- контрольные уровни (дозы, уровни, активности, плотности потоков и др.). Их значения должны учитывать достигнутый в организации уровень радиационной безопасности и обеспечивать условия, при которых радиационное воздействие будет ниже допустимого.

Ионизирующая реакция при воздействии на организм человека может вызвать два вида эффектов, которые клинической медициной относятся к болезням: детерминированные пороговые эффекты (лучевая болезнь, лучевой дерматит, лучевая катаракта, лучевое бесплодие, аномалии в развитии плода и др.) и стохастические (вероятностные) беспороговые эффекты (злокачественные опухоли, лейкозы, наследственные болезни).

Для обеспечения радиационной безопасности при нормальной эксплуатации источников излучения необходимо руководствоваться принципами:

					<i>ПГУ 1.090501.02.001 ПЗ</i>	Лист
						91
Изм.	Лист	№ докум.	Подпись	Дата		

- не превышение допустимых пределов индивидуальных доз облучения человека от всех источников излучения (принцип нормирования);

- запрещение всех видов деятельности по использованию источников излучения, при которых полученная для человека и общества польза не превышает риск возможного вреда, причиненного дополнительным облучением (принцип обоснования);

- поддержание на возможно низком и достижимом уровне с учетом экономических и социальных факторов индивидуальных доз облучения и числа облучаемых лиц при использовании любого источника излучения (принцип оптимизации).

К защитным мероприятиям при использовании закрытых источников ионизирующего излучения относятся:

- уменьшение мощности источников до минимальных величин;
- сокращение времени работы с источниками;
- увеличение расстояния от источника до работающего;
- экранирование источников излучения;
- использование индивидуальных средств защиты, применяемых при работе с такими источниками;
- санитарная обработка обслуживающего персонала;
- личная гигиена.

3.3 Защита от действия ионизирующих излучений

Основные принципы радиационной безопасности заключаются в не превышении установленного основного дозового предела, исключении всякого необоснованного облучения и снижении дозы излучения до возможно низкого уровня.

Для определения индивидуальных доз облучения персонала необходимо систематически проводить радиационный (дозиметрический) контроль, объем которого зависит от характера работы с радиоактивными веществами.

					<i>ПГУ 1.090501.02.001 ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подпись	Дата		92

Каждому оператору, имеющему контракт с источниками ионизирующего излучения, выдается индивидуальный дозиметр для контроля полученной дозы гамма-излучений. В помещениях, где проводится работа с радиоактивными веществами, необходимо обеспечить и общий контроль за интенсивностью различных видов излучений. Эти помещения должны быть изолированы от прочих помещений, оснащены системой приточно-вытяжной вентиляции с кратностью воздухообмена не менее 5. Окраска стен, потолка и дверей в этих помещениях, а также устройство пола выполняются таким образом, чтобы исключить накопление радиоактивной пыли и избежать поглощения радиоактивных аэрозолей, паров и жидкостей отделочными материалами (окраска стен, дверей и в некоторых случаях потолков должна производиться масляными красками, полы покрываются материалами, не впитывающими жидкости, - линолеум, полихлорвиниловым пластиком и др.). Все строительные конструкции в помещениях, где проводится работа с радиоактивными веществами, не должны иметь трещин и несплошностей; углы закругляют для того, чтобы не допустить скопления в них радиоактивной пыли и облегчить уборку. Не менее 1 раза в месяц проводят генеральную уборку помещений с обязательным мытьем горячей мыльной водой стен, окон, дверей, мебели и оборудования. Текущая влажная уборка помещений проводится ежедневно.

Для уменьшения облучения персонала все работы с этими источниками проводят с использованием длинных захватов или держателей. Защита временем заключается в том, что в работу с радиоактивными источниками проводят за такой период времени, чтобы доза облучения, полученная персоналом, не превышала предельно допустимого уровня.

Коллективные средства защиты от ионизирующих излучений регламентируются ГОСТ 12.4.120-83 «Средства коллективной защиты от ионизирующих излучений. Общие требования». В соответствии с этим нормативным документом основными средствами защиты являются

					<i>ПГУ 1.090501.02.001 ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		<i>93</i>

стационарные и передвижные защитные экраны, контейнеры для транспортирования и хранения источников ионизирующих излучений, а также для сбора и транспортировки радиоактивных отходов, защитные сейфы и боксы и др.

Стационарные и передвижные защитные экраны предназначены для снижения уровня излучения на рабочем месте до допустимой величины. Если работу с источниками ионизирующих излучений проводят в специальном помещении - рабочей камере, то экранами служат ее стены, пол и потолок, изготовленные из защитных материалов. Также экраны носят название стационарных. Для устройства передвижных экранов используют различные щиты, поглощающие или ослабляющие излучение.

Экраны изготавливают из различных материалов. Их толщина зависит от вида ионизирующего излучения, свойств защитного материала и необходимой кратности ослабления излучения k . Величина k показывает, во сколько раз необходимо понизить энергетические показатели излучения (мощность экспозиционной дозы, поглощенную дозу, плотность потока частиц и др.), чтобы получить допустимые значения перечисленных характеристик. Например, для случая поглощенной дозы k выражается следующим образом:

$$k = D / D_0, \quad (10)$$

где D - мощность поглощенной дозы;

D_0 - допустимый уровень поглощенной дозы.

Для сооружения стационарных средств защиты стен, перекрытий, потолков и т.д. используют кирпич, бетон, баритобетон и баритовую штукатурку (в их состав входит сульфат бария - $BaSO_4$). Эти материалы надежно защищают персонал от воздействия гамма- и рентгеновского излучения.

Для создания передвижных экранов используют различные материалы. Защита от альфа-излучения достигается применением экранов из обычного

или органического стекла толщиной несколько миллиметров. Достаточной защитой от этого вида излучения является слой воздуха в несколько сантиметров. Для защиты от бета-излучения экраны изготавливают из алюминия или пластмассы (органическое стекло). От гамма- и рентгеновского излучения эффективно защищают свинец, сталь, вольфрамовые сплавы. Смотровые системы изготавливают из специальных прозрачных материалов, например, свинцового стекла. От нейтронного излучения защищают материалы, содержащие в составе водород (вода, парафин), а также бериллий, графит, соединения бора и т.д. Бетон также можно использовать для защиты от нейтронов.

