

Цифровые технологии

Автоматизация тестирования UI с использованием Selenium и Pytest

Дипломный проект

студента

по программе “Инженер–тестировщик. ”

Тема: «Эффективность тестирования web-приложений»

Богданова Е.Ю.

г. Электросталь
2023

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	5
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТАНОВКА ЗАДАЧИ ...	8
1.1 ТЕХНИКО-ЭКОНОМИЧЕСКАЯ ХАРАКТЕРИСТИКА ПРЕДМЕТНОЙ ОБЛАСТИ И ПРЕДПРИЯТИЯ» АНАЛИЗ ДЕЯТЕЛЬНОСТИ «КАК ЕСТЬ».....	8
1.1.1 Характеристика предприятия и его деятельности.....	8
1.1.2 Организационная структура управления предприятием	8
1.1.3 Программная и техническая архитектура ИС предприятия 10	
1.2 ХАРАКТЕРИСТИКА КОМПЛЕКСА ЗАДАЧ, ЗАДАЧИ И ОБОСНОВАНИЕ НЕОБХОДИМОСТИ АВТОМАТИЗАЦИИ.....	12
1.2.1 Выбор комплекса задач автоматизации и характеристика существующих бизнес процессов.....	12
1.2.2 Определение места проектируемой задачи в комплексе задач и ее описание	17
1.2.3 Обоснования необходимости использования вычислительной техники для решения задачи.....	26
1.2.4 Анализ системы обеспечения информационной безопасности и защиты информации	29
1.3 АНАЛИЗ СУЩЕСТВУЮЩИХ РАЗРАБОТОК И ВЫБОР СТРАТЕГИИ АВТОМАТИЗАЦИИ «КАК ДОЛЖНО БЫТЬ»	34
1.3.1 Анализ существующих разработок для автоматизации задачи 34	
1.3.2 Выбор и обоснование стратегии автоматизации задачи...	38
1.3.3 Выбор и обоснование способа приобретения ИС для автоматизации комплекса задач	40
1.4 ОБОСНОВАНИЕ ПРОЕКТНЫХ РЕШЕНИЙ.....	42
1.4.1 Обоснование проектных решений по информационному обеспечению 42	

1.4.2	Обоснование проектных решений по программному обеспечению	48
1.4.3	Обоснование проектных решений по техническому обеспечению	52
2	ПРОЕКТНАЯ ЧАСТЬ	54
2.1	РАЗРАБОТКА ПРОЕКТА АВТОМАТИЗАЦИИ.....	54
2.1.1	Этапы жизненного цикла проекта автоматизации	54
2.1.2	Ожидаемые риски на этапах жизненного цикла и их описание	68
2.1.3	Организационно-правовые и программно-аппаратные средства обеспечения информационной безопасности и защиты информации	73
2.2	ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ЗАДАЧИ	75
2.2.1	Информационная модель и её описание.....	75
2.2.2	Характеристика нормативно-справочной, входной и оперативной информации	76
2.2.3	Характеристика результатной информации.....	81
2.3	ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ЗАДАЧИ	82
2.3.1	Общие положения (дерево функций и сценарий диалога)	82
2.3.2	Характеристика базы данных	85
2.3.3	Структурная схема пакета (дерево вызова программных модулей)	91
2.3.4	Описание программных модулей.....	94
2.4	КОНТРОЛЬНЫЙ ПРИМЕР РЕАЛИЗАЦИИ ПРОЕКТА И ЕГО ОПИСАНИЕ	95
3	ОБОСНОВАНИЕ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ПРОЕКТА.....	102
3.1	ВЫБОР И ОБОСНОВАНИЕ МЕТОДИКИ РАСЧЕТА ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ.....	102

3.2	РАСЧЁТ ПОКАЗАТЕЛЕЙ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ПРОЕКТА	107
	ЗАКЛЮЧЕНИЕ	114
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	116

Введение

Важно понимать, что технологии так или иначе проникнут даже в те сферы, которые еще недавно были доступны только человеческому духу, в т. ч. сюда можно отнести и проектную деятельность.

Сейчас нельзя 100% назвать сферы жизни людей, которые будут автоматизированы. Сам человек также совершенствуется вместе с технологиями, становится более свободным, техника позволяет освободить место в его жизни для более совершенных помыслов.

Проект как система деятельности существует ровно столько времени, сколько его требуется для получения конечного результата. Концепция проекта, однако, не противоречит концепции фирмы или предприятия и вполне совместима с ней. Более того, проект часто становится основной формой деятельности фирмы. С этим связана актуальность рассматриваемой темы.

Постановка вопроса контроля проектом выражается в понятии «проект», где он становится только объектом управления, который имеет определенные лишь для него свойства, но и как общее выражение сути, как базовое качество уровня проработанности проекта.

Сегодня можно найти множество трактовок определения «проект». И базируются они на классическом понимании проекта: одна цель, ограниченность времени и ресурсов. Есть пара негативных сторон – нет связи с проектом в рамках реализованного ранее плана и проектом в виде процедуры реализации данного плана, а также нет отношения между контролем и проектом.

Цель выпускной квалификационной работы состоит в разработке прототипа программного обеспечения для автоматизации тестирования WEB приложения.

К задачам проекта отнесены:

- Проанализировать деятельность компании и определить бизнес-процесс для автоматизации;
- Выбрать средства разработки;
- Определить требования к информационной системе;

- Определить проектные решения по информационному и техническому обеспечению;
- Определить способ приобретения и стратегию автоматизации;
- Выбрать стандарт жизненного цикла;
- Определить риски при реализации проекта и методы их минимизации;
- Описать информационную модель системы;
- Описать программное обеспечение задачи автоматизации;
- Описать контрольный пример реализации системы;
- Провести расчет экономических показателей эффективности.

Компания ООО «Аутсорсинг Контракт» (то есть объект исследования) является предприятием малого бизнеса с основным видом деятельности – разработка программного обеспечения. Спецификой деятельности компании является большое количество удаленных сотрудников, что диктует необходимость наличия единой платформы для работы с ними.

Задачами дипломной работы в связи с указанной целью являются:

- изучить основные понятия теории тестирования;
- исследовать наиболее распространенные web-технологии;
- изучить систему контроля версий Git;
- изучить базовые возможности языка программирования Python;
- изучить основные возможности и методы Selenium;
- исследовать расширение возможностей тестирования с помощью фреймворков;
- раскрыть необходимость использования паттерна Page Object Model;
- изучить различные методы формирования отчетов о тестировании.

Предметом исследования является деятельность компании по тестированию веб-приложений.

Работа состоит из введения, заключения, трех глав.

В первой главе проводится анализ исследуемой компании, ее структуры управления, программной и технической архитектуры информационной системы.

Анализируются бизнес-процессы компании, делаются выводы о необходимости автоматизации исследуемого процесса. Далее формируются

решения по основным видам обеспечения – информационному, программному, техническому.

Во второй главе производится выбор стандарта жизненного цикла ИС, описываются основные риски, меры по защите информации, а также дается характеристика информационному и программному обеспечению ИС, в том числе приводится информационная модель, схемы дерева функций и сценария диалога, структурной схемы, приводится контрольный пример реализации системы.

В третьей главе выбрана методика расчета экономической эффективности и проведен расчет снижения показателей трудовых и стоимостных затрат, делается вывод о срок окупаемости информационной системы.

1 Анализ предметной области и постановка задачи

1.1 Технико-экономическая характеристика предметной области и предприятия» Анализ деятельности «КАК ЕСТЬ»

1.1.1 Характеристика предприятия и его деятельности

Объектом исследования выпускной квалификационной работы является общество с ограниченной ответственностью ООО «Аутсорсинг Контракт».

Сфера деятельности организации - широкий спектр услуг в области информационных технологий, разработка программного обеспечения, создание и продвижение сайтов, включая комплексный интернет маркетинг, внедрение законченных web решений.

Основные показатели деятельности компании приведены в таблице 1.1.

Таблица 1.1

Показатели, характеризующие деятельность ООО «Аутсорсинг Контракт»

Показатель	Единица измерения	2023 год	2019 год	2018 год
Объем реализации услуг	т.руб.	108900	133947	164755
Численность работающих	чел.	15	21	29
Фонд заработной платы персонала	т.руб.	350	521,5	777,035
Среднегодовая зарплата 1-го работающего	т.руб.	23	25	30

1.1.2 Организационная структура управления предприятием

Тип организационной структуры компании - линейный, так как во главе стоит руководитель, сосредоточивший в своих руках все функции управления и осуществляющий единоличное руководство подчиненными ему работниками.

Схема организационной структуры объекта представлена на рисунке 1.1.



Рисунок 1.1 Организационная структура компании

Как видно из представленной схемы, во главе структуры стоит генеральный директор, который выполняет общее руководство предприятием. Все остальные отделы подчинены генеральному директору через руководителей отделов и групп.

Бухгалтерия во главе с главным бухгалтером ведет бухгалтерский и налоговый учет, сдает отчетность, составляет предусмотренные действующим законодательством отчеты, балансы и иные документы, также оформляет в порядке, установленном действующим законодательством, платежные и иные расчетные документы, в том числе по уплате налогов и сборов, осуществляет прием, хранение товарно-материальных ценностей.

К функциям отдела по работе с текущими клиентами относят следующие:

- Осуществлять поиск и привлечение клиентов на комплексное обслуживание в компанию, проведение переговоров с клиентами, предложение действующим клиентам дополнительных услуг, посредством направления писем и проведение переговоров с клиентами.

- Обеспечивать клиентов необходимыми рекламно-информационными материалами.

- Оказывать информационные и консультационные услуги клиентам.

В функции сотрудников отдела продаж входит оказание коммерческих услуг, обеспечение выполнения условий, предусмотренных заключенными соглашениями, сопровождение клиента.

Информационный отдел компании включает в себя начальника отдела, главных специалистов, ведущих специалистов, специалистов 1 категории.

Информационный отдел отвечает за:

- получение от пользователей сообщений об запросах;
- организацию решения запросов на уровне, удовлетворительном для клиента;
- отслеживание статуса запроса в ходе работы над ним;
- информирование клиента о ходе решения его проблем и восстановлении нормального функционирования услуг;
- мониторинг, управление и выдача отчетов о качестве услуг, которые предоставляются каждому отдельному пользователю.

1.1.3 Программная и техническая архитектура ИС предприятия

На рисунке 1.2 показана техническая архитектура ООО «Аутсорсинг Контракт».

Структура представляет собой комплекс серверов, свитч, маршрутизатор и АТС.

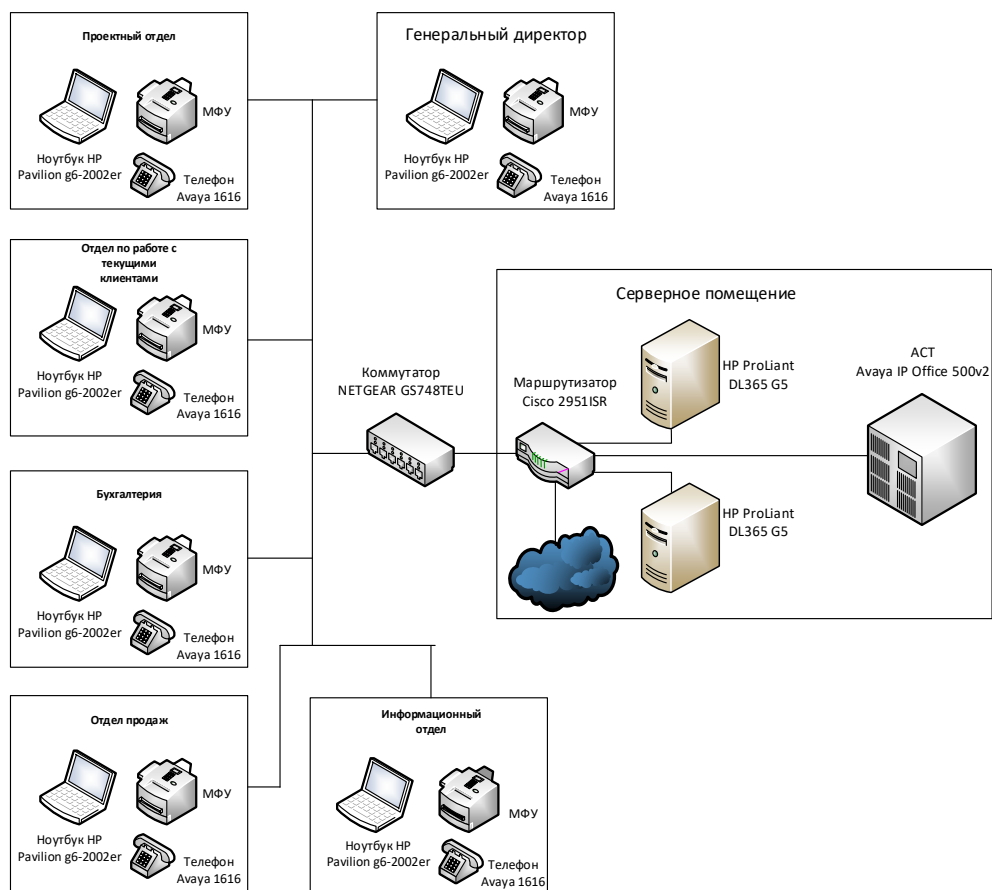


Рисунок 1.2 Техническая архитектура ООО «Аутсорсинг Контракт»

Программная архитектура ООО «Аутсорсинг Контракт» изображена на рисунке 1.3.

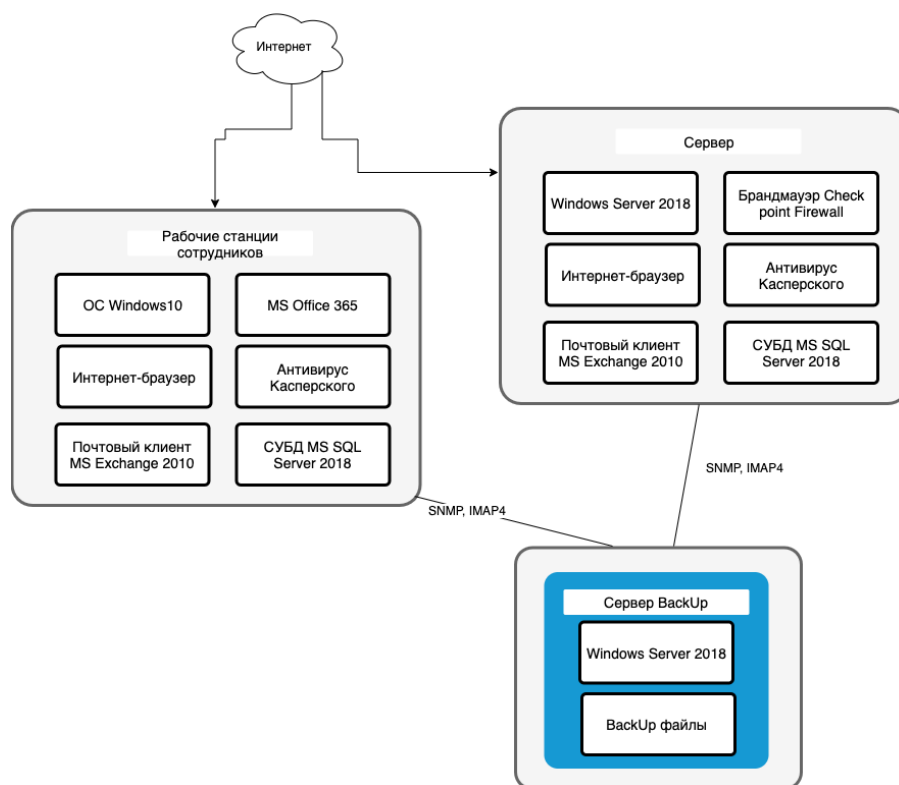


Рисунок 1.3 Программная архитектура ООО «Аутсорсинг Контракт»

На всех рабочих станциях сотрудников установлена операционная система Microsoft Windows 10.

В ООО «Аутсорсинг Контракт» используется «1С Управление Производством Предприятия» версии 8.2. Решение позволяет организовать единую информационную систему для управления деятельностью предприятия: мониторинг и анализ показателей деятельности предприятия, финансами, персоналом, включая расчет заработной платы, отношениями с покупателями и поставщиками, продажами, закупками, складом (запасами), основными средствами и планированием ремонтов.

1.2 Характеристика комплекса задач, задачи и обоснование необходимости автоматизации

1.2.1 Выбор комплекса задач автоматизации и характеристика существующих бизнес процессов

В настоящее время выполнение заказов по разработке сайтов осуществляется в компании путем создания проекта, назначения менеджера проекта и подбора исполнителей по специализациям – программистов,

дизайнеров, верстальщиков и так далее. Работа и общение менеджера проекта с участниками проекта ведётся удаленно, с использованием таких средств, как телефонная связь, интернет-менеджеры и так далее.

Основным действующим лицом является менеджер проекта.

К обязанностям менеджера проектов относятся ведение, реализация и управление проектами, включая:

- управление полным циклом разработки новых технологий и координацию программ со стратегическими партнерами, ускорение цикла внедрения новых версий продуктов, завоевание доверия заказчика и позитивной репутации в результате эффективной работы по проектированию, внедрению и поддержке продуктов;
- контроль выполнения проекта и соблюдением сроков, указанных в плане, ответственность за реализацию проекта в целом, проверку и утверждение продукта во всех его аспектах - функциональность, критерии качества.

Состав рабочей группы проекта приведен на рисунке 1.4.