Защитные сейфы применяются для хранения источников гамма-излучения. Они изготавливаются из свинца и стали.

Для работы с радиоактивными веществами, обладающими альфа- и бета-активностью, используют защитные перчаточные боксы.

Защитные контейнеры и сборники для радиоактивных отходов изготавливаются из тех же материалов, что и экраны - органического стекла, стали, свинца и др.

При проведении работ с источниками ионизирующих излучений опасная зона должна быть ограничена предупреждающими надписями.

Принцип действия приборов, предназначенных для контроля за персоналом, который подвергается воздействию ионизирующих излучений, основан на различных эффектах, возникающих при взаимодействии этих излучений с веществом. Основные методы обнаружения и измерения радиоактивности - ионизация газа, сцинтилляционные и фотохимические методы. Наиболее часто используется ионизационный метод, основанный на измерении степени ионизации среды, через которую прошло излучение.

Сцинтилляционные методы регистрации излучений основаны на способности некоторых материалов, поглощая энергию ионизирующего излучения, превращать ее в световое излучение. Примером такого материала

может служить сульфит цинка (ZnS). Сцинтилляционный счетчик представляет собой фотоэлектронную трубку с окошком, покрытым сульфидом цинка. При попадании внутрь этой трубки излучения возникает слабая вспышка света, которая приводит к возникновению в фотоэлектронной трубке импульсов электрического тока. Эти импульсы усиливаются и подсчитываются.

Фотохимические методы, или методы автордиографии, основаны на воздействии радиоактивного образца на слой фотоэмульсии, содержащий галогениды серебра. Уровень радиоактивности образца оценивают после проявления пленки.

Существуют и другие методы определения ионизирующих излучений, например, калориметрические, которые основаны на измерении количества тепла, выделяющегося при взаимодействии излучения с поглощающим веществом.

Приборы дозиметрического контроля делятся на две группы: дозиметры, используемые для количественного измерения мощности дозы, и радиометры или индикаторы излучения, применяемые для быстрого обнаружения радиоактивных загрязнений.

Из отечественных приборов применяются, например, дозиметры марок ДРГЗ-04 и ДКС-04. Первый используется для измерения гамма- и рентгеновского излучения в диапазоне энергий 0,03-3,0 МэВ. Шкала прибора проградуирована в микрорентген/секунду (мкР/с). Второй прибор используется для измерения гамма- и бета-излучения в энергетическом диапазоне 0,5-3,0 МэВ, а также нейтронного излучения (жесткие и тепловые нейтроны). Шкала прибора проградуирована в миллирентгенах в час (мР/ч). Промышленность выпускает также бытовые дозиметры, предназначенные для населения, например, бытовой дозиметр «Мастер-1» (предназначен для измерения дозы гамма-излучения), дозиметр-радиометр бытовой АНРИ-01 («Сосна»).

К средствам индивидуальной защиты от ионизирующих излучений относится спецодежда - халаты, комбинезоны, полукombineзоны и шапочки, изготовленные из хлопчатобумажной ткани. При значительном загрязнении производственного помещения радиоактивными веществами на спецодежду из ткани дополнительно надевают пленочную одежду (нарукавники, брюки, фартук, халат и т.д.), изготовленную из пластика. Для защиты рук следует использовать просвинцованные резиновые перчатки.

В тех случаях, когда приходится работать в условиях значительного радиационного загрязнения, для защиты персонала используют пневмокостюмы (скафандры) из пластмассовых материалов с поддувом по гибким шлангам воздуха или снабженные кислородным аппаратом. Для поддержания нормальных температурных условий в скафандре расход воздуха должен составлять 150-200 л/мин.

Для защиты органов зрения от излучения применяют очки со стеклами, содержащими специальные добавки (фосфат вольфрама или свинец), а при работе с источниками альфа- и бета-излучений глаза защищают щитками из органического стекла.

В данном разделе дипломной работы была изложена информация о защите от ионизирующих излучений работников, обслуживающих автоматизированную систему. На основании изученной литературы по данной проблеме были рассмотрены основные характеристики ионизирующих излучений, нормы радиационной безопасности и защита от действия ионизирующих излучений.

Заключение

Использование современных информационных технологий ориентированных на выполнение процессов управления компанией значительно повышает уровень её конкурентоспособности. В целом, автоматизация даёт возможность:

- снизить расходы на выполнение целого ряда повседневных задач, выполняемых профильными специалистами компании;
- значительно повысить уровень эффективности управления деятельностью производства и в целом всех бизнес процессов компании;
- повысить качество выполнения сотрудниками своих повседневных обязанностей и снизить рутинность их работы;
- обеспечить эффективный и максимально точный учёт движения всех материальных средств компании;
- позволить руководителям как среднего, так и нижнего звеньев максимально оперативном точно осуществлять анализ деятельности всех подконтрольных им подразделений компании;
- увеличить общую эффективность обмена данными и всей сопутствующей информации;
- обеспечивать безопасность всех хранимых данных.

Хочу обратить ваше внимание непосредственно на тот факт, что создание и внедрение автоматизированной информационной подсистемы даст уникальную по своей эффективности и возможностям методику экономии времени на рабочем месте, а также повышению результативности труда профильных специалистов компании. В целом, автоматизированные рабочего места путём внедрения информационной системы позволяет значительно снизить временные затраты на осуществление рутинной бумажной работы, а также заполнения форм и поиск документов.