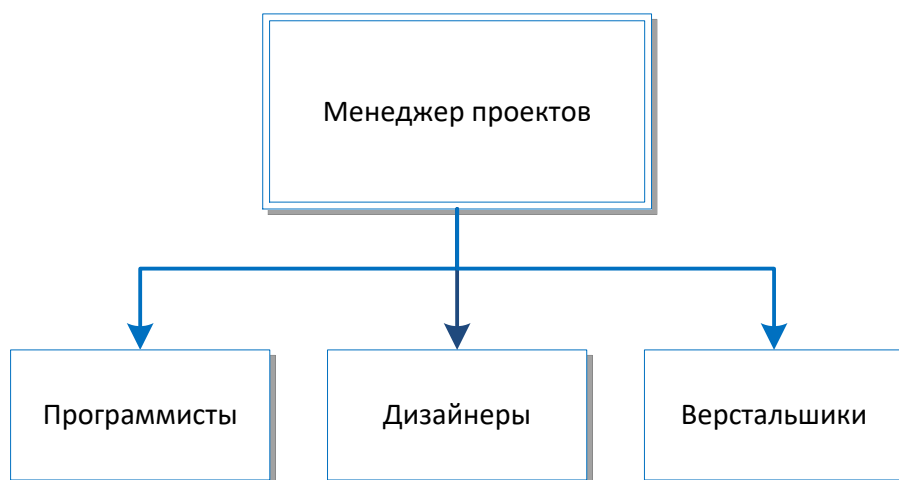


Рис. 1.4 Состав группы, работающей над проектом

Динамика изменения количества проектов, которые одновременно ведет компания ООО «Аутсорсинг Контракт» показывает тенденцию к увеличению количества проектов компании. Общее количество проектов в 2021 году по сравнению с 2020 годом увеличилось на 130% и, судя по представленному тренду, будет расти и дальше.

С увеличением данного показателя управление на уровне совещаний с руководителем и контролем с его стороны, без ввода автоматизированной

системы становится не только проблематичным, но и во многих вопросах невозможным.

Рассмотрим с помощью технологии IDEF, как в настоящее время происходит обработка, выполнение и контроль проектов. На рисунке 1.5 показана общая характеристика деятельности менеджера проектов.

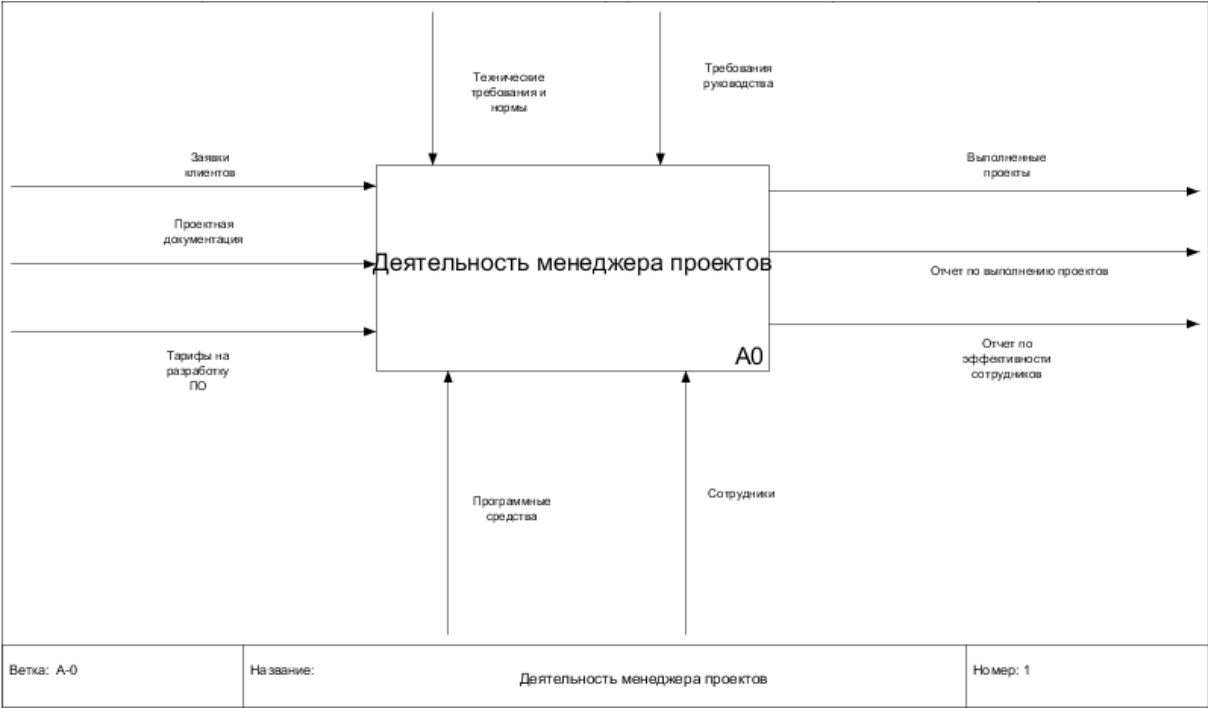


Рис. 1.5 Общая характеристика деятельности менеджера проектов

Декомпозиция данной схемы приведена на рисунке 1.6.

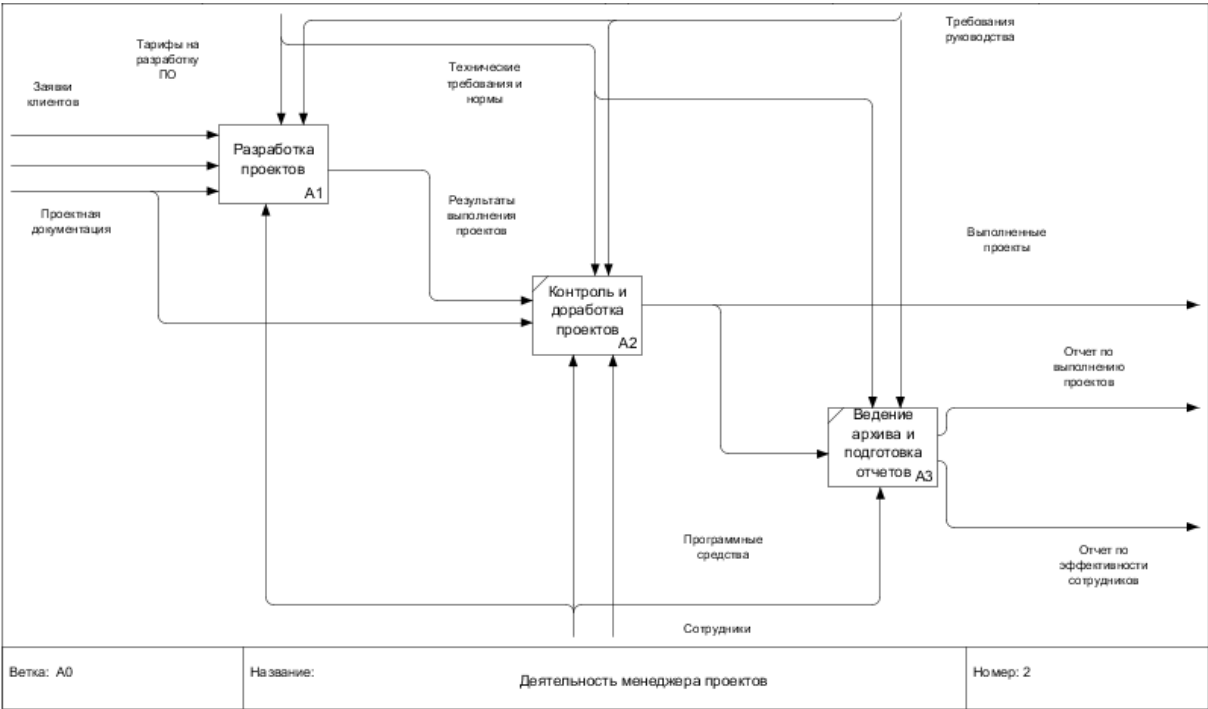


Рис. 1.6 Декомпозиция основного процесса

Как видно из приведенной схемы, основными блоками в деятельности отдела являются следующие:

- Разработка проектов по заявкам клиентов;
- Контроль выполнения проектов и их доработка (при необходимости);
- Ведение архива документации, в том числе и по выполненным проектам.

Крупных и сложных систем, которые ежедневно обрабатывают большие объёмы данных, нуждаются в проведение нагрузочного тестирования. Часто бывает, что система хорошо справляется с обработкой небольшого объёма данных, но с повышением интенсивности возникают различного рода проблемы, такие как: лимит системных ресурсов, конфликт взаимодействия смежных элементов, использование ресурсоёмких запросов и т. д.

Подобного рода проблемы необходимо отлавливать на тестовых стендах и не допускать выхода в эксплуатацию, так как в условиях реальной пиковой нагрузки, необходимо обеспечить максимально надёжную систему без сбоев. Так как в случае чего сбой может привести к замедлению системы или же к полной остановке, что для «mission-critical» систем чревато огромными финансовыми потерям, которые превышают стоимость нагрузочного тестирования. Побочным эффектом от нагрузочного тестирования является то, что помимо оценки системы под нагрузкой, регулярно проводимые тестирования позволяют собирать статистику с системы, на основе которой можно выполнить сравнительный анализ эффективности доработок различных систем, а так же оценивать производительность железа перед заказом крупной поставки.

Для проведения нагрузочного теста можно выделить следующие самостоятельные этапы:

1. Анализ заявленных требований перед тестированием;
2. Настройка тестового стенда;

3. Разработка сценариев нагрузки;
4. Реализация модели нагрузки;
5. Проведение нагрузочных испытаний;
6. Анализ результатов тестирования.

На практике же первые четырёх этапа сливаются в один сложный подготовительный этап тестирования. Этот этап является наиболее сложным и отнимает очень много времени, в среднем это где-то 1 - 2 дня, для одной доработки.

Реализация модели нагрузки заключается в разработке тестовых скриптов, реализующих отдельные бизнес-операции. Для проведения нагрузочного испытания необходимо составить сценарий, который включает в себя один или несколько автоматизированных скриптов. В ходе прогона сценария, скрипты эмулируют работу реальных пользователей.

По итогам испытаний тестирование позволяет получить графики зависимости измеряемых параметров при изменении нагрузки (числа пользователей) и неизменной конфигурации системы. На графиках показаны примеры зависимостей времени отклика и загрузки системных данных от числа пользователей (пользовательской нагрузки).

Эти результаты оформляются в виде отчёта с дополнительными комментариями об «узких местах» программной системы с указанием причин. Тем самым указывая на явные первостепенные причины появления возможной аварийной ситуации, такие данные носят название первичных результатов.

Благодаря первичным результатам могут быть построены прогнозные модели, позволяющие оценить производительность тестируемого ПО. Полученные таким образом результаты относятся к классу вторичных или аналитических результатов проведённого тестирования.

Если результаты тестирования подтвердили необходимость увеличения мощности системных ресурсов, то фактически встаёт вопрос о

выборе программно-аппаратной платформы, на которой ПО будет поддерживать в системе параллельную работу заданного количества пользователей. В этом случае, возможно, потребуется проведение серии испытаний системы на различных платформах.

Как правило, нагрузочное тестирование является неотъемлемым этапом внедрения системы, однако довольно распространенной становится ситуация, когда необходимо тестирование промышленной системы, работающей в режиме 24/7. Такая необходимость, например, возникает, когда система, успешно проработав несколько месяцев или даже лет, приближается к своей пиковой нагрузке по объему данных или количеству пользователей.

Чтобы избежать остановки системы на время проведения тестирования, а также лишних рисков, обычно создается тестовый стенд, который воспроизводит работу промышленной системы и на базе которого проводятся все эксперименты. Однако в этом случае задача тестировщиков может усложняться в силу следующих обстоятельств:

редко удается полностью продублировать конфигурацию системы на тестовом стенде и используется менее мощное оборудование;

в ряде случаев, из соображений конфиденциальности и безопасности, владелец промышленной системы (заказчик) ограничивает доступ к информации о логике работы процессов на сервере и клиентах, о конфигурации и архитектуре системы.

Дополнительные сложности могут быть обусловлены шифрованием уникальным ключом каждой сессии и каждого передаваемого пакета, нестандартным протоколом обмена и т. д. В частности, все указанные сложности характерны практически для любого серьезного финансового приложения.

1.2.2 Определение места проектируемой задачи в комплексе задач и ее описание

В результате анализа деятельности компании, приведенного в пунктах выше, можно сделать вывод, что рассматриваемая компания является достаточно крупной, имеет большое количество разных бизнес-процессов различной степени автоматизации, но некоторые процессы (среди них и процесс документооборота) не автоматизированы вообще. В результате часть управленческой информации, которая необходима для приема управленческих решений, не доходит до руководства, кроме того, в компании замедлен процесс оборота документальной информации, которую необходимо с бумажных носителей переносить в информационную систему компании.

Проект – это некая задача с указанными начальными данными и требуемыми итогами, которые отражают принцип решения. Сам проект включает проблему, ее вариации реализации и реализованный итог.

Исходя из работы содружества руководителей проектов, определим ряд функций, которые требуют решения от менеджера проекта:

- Включение и состав работ (создание методологии, описание области исследования проекта, изменение работ, генерация отчетных документов, применение методик контроля, описание полученных итогов);
- Уровень проекта (ведение и отслеживание качества);
- Длительность (оценка затраченного времени, длительности, планы по календарю);
- Стоимость (учет цены работ, подготовка бюджета и смет, изменение наценки, применение стоимостных параметров);
- Риск;
- Ресурсы человеческие и временные;
- Доступность взаимодействия;
- Постоянство поставок.

Отслеживание всех приведенных функций включает десятки, а иногда и сотни минимальных факторов и составляющих. Суть контроль проекта позволяет придать этим процедурам некую систематизацию и направленность.

Потому суть контроля проекта определяется как связанный процесс.

Бюджет проекта. Явная ошибка состоит в стороннем понимании механизма оценки проекта и его оплаты. Часто можно запутаться в начислении и продумывании бюджетных средств. В отличие от оплаты, сам процесс учета включает множество параметров, которые влияют на проект.

Если компания дает средства без изначального подсчета, то часто сумма будет меньше, и сделать что-то нормальное будет нереально. Проблема определения бюджетных средств сложна и непроста. И неадекватно выделенный бюджет часто приводит к появлению разного рода проблем. Малый бюджет часто ведет к огромным расходам впоследствии.

Для проекта требуется наличия целей, конкретного типа ресурсов, повышенного уровня закрытости начальных показателей, возможность реализации особых ситуаций, наличие конкретных длительностей начала и окончания проекта. Его реализация включает координацию в границе внедрения.

Явными в понимании подобного подхода становится смена вертикальной структуры фирмы на горизонтальную. Изначально процессный подход описали М. Хамера и Дж. Чампи в труде «Перестроение компаний. Манифест революции в бизнесе». Тут описаны важные моменты контроля фирмы в рамках подхода на функциональном уровне, а также описано, что в XXI в. фирмы с такой структурой показывали «негибкость, малую адаптивность и ориентацию на клиента, заикленность на работах, а не на итоговых показателях, бумагомарательство, малый процент новшеств, огромны издержки». Скорректированные условия ведут компании к обновленному управленческому подходу.

Определим некоторые итоги, достигаемые при реализации ПОП:

1) Уменьшение длительности работы процесса при взаимном росте уровня выполняемых работ благодаря оптимизации передачи данных по цепочке управления. Работа руководства делается для того, чтобы внедрить все функции и процессы, при этом изменяются отправляемые данные, снижается итоговый результат;

2) Есть шанс узнать эффективность функций, создаваемых в рамках процесса, исходя из результатов самого процесса. При подходе в рамках функций все итоги персонала описаны исходя из отдельного понимания самого

руководства об уровне итогов работ, а не с точки зрения совокупной важности отдельного процесса;

3) Поддержка корректности итогов в протекании процессов. Подход в рамках функций отражается не состыковками в интересах и работах отделов компании, что в итоге ведёт к сокращению самих процессов;

4) Снижение ценовых расходов и издержек. В рамках подхода в функции расходы растут из-за множества операций, проводимых для передачи данных и конкретных итогов между отделами, а также из-за множества сторонних операций, создаваемых по факту отстраненности исполнителя от своей личной роли;

5) Возможность реализации системы мотивирования, базирующейся на благодарности сотрудников в рамках реализации неких итогов работ, где они участвуют. При функциональном подходе заинтересованность персонала в итоге проекта крайне мала, т.к. основным потребителям итогов их деятельности выступают руководство

ПОП к контролю поможет улучшить работу так, чтобы она стала более изменчивой и направленной на рост качества выпускаемого продукта, изменить в меньшую сторону цены и в большую удовлетворенность и лояльность клиента.

Рассмотрим процесс разработки проектов (Рис. 1.7).

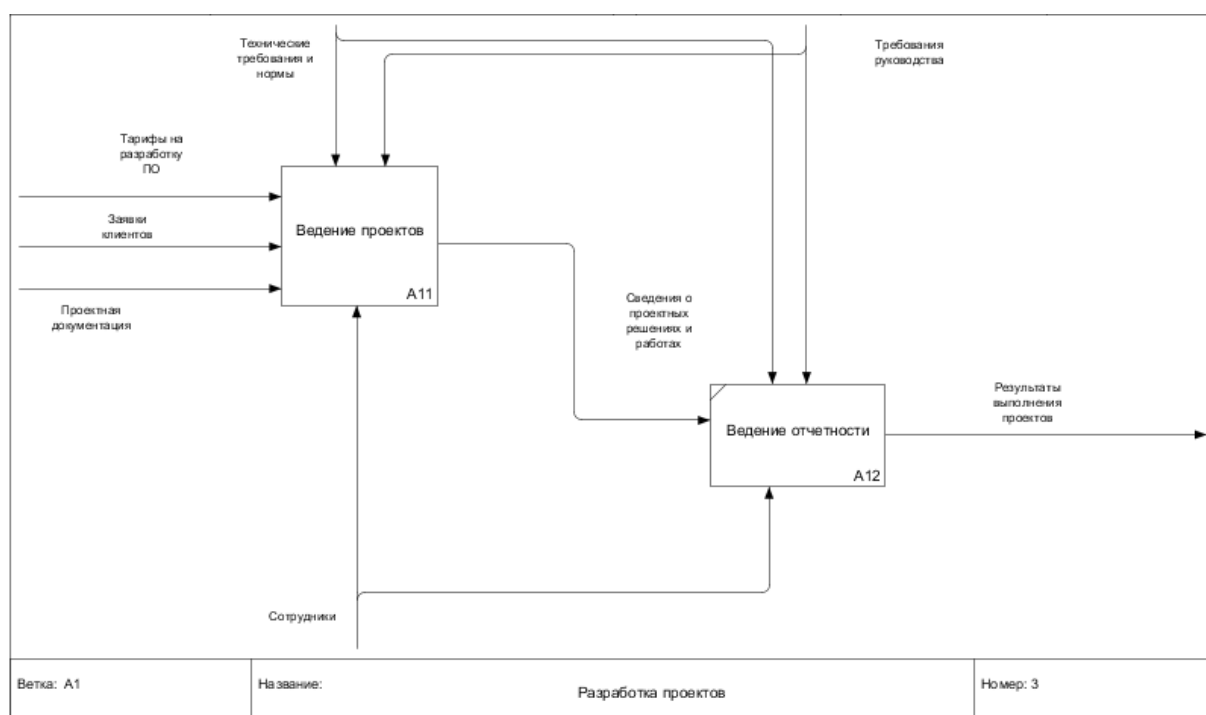


Рис. 1.7 Характеристика процесса разработки проектов

Разработка проектов включает в себя ведение проектов и ведение отчетности по ним.