Все указанные процедуры информационная система будет осуществлять автоматически, путём всестороннего анализа информации, которая имеется в

					ПГУ 1.090501.02.001 ПЗ	Лист
						98
Изм.	Лист	№ докум.	Подпись	Дата		

наличии непосредственно в базе данных. Также непосредственное использование информационной подсистемы даст уникальную возможность более глубоко и в максимальном объеме осуществлять сбор и анализ нужных данных.

В последнее время все чаще в повседневную работу специалистов компании внедряются новые инновационные технологии. Из них отдельного упоминания заслуживают такие технологии как:

- автоматическое заполнение всей необходимой для работы компании документации;
- исключение недостатков присущих всем бумажным носителям;
- совместный авторизованный доступ непосредственно к справочной информации, касающейся персональных данных всех клиентов компании;
- автоматизация рутинных операций;
- доступ к информационно-справочным ресурсам;
- регулярно осуществляемый контроль качества;
- стандартизация учета.

Для быстрой и полной адаптации пользователя к системе был разработан удобный дружественный интерфейс пользователя и подробное описание работы с системой в руководстве пользователя. Вполне обосновано полагаю, что разработанная в моей дипломной работе подсистема на все 100% по всем параметрам соответствует всем информационным требованиям рассматриваемой в данной работе компании, и гарантированно сможет эффективно поддерживать всесторонне данное, обозначенное выше, соответствие на протяжении всего цикла жизнедеятельности системы.

При этом, всесторонне проведенный подробный экономический анализ наглядно продемонстрировал, что после внедрения разработанной информационной системы по управлению компанией, оказывающей услуги по транспортировке различных грузов привело к весомому росту экономических показателей производственного предприятия, а также, в конечном итоге, все

это в полной мере способствовало быстрой окупаемости информационной системы в целом

В результате проведенной работы ясно, что цель дипломного проекта — разработка корпоративного мессенджера, выполнена.

В ходе сравнения и анализа, выбраны язык программирования и система управления базами данных, описана структура мессенджера, в том числе база данных, диаграмма классов, приведены скриншоты программы, рассчитана стоимость ее разработки.

					<i>ПГУ 1.090501.02.001 ПЗ</i>	<i>Лист</i>
						<i>100</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		

Список использованных источников

1. Денис Колисниченко: PHP и MySQL. Разработка веб-приложений, - Москва, Издательство: BHV, 2015 г., 592 стр.
2. Каллум Хопкинс: PHP. Быстрый старт, - Москва, Издательство: Эксмо-Пресс, 2014 г., 160 стр.
3. Кевин Янк: PHP и MySQL. От новичка к профессионалу, - Москва, Издательство: Эксмо, 2013 г., 384 стр.
4. А. Тюгашев: Языки программирования. Учебное пособие. Стандарт третьего поколения, - Москва, Издательство: Питер, 2014 г. 336 стр.
5. Александр Жадаев: PHP для начинающих, - Москва, Издательство: Питер, 2014 г., 288 стр.
6. Бретт Маклафлин: PHP и MySQL. Исчерпывающее руководство, - Москва, Издательство: Питер, 2014 г., 544 стр.
7. Вадим Дунаев: HTML, скрипты и стили, - Москва, Издательство: BHV, 2015 г., 824 стр.

Приложение А

(обязательное)

Корпоративный мессенджер для приема и передачи текстовых
сообщений

Листинг программных модулей

					ПГУ 1.090501.02.001 ПЗ	Лист
						102
Изм.	Лист	№ докум.	Подпись	Дата		

```

    }

    /**
     * Метод проверки "Имеет ли пользователь разрешение на
    просмотр текущего документа"
     * @param user Пользователь
     * @return Результат проверки (да, нет)
     */
    public Boolean testViewAccess(User user) {

        if (user == null) {
            return false;
        }
        if (locked == false) {
            return true;
        }
        if (idUser.getId().equals(user.getId())) {
            return true;
        }
        if (user.getIdUserType().getId().equals(1)) {
            return true;
        }

        Collection<DocumentRule> rules =
documentRuleCollection;
        for (DocumentRule rule : rules) {
            if ((rule.getIdUser() != null &&
rule.getIdUser().getId().equals(user.getId()) &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()) && rule.getIdUserType().getId().equals(1)) && rule.getIdUserType().getId().equals(1)) {
                return true;
            }
        }
        return false;
    }

    /**
     * Метод проверки "Имеет ли пользователь разрешение на
    просмотр текущего документа" (возвращает исключение)
     * @param user Пользователь
     * @throws Exception Результат проверки
     */
    public void controlViewAccess(User user) throws Exception
    {
        if (testViewAccess(user) == false) {
            throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getString("AccessDenied"));
        }
    }

```

```

    }
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
удаление текущего документа"
 * @param user Пользователь
 * @return Результат проверки
 */
public Boolean testDeleteAccess(User user) {

    if (user == null) {
        return false;
    }

    if (locked == false) {
        return true;
    }
    if (idUser.getId().equals(user.getId())) {
        return true;
    }
    if (user.getIdUserType().getId().equals(1)) {
        return true;
    }
    Collection<DocumentRule> rules =
getDocumentRuleCollection();
    for (DocumentRule rule : rules) {
        if (rule.getIdUser() != null) {
            if
(rule.getIdUser().getId().equals(user.getId())
rule.getIdAccessDelete() == true) {
                return true;
            }
        }
        if (rule.getIdUserType() != null) {
            if
(rule.getIdUserType().getId().equals(user.getIdUserType().getId(
)) && rule.getIdAccessDelete() == true) {
                return true;
            }
        }
    }
    return false;
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
удаление текущего документа" (возвращает исключение)
 * @param user Пользователь
 * @throws Exception Результат проверки
 */
public void controlDeleteAccess(User user) throws