Ведение проекта включает в себя следующие подпроцессы (Рис. 1.8):

- Регистрация и распределение проектов;
- Непосредственное выполнение проектов, где результат формируется в виде файлов;
- Учет выполненных работ каждым специалистом.

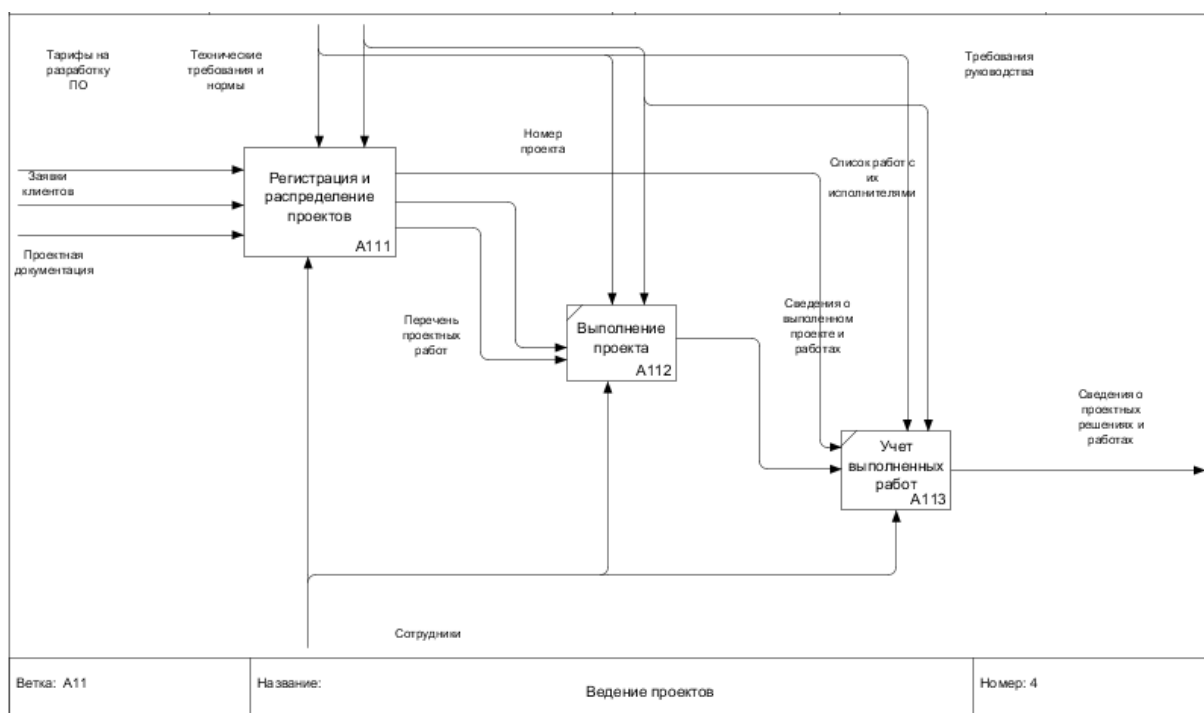


Рис. 1.8 Схема декомпозиции процесса ведения проектов

Данный процесс также состоит из следующих подпроцессов (Рис. 1.9).

- Регистрация проекта, которая заключается в его записи специальную книгу учета;
- Первичный анализ требования проекта, на основании которого производится выбор исполнителя для конкретного проекта с учетом также и его загруженности;
- Распределение проектов между исполнителями, то есть назначение специалиста для исполнения конкретного проекта.

При назначении проекта формируются все требования, его касающиеся, а также указываются сроки его выполнения. Все эти действия выполняет менеджер проектов.

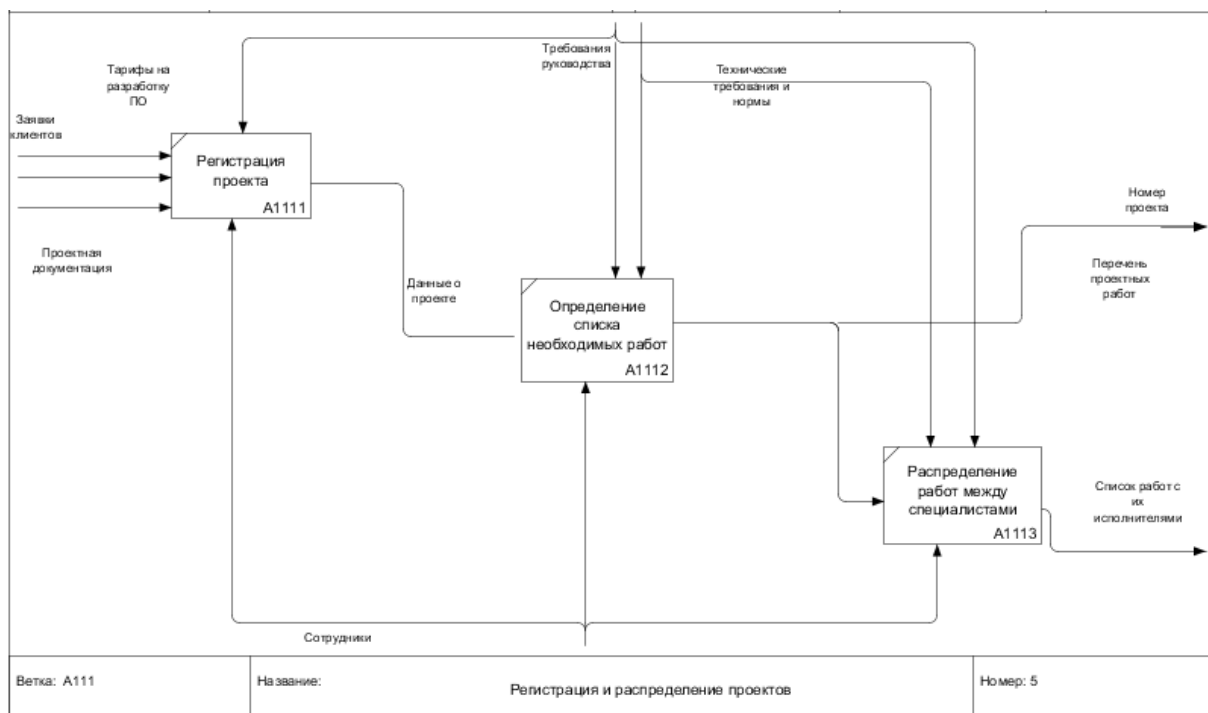


Рис. 1.9 Характеристика процесса регистрации и распределения проекта

На данном этапе развития компании назрела необходимость использовать новые коммуникационные возможности в сфере управления проектами.

Регистрация проекта осуществляется на основании поданной клиентом заявки. Также используются данные клиента. В результат формируется регистрационная карточка проекта, в которой указаны сведения о проекте, дата его завершения, данные клиента, прикреплено техническое задание (при наличии).

Далее, на основании проектной документации и регистрационной карточки проекта осуществляется анализ требований к проекту, в результате чего формируется план-график выполнения проекта, в котором указаны этапы проекта с датами их выполнения. На основании план-графика списка имеющихся исполнителей формируется план распределения работ между исполнителями, в котором указаны задачи для каждого привлекаемого сотрудника и срок их выполнения.

После выполнения проекта на основании плана распределения и сделанных в нем отметок о выполнении мероприятий формируется акт выполненных работ и отчет об их выполнении.

На основании актов выполненных работ раз в месяц формируется сводная ведомость выполненных работ, в которой указывается наименование проекта,

дата приема его в работу и дата окончания, затраченное время, привлекаемые исполнители.

В результате анализа выполненных проектов менеджером формируется аналитическая ведомость, в которой, кроме указанных выше реквизитов, приводятся итоговые показатели по успешности выполнения проектов.

Одним из этапов разработке веб-приложений является их тестирование.

Тестирование может быть разных видов.

Модульное тестирование (юнит-тестирование) — тестируется минимально возможный для тестирования компонент, например, отдельный класс или функция. Часто модульное тестирование осуществляется разработчиками ПО.

Интеграционное тестирование — тестируются интерфейсы между компонентами, подсистемами. При наличии резерва времени на данной стадии тестирование ведётся итерационно, с постепенным подключением последующих подсистем.

Системное тестирование — тестируется интегрированная система на её соответствие требованиям.

Альфа-тестирование — имитация реальной работы с системой штатными разработчиками, либо реальная работа с системой потенциальными пользователями/заказчиком. Чаще всего альфа-тестирование проводится на ранней стадии разработки продукта, но в некоторых случаях может применяться для законченного продукта в качестве внутреннего приёмочного тестирования. Иногда альфа-тестирование выполняется под отладчиком или с использованием окружения, которое помогает быстро выявлять найденные ошибки. Обнаруженные ошибки могут быть переданы тестировщикам для дополнительного исследования в окружении, подобном тому, в котором будет использоваться ПО.

Бета-тестирование — в некоторых случаях выполняется распространение версии с ограничениями (по функциональности или времени работы) для некоторой группы лиц, с тем чтобы убедиться, что продукт содержит достаточно мало ошибок. Иногда бета-тестирование выполняется для того, чтобы получить обратную связь о продукте от его будущих пользователей.

Часто для свободного/открытого ПО стадия Альфа-тестирования характеризует функциональное наполнение кода, а Бета тестирования — стадию исправления ошибок. При этом как правило на каждом этапе разработки промежуточные результаты работы доступны конечным пользователям.

Существует несколько признаков, по которым принято производить классификацию видов тестирования. Обычно выделяют следующие:

По объекту тестирования:

- Функциональное тестирование (functional testing)
- Нагрузочное тестирование
- Тестирование производительности (perfomance/stress testing)
- Тестирование стабильности (stability/load testing)
- Тестирование удобства использования (usability testing)
- Тестирование интерфейса пользователя (UI testing)
- Тестирование безопасности (security testing)
- Тестирование локализации (localization testing)
- Тестирование совместимости (compatibility testing)

По знанию системы:

- Тестирование чёрного ящика (black box)
- Тестирование белого ящика (white box)
- Тестирование серого ящика (gray box)

По степени автоматизированности:

- Ручное тестирование (manual testing)

- Автоматизированное тестирование (automated testing)
- Полуавтоматизированное тестирование (semiautomated testing)

По степени изолированности компонентов:

- Компонентное (модульное) тестирование (component/unit testing)
- Интеграционное тестирование (integration testing)
- Системное тестирование (system/end-to-end testing)

По времени проведения тестирования:

- Альфа тестирование (alpha testing)
- Тестирование при приёмке (smoke testing)
- Тестирование новых функциональностей (new feature testing)
- Регрессионное тестирование (regression testing)
- Тестирование при сдаче (acceptance testing)
- Бета тестирование (beta testing)

По признаку позитивности сценариев:

- Позитивное тестирование (positive testing)
- Негативное тестирование (negative testing)

По степени подготовленности к тестированию:

- Тестирование по документации (formal testing)
- Эд Хок (интуитивное) тестирование (ad hoc testing)

Тестирование «белого ящика» и «чёрного ящика»

В терминологии профессионалов тестирования (программного и некоторого аппаратного обеспечения), фразы «тестирование белого ящика» и «тестирование чёрного ящика» относятся к тому, имеет ли разработчик тестов доступ к исходному коду тестируемого ПО, или же тестирование выполняется через пользовательский интерфейс либо прикладной программный интерфейс, предоставленный тестируемым модулем.

При тестировании белого ящика (англ. white-box testing, также говорят — прозрачного ящика), разработчик теста имеет доступ к исходному коду программ и может писать код, который связан с библиотеками тестируемого ПО. Это типично для юнит-тестирования (англ. unit testing), при котором тестируются только отдельные части системы. Оно обеспечивает то, что компоненты конструкции — работоспособны и устойчивы, до определённой степени. При тестировании белого ящика используются метрики покрытия кода.

При тестировании чёрного ящика, тестировщик имеет доступ к ПО только через те же интерфейсы, что и заказчик или пользователь, либо через внешние интерфейсы, позволяющие другому компьютеру либо другому процессу подключиться к системе для тестирования. Например, тестирующий модуль может виртуально нажимать клавиши или кнопки мыши в тестируемой программе с помощью механизма взаимодействия процессов, с уверенностью в том, все ли идёт правильно, что эти события вызывают тот же отклик, что и реальные нажатия клавиш и кнопок мыши. Как правило, тестирование чёрного ящика ведётся с использованием спецификаций или иных документов, описывающих требования к системе. Как правило, в данном виде тестирования критерий покрытия складывается из покрытия структуры входных данных, покрытия требований и покрытия модели (в тестировании на основе моделей).

1.2.3 Обоснования необходимости использования вычислительной техники для решения задачи

Таким образом, распределение проектов, их выполнение и контроль выполнения в настоящее время осуществляется вручную, без применения средств автоматизации. В этом случае многие аспекты проекта остаются без внимания компетентного руководителя, что приводит к ошибкам со стороны отдела и влечет за собой не только штрафы и неустойки, которые выплачивает компания клиентам

согласно договорам, но и к потере так называемого “goodwill”(с англ.: деловая репутация, престиж, авторитет). Более того, это приводит к еще большим затратам со стороны клиента, так как задержка сроков сдачи проектов означает упущенную прибыль, а также прямые издержки различного рода.

Графическая схема документооборота представлена на рисунке 1.10.

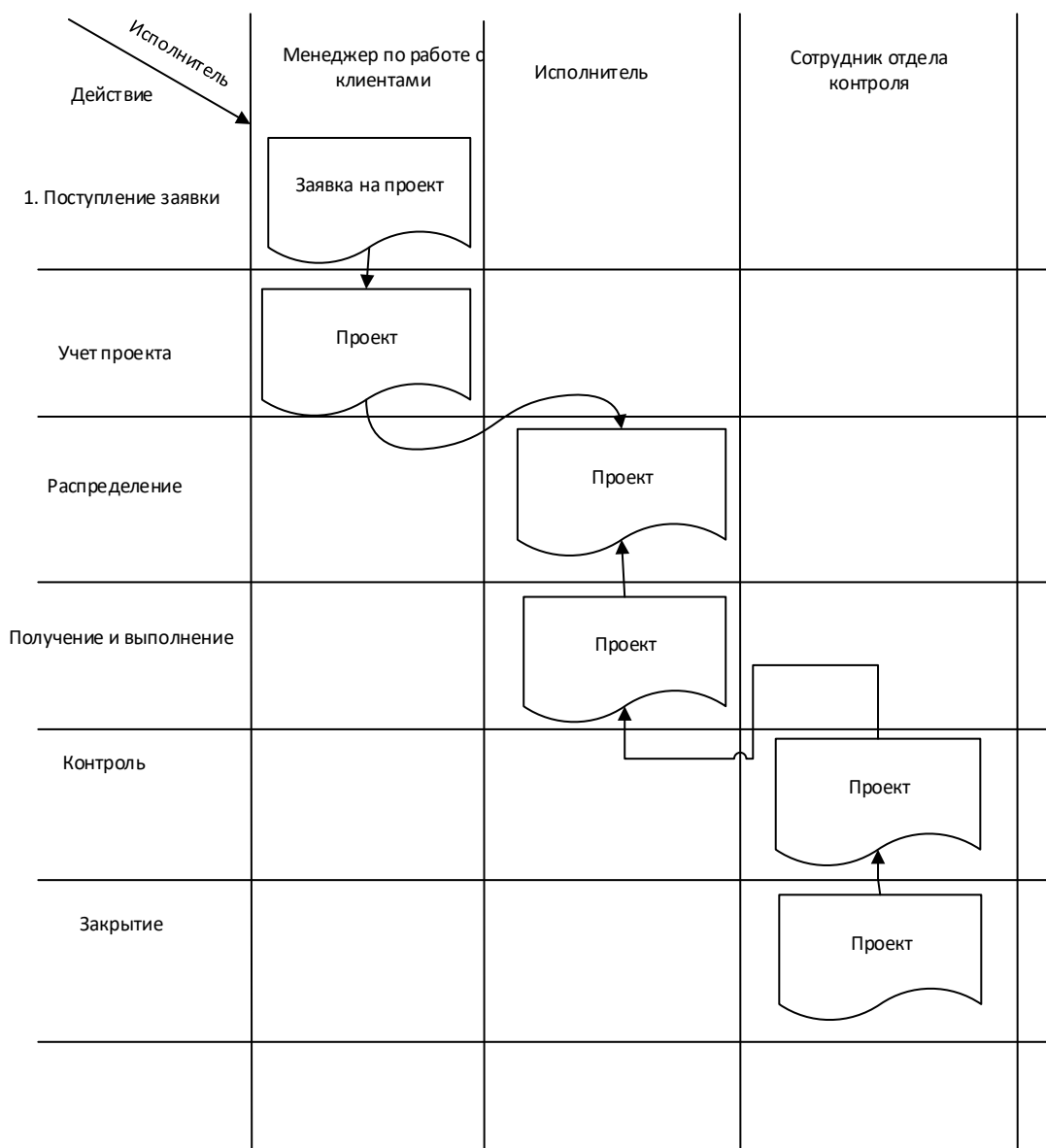


Рис. 3 Графическая схема документооборота.