```

```

Exception {
    if (testDeleteAccess(user) == false) {
        throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
    }
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
добавление документа"
 * @param user Пользователь
 * @return Результат проверки
 */
public Boolean testInsetAccess(User user) {

    if (user == null) {
        return false;
    }

    if (locked == false) {
        return true;
    }
    if (idUser.getId().equals(user.getId())) {
        return true;
    }
    if (user.getIdUserType().getId().equals(1)) {
        return true;
    }
    Collection<DocumentRule> rules =
getDocumentRuleCollection();
    for (DocumentRule rule : rules) {
        if ((rule.getIdUser() != null &&
rule.getIdUser().getId().equals(user.getId()) &&
rule.getAccessInsert())
|| (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId())
) && rule.getAccessInsert())) {
            return true;
        }
    }
    return false;
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
добавление документа" (возвращает исключение)
 * @param user Пользователь
 * @throws Exception Результат проверки
 */
public void controlInsertAccess(User user) throws
Exception {

```

```

        if (testInsetAccess(user) == false) {
            throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
        }
    }

    /**
     * Метод проверки "Имеет ли пользователь разрешение на
    обновление документа"
     * @param user Пользователь
     * @return Результат проверки
     */
    public Boolean testUpdateAccess(User user) {

        if (user == null) {
            return false;
        }

        if (locked == false) {
            return true;
        }
        if (idUser.getId().equals(user.getId())) {
            return true;
        }
        if (user.getIdUserType().getId().equals(1)) {
            return true;
        }
        Collection<DocumentRule> rules =
getDocumentRuleCollection();
        for (DocumentRule rule : rules) {
            if ((rule.getIdUser() != null &&
rule.getIdUser().getId().equals(user.getId())
rule.getIdAccessUpdate())
                || (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()
) && rule.getIdAccessUpdate())) {
                return true;
            }
            if (locked == false) {
                return true;
            }
            if (idUser.getId().equals(user.getId())) {
                return true;
            }
            if (user.getIdUserType().getId().equals(1)) {
                return true;
            }
        }

        Collection<DocumentRule> rules =
documentRuleCollection;
        for (DocumentRule rule : rules) {
            if ((rule.getIdUser() != null &&

```

```

rule.getIdUser().getId().equals(user.getId())                                &&
rule.getAccessView()
                                || (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()
) && rule.getAccessView())) {
    return true;
}
}
return false;
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
просмотр текущего документа" (возвращает исключение)
 * @param user Пользователь
 * @throws Exception Результат проверки
 */
public void controlViewAccess(User user) throws Exception
{
    if (testViewAccess(user) == false) {
        throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
    }
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
удаление текущего документа"
 * @param user Пользователь
 * @return Результат проверки
 */
public Boolean testDeleteAccess(User user) {

    if (user == null) {
        return false;
    }

    if (locked == false) {
        return true;
    }
    if (idUser.getId().equals(user.getId())) {
        return true;
    }
    if (user.getIdUserType().getId().equals(1)) {
        return true;
    }
    Collection<DocumentRule> rules =
getDocumentRuleCollection();
    for (DocumentRule rule : rules) {
        if (rule.getIdUser() != null) {
            if

```

```

(rule.getIdUser().getId().equals(user.getId()))
rule.getAccessDelete() == true) {
    return true;
}
}
if (rule.getIdUserType() != null) {
    if
(rule.getIdUserType().getId().equals(user.getIdUserType().getId(
)) && rule.getAccessDelete() == true) {
        return true;
    }
}
}
return false;
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
удаление текущего документа" (возвращает исключение)
 * @param user Пользователь
 * @throws Exception Результат проверки
 */
public void controlDeleteAccess(User user) throws
Exception {
    if (testDeleteAccess(user) == false) {
        throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
    }
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
добавление документа"
 * @param user Пользователь
 * @return Результат проверки
 */
public Boolean testInsetAccess(User user) {

    if (user == null) {
        return false;
    }

    if (locked == false) {
        return true;
    }
    if (idUser.getId().equals(user.getId())) {
        return true;
    }
    if (user.getIdUserType().getId().equals(1)) {
        return true;
    }
}

```

```

        Collection<DocumentRule>          rules          =
getDocumentRuleCollection();
        for (DocumentRule rule : rules) {
            if      ((rule.getIdUser()      !=      null      &&
rule.getIdUser().getId().equals(user.getId())      &&
rule.getAccessInsert())
                    || (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()
) && rule.getAccessInsert())) {
                return true;
            }
        }
        return false;
    }

    /**
     * Метод проверки "Имеет ли пользователь разрешение на
добавление документа" (возвращает исключение)
     * @param user Пользователь
     * @throws Exception Результат проверки
     */
    public void controlInsertAccess(User user) throws
Exception {
        if (testInsetAccess(user) == false) {
            throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
        }
    }

    /**
     * Метод проверки "Имеет ли пользователь разрешение на
обновление документа"
     * @param user Пользователь
     * @return Результат проверки
     */
    public Boolean testUpdateAccess(User user) {

        if (user == null) {
            return false;
        }

        if (locked == false) {
            return true;
        }
        if (idUser.getId().equals(user.getId())) {
            return true;
        }
        if (user.getIdUserType().getId().equals(1)) {
            return true;
        }
    }
    Collection<DocumentRule>          rules          =

```

```

getDocumentRuleCollection();
    for (DocumentRule rule : rules) {
        if ((rule.getIdUser() != null &&
rule.getIdUser().getId().equals(user.getId()) &&
rule.getAccessUpdate())
            || (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()
) && rule.getAccessUpdate())) {
            return true;

        if (locked == false) {
            return true;
        }
        if (idUser.getId().equals(user.getId())) {
            return true;
        }
        if (user.getIdUserType().getId().equals(1)) {
            return true;
        }

        Collection<DocumentRule> rules =
documentRuleCollection;
        for (DocumentRule rule : rules) {
            if ((rule.getIdUser() != null &&
rule.getIdUser().getId().equals(user.getId()) &&
rule.getAccessView())
                || (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()
) && rule.getAccessView())) {
                return true;
            }
        }
        return false;
    }

/**
 * Метод проверки "Имеет ли пользователь разрешение на
просмотр текущего документа" (возвращает исключение)
 * @param user Пользователь
 * @throws Exception Результат проверки
 */
public void controlViewAccess(User user) throws Exception
{
    if (testViewAccess(user) == false) {
        throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
    }
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на

```

```

удаление текущего документа"
    * @param user Пользователь
    * @return Результат проверки
    */
    public Boolean testDeleteAccess(User user) {

        if (user == null) {
            return false;
        }

        if (locked == false) {
            return true;
        }
        if (idUser.getId().equals(user.getId())) {
            return true;
        }
        if (user.getIdUserType().getId().equals(1)) {
            return true;
        }
        Collection<DocumentRule> rules =
        getDocumentRuleCollection();
        for (DocumentRule rule : rules) {
            if (rule.getIdUser() != null) {
                if
                (rule.getIdUser().getId().equals(user.getId())
                rule.getIdAccessDelete() == true) {
                    return true;
                }
            }
            if (rule.getIdUserType() != null) {
                if
                (rule.getIdUserType().getId().equals(user.getIdUserType().getId(
                )) && rule.getIdAccessDelete() == true) {
                    return true;
                }
            }
        }
        return false;
    }

    /**
    * Метод проверки "Имеет ли пользователь разрешение на
    удаление текущего документа" (возвращает исключение)
    * @param user Пользователь
    * @throws Exception Результат проверки
    */
    public void controlDeleteAccess(User user) throws
    Exception {
        if (testDeleteAccess(user) == false) {
            throw new
            Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
            g("AccessDenied"));
        }
    }

```

```

    }
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
 * добавление документа"
 * @param user Пользователь
 * @return Результат проверки
 */
public Boolean testInsetAccess(User user) {

    if (user == null) {
        return false;
    }

    if (locked == false) {
        return true;
    }
    if (idUser.getId().equals(user.getId())) {
        return true;
    }
    if (user.getIdUserType().getId().equals(1)) {
        return true;
    }
    Collection<DocumentRule> rules =
getDocumentRuleCollection();
    for (DocumentRule rule : rules) {
        if ((rule.getIdUser() != null &&
rule.getIdUser().getId().equals(user.getId()) &&
rule.getAccessInsert()
|| (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()
) && rule.getAccessInsert())) {
            return true;
        }
    }
    return false;
}

/**
 * Метод проверки "Имеет ли пользователь разрешение на
 * добавление документа" (возвращает исключение)
 * @param user Пользователь
 * @throws Exception Результат проверки
 */
public void controlInsertAccess(User user) throws
Exception {
    if (testInsetAccess(user) == false) {
        throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
    }
}

```

```

    }

    /**
     * Метод проверки "Имеет ли пользователь разрешение на
     * обновление документа"
     * @param user Пользователь
     * @return Результат проверки
     */
    public Boolean testUpdateAccess(User user) {

        if (user == null) {
            return false;
        }

        if (locked == false) {
            return true;
        }
        if (idUser.getId().equals(user.getId())) {
            return true;
        }
        if (user.getIdUserType().getId().equals(1)) {
            return true;
        }
        Collection<DocumentRule> rules =
getDocumentRuleCollection();
        for (DocumentRule rule : rules) {
            if ((rule.getIdUser() != null &&
rule.getIdUser().getId().equals(user.getId()) &&
rule.getAccessUpdate()
|| (rule.getIdUserType() != null &&
rule.getIdUserType().getId().equals(user.getIdUserType().getId()
) && rule.getAccessUpdate())) {
                return true;
            }
        }
        return false;
    }

    /**
     * Метод проверки "Имеет ли пользователь разрешение на
     * обновление документа" (возвращает исключение)
     * @param user Пользователь
     * @throws Exception Результат проверки
     */
    public void controlUpdateAccess(User user) throws
Exception {
        if (testUpdateAccess(user) == false) {
            throw new
Exception(ResourceBundle.getBundle("/resources/Bundle").getStrin
g("AccessDenied"));
        }
    }
}

```

}
/*

					<i>ПГУ 1.090501.02.001 ПЗ</i>	<i>Лист</i>
						<i>114</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		

Приложение Б

(обязательное)

Корпоративный мессенджер для приема и передачи текстовых
сообщений

Экранные формы

					ПГУ 1.090501.02.001 ПЗ	Лист
						115
Изм.	Лист	№ докум.	Подпись	Дата		

Вход

Пожалуйста, заполните следующие поля для входа:

Логин

Пароль

☒ Запомнить меня

Если вы забыли пароль, то можно его [восстановить](#).

[Я ещё не зарегистрирован](#)

Вход

Рисунок Б.1 - Вход в систему

Корпоративный мессенджер

Проекты

Сообщения

Панель администратора ▾

Выход (admin)

[Главная](#) / [Проекты](#)

Проекты

Создать проект

Показаны записи 1-8 из 8.

#	Название	Статус проекта	
	<input type="text"/>	<input type="text"/>	
1	Проект 1	Открыт	  
2	Проект 2	Открыт	  
3	Проект 3	Завершен	  
4	Проект 2	Открыт	  
5	Проект 34	Открыт	  
6	Проект 123	Открыт	  
7	Проект 123	Открыт	  

Рисунок Б.2 - Экранная форма добавления проектов рабочей группы

Корпоративный мессенджер

ПроектыСообщенияПанель администратораВыход (admin)

Главная / Проекты / Создать проект

Создать проект

Название

Описание

Статус проекта

Открыт

Добавить

© Корпоративный мессенджер 2022

Рисунок Б.3 - Создание проекта

Корпоративный мессенджер

ПроектыСообщенияПанель администратораВыход (admin)

Главная / Проекты / Проект 1

Проект 1

Создать проектРедактироватьУдалить

Название	Проект 1
Описание	Описание проекта 1
Статус проекта	Открыт
Дата регистрации	2022-05-31 02:08:59
Дата закрытия	(не задано)
Создал	Зайцев Алексей Петрович

Участники проектаРабочие группы

Рисунок Б.4 - Данные проекта

Участники проекта

Добавить участника проекта

Показаны записи 1-3 из 3.