Остановимся подробнее на проблемах, которые ввиду описанной выше ситуации необходимо решить компании ООО «Аутсорсинг Контракт»:

1. Отсутствие стандартного шаблона графика проекта;
2. Отсутствие понятной системы контроля над сроками реализации проекта;

3. Отсутствие единых методов анализа и прогнозирования хода реализации проекта;
4. Отсутствие стандартной отчетности по проекту;
5. Отсутствие контроля над каждой задачей по проекту.

Основной задачей информационной системы, которую необходимо разработать, является автоматизация процесса тестирования web-приложений.

В таблице 1.2 представлен объем потоков информации, обрабатываемых одним менеджером проектов за год.

Таблица 1.2

Объем потоков данных обрабатываемых документов

№ п/п	Наименование документа	Объем документа в символах	Объем документов в месяц	Объем в документах в год	Объем в символах в год
1	Данные о клиенте	29062	30	360	10462320
2	Заявка	16641	6	72	1198152
3	Проектная документация	2928	6	72	210816
4	План график	1131	15	180	203580
5	Список исполнителей	2159	120	1440	3108960
	Итого по входным документам	51921	177	2124	15183828
6	План распределения работ между специалистами	10359	1	12	124308
7	Отчет о выполнении	1590	6	72	114480
8	Акт выполненных работ	2444	6	72	175968
9	Ведомость выполненных работ	10142	1	12	121704
10	Аналитическая ведомость	1391	40	480	667680
	Итого по выходным документам	25926	54	648	1204140
	ИТОГО	77847	231	2772	16387968

Таким образом, можно сделать вывод, что выполняется очень большое количество рутинных операций, что может повлечь за собой увеличение ошибок. Возникает возможность ошибок пользователя при вводе данных и обработке данных. Кроме того, тратится время на проверку правильности составления ведомостей и отчетов. Следовательно, уменьшается оперативность работы.

Опираясь на вышеизложенные факты и учитывая, что объем информации имеет тенденцию к росту, можно сделать вывод о целесообразности использования средств вычислительной техники для решения данной задачи. Целью использования средств вычислительной техники при этом будет повышение оперативности решения задачи.

Результатом автоматизации работы менеджера является программа для учета распределения и учета выполнения проектов, которая способствует повышению качества и скорости обработки информации, тем самым снижая временные затраты, а также снижению трудовых и стоимостных затрат.

Данная программа должна отвечать следующим требованиям:

- Дружественный интерфейс;
- Отсутствие требований специальных знаний для эксплуатации;
- Сохранение целостности данных.

1.2.4 Анализ системы обеспечения информационной безопасности и защиты информации

Обеспечение информационной безопасности можно рассматривать как одну из важнейших задач для любой компании, работающей с информацией, разглашение которой может дорого стоить. Характерным в такой работе является то, что почти каждый человек понимает для себя важность защиты данных, но на практике лишь несколько понимают необходимость такой защиты и представляют реальные последствия и методы их предотвращения. У большинства же людей информационная угроза ассоциируется с сценами из популярных фильмов о «хакерах». На практике же все не так просто, ведь ИБ обязана учитывать множество факторов, которые так или иначе связаны с рассматриваемым объектом.

Каждый год появляется множество новых научных открытий, особенно заметно это в сфере информационных технологий. Реализуются новые технологии обработки данных, их хранения, передачи, благодаря этому растут и вычислительные мощности, направляемые на преодоление даже самых современных систем защиты. Продвижение таких направлений, как физика волн, радиоэлектроника, электромагнитные импульсы, позволяют изобретать все более

изощренные способы получения данных, при которых даже не устанавливается непосредственного контакта с компьютером жертвы.

Защита ресурсов обычно осуществляется на этапе их разработки, эксплуатации, производства или модернизации, а также в неразрывной цепочке ввода, обработки, передачи и хранения данных. Применение таких принципов позволяет дать возможность создавать СЗИ, у которой нет слабых звеньев как на отдельных этапах жизненного цикла КС, так и при работе любого элемента и выбранного режима работы КС. Однако сейчас в компании такая взаимосвязь отсутствует.

Механизмы защиты, которые активно применяются для создания систем защиты, должны быть увязаны с временем, местом и характером действий. Комплексность также предполагает реализацию различных методов и свойств защиты данных: программных, технических, криптографических, правовых или операционных. Поэтому даже простая СЗИ является комплексной.

Система защиты должна иметь несколько уровней, которые перекрывают друг друга, т.е. подобные системы строят обычно по типу «матрешки»:

- Контроль по периметру объекта;
- Контроль по периметру здания;
- Охрана внутри помещений;
- Защита аппаратных средств;
- Защита программных средств;
- Защита данных.

В рассматриваемом компании сейчас реализована только защита по периметру, а также частичная защита данных.

Многие источники угроз используют уязвимости для нарушения безопасности данных, получения некой выгоды (в виде нанесения ущерба собственнику данных). Кроме того, реальны и не злонамеренные действия источников угрозы, которые просто активируют уязвимости, наносящие непосредственный вред.

Каждой угрозе можно сопоставить разные уязвимости. Ослабление или полное устранение уязвимостей в значительной степени может влиять на реализацию угрозы безопасности данных.

Для решения проблем безопасности КС компании, описанных выше, требуется реализовать ряд задач:

- 1) Обновить требования в рамках ИБ в процессе разработки и внедрения ИС;
 - 2) Поддерживать соответствие норм и СЗИ в ИС положениям всех регулирующих документов по ИБ;
 - 3) Обновить правовую и нормативную базу поддержки ИБ, скоординировать работу по ЗИ;
 - 4) Поддерживать достоверность, полноту и оперативность прихода данных, а также поддерживать информационно процесс определения управленческих решений;
 - 5) Защитить от третьих лиц процесс работы ИС, обновить СиЗИ, ее форму, организацию, методики предотвращения и минимизации последствий ИБ;
 - 6) Предотвратить совместно с организационно-правовыми мерами и техническими СЗИ возможные НСД и незаконные посягательства на ИС от третьих лиц и персонала, не имеющего нужных полномочий;
 - 7) Записывать все события, которые могут влиять на ИБ, полный контроль и отчетность по каждой реализованной операции, совершённой внутри ИС;
 - 8) Выявлять, прогнозировать и оценивать источники угроз, условия и причины, которые помогают повышаться ущербу субъектов, нарушать корректное функционирование и повышать в ИС процесс изучения рисков реализации угроз, оценивания возможного ущерба, минимизации неприемлемых воздействий ИБ, реализации среды для уменьшения, возмещения и локализации возможного возмещения ущерба;
 - 9) Поддерживать доступность отката до актуального состояния ИС в рамках нарушения ИБ;
 - 10) Подготовить совокупную систему ИБ.
- Действия, необходимые для ЗИ, состоят из:
- 1) Правовой защищенности – законов, актов, процедур и правил, поддерживающих ЗИ в рамках правового поля;

2) Организационная защищенность – разделение деятельности компании и связей с исполнителями на правовой основе, что позволяет минимизировать или вообще исключить любой ощутимый вред;

3) Инженерно-техническая защищённость – применение разных технических средств, которые защищают от возможного ущерба бизнес-деятельности.

Для обеспечения комплексной информационной безопасности в ООО «Аутсорсинг Контракт» используется интегрированная система безопасности «Интегра-С».

Основные задачи системы следующие[8]:

- Дистанционное визуальное и электронное наблюдение за контролируемыми зонами объектов;
- комплексное управление системами активной защиты, оповещения и доступа на территорию объекта;
- автоматизация контрольно-пропускного режима;
- представление полной и достоверной информации о состоянии объекта;
- управление режимами работы объекта;
- передача информации оперативным дежурным на большие расстояния;
- защита от несанкционированного доступа к информации;
- контроль и фиксирование действий дежурного персонала.

Далее будут рассмотрены и предложены решения для различных областей внутренних и внешних элементов защиты.

SQL-инъекция является самым популярным средством взлома сайтов и программ, работающих с БД, основанная на внедрении в запрос некоего SQL-запроса.

Внедрение SQL, в рамках применяемой СУБД и условий внедрения, даст шанс атакующему выполнить отдельный запрос к БД (к примеру, прочитать содержимое всех таблиц, удалить, добавить или скорректировать данные), иметь доступ для чтения и/или записи отдельных файлов и реализации произвольных команд на исходном сервере.

Межсайтовый скриптинг, также известный, как XSS, является уязвимостью интерактивных ИС в сети. XSS возникает, когда в составляемые сервером страницы каким-то образом попадают пользовательские скрипты. Специфика таких атак состоит в том, что вместо самой атаки сервера они применяют уязвимый сервер для реализации атаки на клиента (подробнее на рис. 8).

Описанные выше XSS атаки являются не единственным способом кражи данных пользователя. ПК сотрудника может быть заражен вирусом или его данные могут перехватываться из его сети, с помощью анализатора трафика.

Поэтому нужно как можно сильнее затруднить возможность применения злоумышленником данных сессии другого пользователя. Известным методом для защиты от кражи сессионных данных стала проверка на смену IP-адреса в рамках единой пользовательской сессии. Но, как стало известно, DHCP у поставщиков услуг Интернет-соединения (провайдеров) зачастую настроен так, что IP-адрес в течение конкретного сеанса связи с Интернетом может меняться до нескольких раз за час. Да и статический IP-адрес не гарантия того, что пользователь будет тот же: злоумышленник может быть в той же локальной сети, что и атакованный.