#	Пользователь	Тип участника	
	<input type="text"/>	<input type="text"/>	
1	Гудкова Анна Павловна	Пользователь	  
2	Зайцев Алексей Петрович	Администратор	  
3	Иванова Анна Владимирович	Администратор	  

Рисунок Б.5 - Список участников проекта

Корпоративный мессенджер

Проекты Сообщения Панель администратора ▾ Выход (admin)

Главная / Проекты / Проект 1 / Участники проекта / Добавить участника проекта

Добавить участника проекта

Пользователь

Тип участника

Добавить

Рисунок Б.6 - Добавление участников проекта

Корпоративный мессенджер

Проекты Сообщения Панель администратора ▾ Выход (admin)

Главная / Проекты / Проект 1 / Участники проекта / Гудкова Анна Павловна

Гудкова Анна Павловна

Добавить участника проекта Редактировать Удалить

Проект	Проект 1
Пользователь	Гудкова Анна Павловна
Тип участника	Пользователь

Рисунок Б.7 - Данные участника проекта

Корпоративный мессенджер

ПроектыСообщенияПанель администратора ▾Выход (admin)

Главная / Проекты / Проект 1 / Рабочие группы

Рабочие группы

Создать рабочую группу

Показаны записи 1-1 из 1.

#	Название	Статус группы	
	<input type="text"/>	<div>▼</div>	
1	Группа 1	Действующая	  

Рисунок Б.8 - Рабочие группы

Корпоративный мессенджер

ПроектыСообщенияПанель администратора ▾Выход (admin)

Главная / Проекты / Проект 1 / Рабочие группы / Создать рабочую группу

Создать рабочую группу

Название

Описание

Статус группы

Действующая

▼

Добавить

Рисунок Б.9 - Форма создания рабочей группы

Корпоративный мессенджер

ПроектыСообщенияПанель администратораВыход (admin)

Главная / Проекты / Проект 1 / Рабочие группы / Группа 1

Группа 1

Создать рабочую группуРедактироватьУдалить

Проект	Проект 1
Дата регистрации	2022-05-31 15:40:47
Дата закрытия	(не задано)
Название	Группа 1
Описание	Описание группы 1
Статус группы	Действующая

Участники группыЗадачи группыСообщения группыДокументы группы

Рисунок Б.10 - Описание рабочей группы

Корпоративный мессенджер

ПроектыСообщенияПанель администратораВыход (admin)

Главная / Проекты / Проект 1 / Рабочие группы / Группа 1 / Участники группы

Участники группы

Добавить участника группы

Показаны записи 1-2 из 2.

#	Пользователь	Тип участника	
1	Зайцев Алексей Петрович	Администратор	
2	Гудкова Анна Павловна	Пользователь	

Рисунок Б.11 - Участники группы

Корпоративный мессенджер

ПроектыСообщенияПанель администратораВыход (admin)

Главная / Проекты / Проект 1 / Рабочие группы / Группа 1 / Участники группы / Добавить участника группы

Добавить участника группы

Пользователь

Тип участника

Добавить

Рисунок Б.12 - Добавление участника группы

[Главная](#) / [Проекты](#) / [Проект 1](#) / [Рабочие группы](#) / [Группа 1](#) / [Участники группы](#) / Гудкова Анна Павловна

Гудкова Анна Павловна

[Добавить участника группы](#)[Редактировать](#)[Удалить](#)

Рабочая группа	Группа 1
Пользователь	Гудкова Анна Павловна
Тип участника	Пользователь

Рисунок Б.13 - Данные участника группы

[Главная](#) / [Проекты](#) / [Проект 1](#) / [Рабочие группы](#) / [Группа 1](#) / [Документы группы](#)

Документы группы

[Добавить документ](#)

Показаны записи 1-3 из 3.

#	Дата регистрации	Название	Имя файла	
	<input type="text"/>	<input type="text"/>	<input type="text"/>	
1	2022-05-31 16:41:47	Документ 1	Desert.jpg	↓ ↑ ✎ ✖
2	2022-06-08 22:56:47	Документ 2	Tulips.jpg	↓ ↑ ✎ ✖
3	2022-06-09 01:09:30	Документ 3	Penguins.jpg	↓ ↑ ✎ ✖

Рисунок Б.14 - Документы группы

[Главная](#) / [Проекты](#) / [Проект 1](#) / [Рабочие группы](#) / [Группа 1](#) / [Документы группы](#) / [Добавить документ](#)

Добавить документ

Название

Описание

[Выбрать ...](#)[Добавить](#)

Рисунок Б.15 - Добавление документа

Корпоративный мессенджер

ПроектыСообщенияПанель администратора ▾Выход (admin)

Главная / Проекты / Проект 1 / Рабочие группы / Группа 1 / Документы группы / Документ 1

Документ 1

Добавить документРедактироватьУдалить

Рабочая группа	Группа 1
Дата регистрации	2022-05-31 16:41:47
Название	Документ 1
Описание	Описание файла 1
Имя файла	Desert.jpg
Создал	Зайцев Алексей Петрович

Скачать

Рисунок Б.16 - Данные документа

Корпоративный мессенджер

ПроектыСообщенияПанель администратора ▾Выход (admin)

Главная / Проекты / Проект 1 / Рабочие группы / Группа 1 / Сообщения группы

Сообщения группы

Создать сообщение

Показаны записи 1-3 из 3.

Моё сообщение8.06.2022 2:27

Сообщение

Иванова Анна Владимирович8.06.2022 1:53

Ответ на тестовое сообщение

Рисунок Б.17 - Сообщения внутри рабочей группы

Приложение В

(обязательное)

Корпоративный мессенджер для приема и передачи текстовых
сообщений

Схема данных

					ПГУ 1.090501.02.001 ПЗ	Лист
						123
Изм.	Лист	№ докум.	Подпись	Дата		

Приложение Г

(обязательное)

Корпоративный мессенджер для приема и передачи текстовых
сообщений

Диаграмма классов

					ПГУ 1.090501.02.001 ПЗ	Лист
						125
Изм.	Лист	№ докум.	Подпись	Дата		

Приложение Д

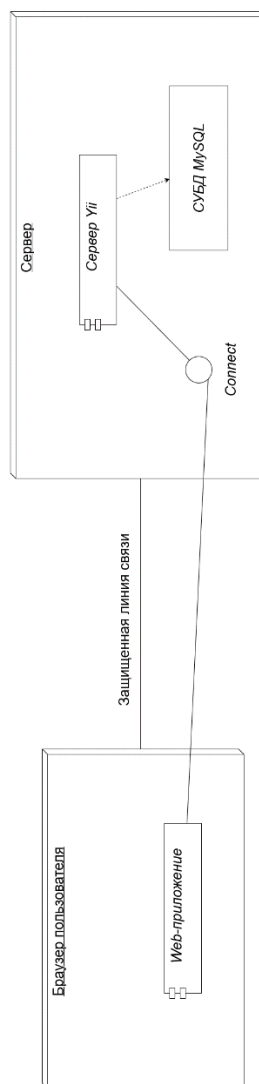
(обязательное)

Корпоративный мессенджер для приема и передачи текстовых
сообщений

UML-диаграммы

					ПГУ 1.090501.02.001 ПЗ	Лист
						127
Изм.	Лист	№ докум.	Подпись	Дата		

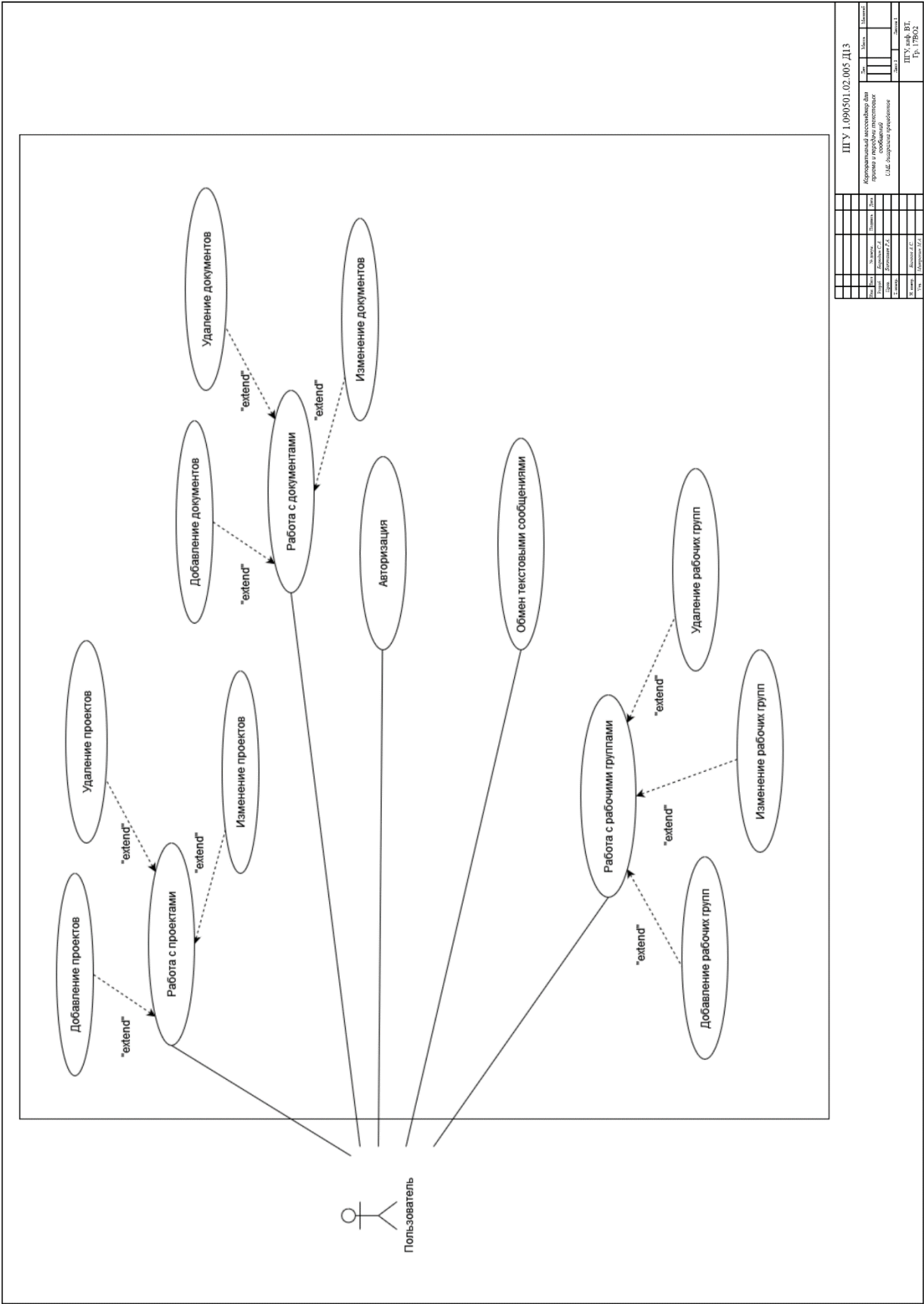
Приложение Д.1 – Диаграмма развертывания

[illegible]

<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>

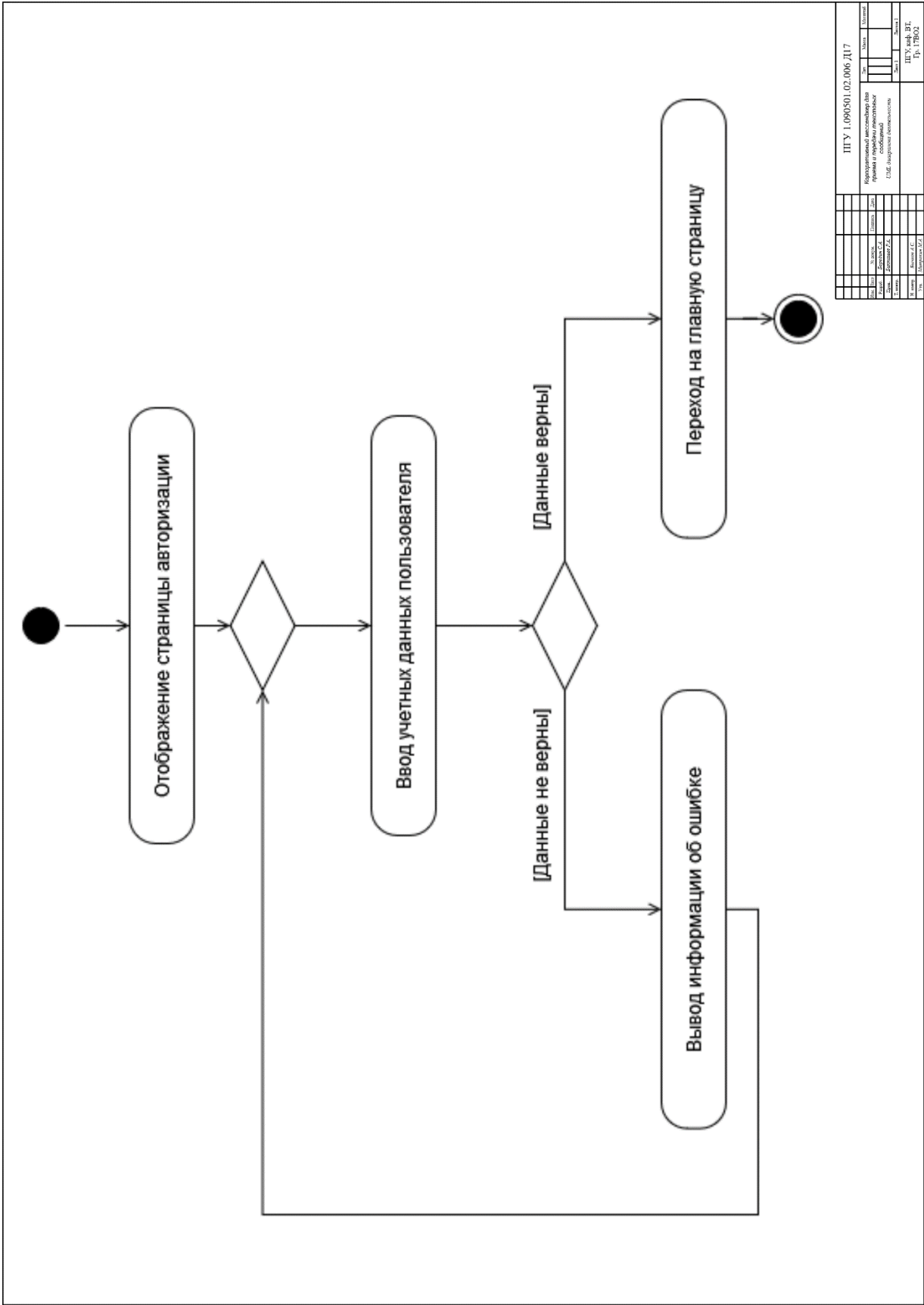
ПГУ 1.090501.02.001 ПЗ

Приложение Д.2 – Диаграмма прецедентов



ПГУ 1.090501.02.005 Д13									
Информационный ресурс системы для проектирования и разработки программного обеспечения									
С.М.С. Информационные ресурсы									
ПГУ, инф. БД, Гл. 1, 19802									
№ докум.	№ листа	Исполнитель	Дата	Масштаб	Лист				
1	1								
2	2								
3	3								
4	4								
5	5								
6	6								
7	7								
8	8								
9	9								
10	10								
11	11								
12	12								
13	13								
14	14								
15	15								
16	16								
17	17								
18	18								
19	19								
20	20								
21	21								
22	22								
23	23								
24	24								
25	25								
26	26								
27	27								
28	28								
29	29								
30	30								
31	31								
32	32								
33	33								
34	34								
35	35								
36	36								
37	37								
38	38								
39	39								
40	40								
41	41								
42	42								
43	43								
44	44								
45	45								
46	46								
47	47								
48	48								
49	49								
50	50								
51	51								
52	52								
53	53								
54	54								
55	55								
56	56								
57	57								
58	58								
59	59								
60	60								
61	61								
62	62								
63	63								
64	64								
65	65								
66	66								
67	67								
68	68								
69	69								
70	70								
71	71								
72	72								
73	73								
74	74								
75	75								
76	76								
77	77								
78	78								
79	79								
80	80								
81	81								
82	82								
83	83								
84	84								
85	85								
86	86								
87	87								
88	88								
89	89								
90	90								
91	91								
92	92								
93	93								
94	94								
95	95								
96	96								
97	97								
98	98								
99	99								
100	100								

Приложение Д.3 – Диаграмма деятельности



ПТУ 1.090501.02.006 Д17									
Информационный паспорт для применения в образовательных организациях СЗД образовательных учреждений									
ПТУ каф. ИТ, г.р. 17803									
№ п/п	№ докум.	Дата	Исполн.	Провер.	Дата	Исполн.	Провер.	Дата	Исполн.
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									
25									
26									
27									
28									
29									
30									
31									
32									
33									
34									
35									
36									
37									
38									
39									
40									
41									
42									
43									
44									
45									
46									
47									
48									
49									
50									
51									
52									
53									
54									
55									
56									
57									
58									
59									
60									
61									
62									
63									
64									
65									
66									
67									
68									
69									
70									
71									
72									
73									
74									
75									
76									
77									
78									
79									
80									
81									
82									
83									
84									
85									
86									
87									
88									
89									
90									
91									
92									
93									
94									
95									
96									
97									
98									
99									
100									