Как уже упоминалось, инструменты web-разработчика ограничены стандартными данными, доступными средствами html, так что следующим шагом в развитии данной системы стала проверка данных, содержащих информацию о браузере пользователя (UserAgent), а также параметрах его экрана: разрешении, цветовой схеме и т.п., которые доступны инструментами JavaScript. Данное изменение позволило более стабильно отслеживать кражи сессионных данных, ведь изменение браузера или параметров экрана в рамках одной сессии одним пользователем очень маловероятно.

Совокупность подобных проверок уже сейчас внедрен в базовый алгоритм инициализации сессии пользователя. Проверка, в случае наличия MAC-адреса, почти полностью гарантирует безопасность сессионных данных, однако, даже в случае применения обычного браузера, что происходит зачастую, проверки стандартных данных чаще всего бывает достаточно.

Далее в данной системе может быть проверка IP-адреса пользователя, в случае его смены, на неизменность параметров, доступных через WHOIS, таких как: ID Интернет провайдера, город и т.д.

Стандартным средством определения считаются регистрационные данные пользователя: имя пользователя и пароль. Если эти данные у пользователя крадут, то есть риск безвозвратной потери контроля над аккаунтом.

1.3 Анализ существующих разработок и выбор стратегии автоматизации «КАК ДОЛЖНО БЫТЬ»

1.3.1 Анализ существующих разработок для автоматизации задачи

Система управления проектами (СУП) является совокупностью организационных и технических методик и инструментов, поддерживающих управление проектами в компании и позволяющих повышать эффективность их внедрения. Зачастую понятие СУП определяется более узко, как АИС или ИС управления проектами, т.е. как некая программа. Методическую и организационную составляющие обычно включают в термин корпоративной СУП.

qTrack – некая СУП с интересными возможностями взаимодействия - сервис полностью объединён с e-mail: описывать задачи теперь также легко, как по e-mail, а контролировать их выполнение максимально удобно, как в трекере. Основные возможности имеются бесплатно.

Мегаплан — это российская СУП, позволяющая устанавливать задачи и поручения, отслеживать их реализацию, хранить БД сотрудников фирмы, учитывать историю клиентов и т.д. [1]. Создатели ПО - одноименная компания Мегаплан.

ПО становится полноценным Groupware продуктом. Программа может работать как на личном сервере компании, так и на серверах поставщика продукта «Мегаплан» (SaaS). Сейчас это почти единственный российский программный комплекс, успешно продающийся как SaaS сервис. [2]

Worksection – отечественная онлайн-система для контроля проектов. Включает dashboard, задачи с описаниями, календарь, файловую систему, инструмент для учета времени, тэги. Есть поддержка SSL, пользовательского FTP и т.п. Программа простая и доступная.

Программное средство для решения оптимизационной задачи о назначениях в условиях стоимостных и временных ограничений по простым и комплексным проектам

Программа предназначена для решения оптимизационной задачи о назначениях в условиях стоимостных и временных ограничений по простым и комплексным проектам предназначена для определения оптимального варианта назначений в условиях стоимостных и временных ограничений по простым и комплексным проектам. Область применения программы - научно-исследовательская деятельность в области программно-целевого планирования управления развитием сложных технико-экономических систем. При проведении исследований программа может быть использована в качестве готового программного модуля в более сложных информационных системах поддержки принятия решений используемых для формирования долгосрочных программ и планов развития сложных технико-экономических систем.

Язык программирования: .NET Framework

Программное обеспечение для управления производственными проектами в ТЭК с учетом рисков

Программа предназначена для специалистов, занимающихся управлением сложными производственными проектами в ТЭК в условиях с высокой степенью неопределенности и ограниченности ресурсов. Позволяет моделировать различные варианты реализации проекта с учетом особенностей ТЭК в зависимости от объема и типа потребляемых ресурсов. Возможно рассчитать оптимальный объем необходимых для реализации проекта ресурсов различных типов, который позволит достичь заданных значений характеристик проекта и минимизировать риск невыполнения проекта в установленный срок.

Язык программирования: General Purpose Simulation System (GPSS)

Система управления проектами ЕКС

Программа представляет собой WEB-приложение для групповой работы, реализующее возможности электронного документооборота, единого информационного портала и системы управления проектами. Программа основана на базе системы Microsoft Sharepoint и позволяет каждому

подразделению создавать свои информационные разделы с публичной или закрытой информацией. Тип ЭВМ: IBM PC-совмест. ПК X64. ОС: Microsoft Windows Server.

РКИТ: Управление проектами

Программа предназначена для управления проектами на всех этапах: от планирования проекта до анализа результатов. Область применения: работает с любым кастомизированным решением на Docsvision версии 5.4 и выше. Функциональные возможности – программа позволяет: сформировать план проекта, разбить на этапы и задачи, назначить исполнителей; использовать шаблоны для задач и документов проекта; фиксировать риски; управлять проектом в реальном времени: через панель управления и диаграмму Ганта контролировать исполнение, сроки; изменять сроки, задачи, участников. Программа предоставляет инструменты для работы с задачами в почте и веб-браузере, вычисляет реальную трудоемкость задач и проекта, позволяет анализировать причины отклонения факта от плана, формирует отчеты по трудоемкости, исполнению проекта, портфелю проектов.

Язык программирования: C#

Система управления проектной деятельностью (СУПД)

Система управления проектной деятельностью (СУПД) предназначена для: формирования и согласования отчетности по проекту; формирования рейтинга проектов и проектных команд; формирования отчетов об исполнительской дисциплине сотрудников, вовлеченных в проектную деятельность; ведения документооборота по проекту; работы с задачами и поручениями; управления изменениями и мониторинга реализации и проекта; ведения реестра рисков проектов; календарного планирования; ведения бюджета проектов с детализацией по периодам и источникам финансирования; коммуникации между участникам и проектной деятельности; мониторинга выполнения контрольных точек и влияющих на их выполнение задач и поручений в режиме онлайн.

Язык программирования: C# с использованием .NET Framework, JavaScript

Сравнение разработок по основным параметрам представлено в таблице

1.3.

Таблица 1.3

Сравнение систем управления проектами

Параметры	qTrack	Мегплан	Worksection	Claris	teamtools	ПланФикс
Объединение сотрудников в группы	+	-	+	-	+	-
Подключение клиентов в систему	+	+	-	+	-	+
Система оповещений по e-mail	+	+	+	-	+	-
Система оповещений по jabber	+	-	+	-	+	-
Внутренняя система оповещений	+	-	+	-	+	-
SSL-шифрование	+	+	-	+	-	+
Общее количество баллов	3.5	5	3.5	3.5	3	6
Наличие сущности "Проект"	+	-	+	-	+	-
Разграничение доступа к проектам	+	+	-	+	-	+
Диаграмма Ганта	+	+	+	-	+	-
Наличие сущности "Задача"	+	-	+	-	+	-
Вложенность задач	+	-	+	-	+	-
Разграничение доступа к задачам	+	+	-	+	-	+
Возможность указать точную дату выполнения	+	+	+	-	+	-
Возможность указать точное время выполнения	+	-	+	-	+	-

Статусы состояния задач	+	-	+	-	+	-	
Повторяющиеся задачи	+	+	-	+	-	+	
Шаблоны задач	+	+	+	-	+	-	
Оповещение о назначенной задаче	+	-	+	-	+	-	
Оповещение о просроченной задаче	+	-	+	-	+	-	
Уведомление в настраиваемое пользователем время	+	+	-	+	-	+	
Учет рабочего времени (Тайм-трекинг)	+	+	+	-	+	-	

Рассмотренные решения выполняют свойственные такого рода системам задачи, однако имеют один общий недостаток – избыточность функций и сложность внедрения и дальнейшего администрирования. Поэтому необходимо разрабатывать систему управления проектами самостоятельно, с учетом всех требований внутреннего заказчика.

1.3.2 Выбор и обоснование стратегии автоматизации задачи

Стратегия автоматизации изначально должна удовлетворять приоритетам и стратегии (задачам) бизнеса и выражать пути достижения этого соответствия.

Хаотичная автоматизация становится одним из наиболее неактуальных видов вложения средств в развитие компании. Под хаотичностью процесса в таком случае подразумевается отсутствие плана развития. Обычно при таком подходе процесс применения информационных технологий выражается мимолетными отдельными задачами, и не связан с актуальными потребностями бизнеса. В качестве критериев принятия решений при этом выступают: уровень осведомленности и предпочтений лиц, принимающих решения, доступность покупки сейчас с выгодной скидкой какую-либо технику или ПО и т. д. Обычно в результате компания в лучшем случае получает раздробленные прикладные

системы, стоимость внедрения которых в ряде за частую больше стоимости комплексного решения.

Помимо этого, существует несколько стратегий оптимизации, а именно:

1) Кусочная (хаотичная) автоматизация, которая представляет из себя один из самых неэффективных методов инвестирования средств для подъема компании. Под самой автоматизацией обычно понимают отсутствие плана дальнейшего развития и создания. И такой подход обычно характеризуют сиюминутные внедрения информационных технологий, которые очень часто не соответствуют реальным потребностям бизнеса. И в таком случае само предприятие получит прикладную систему, но ее стоимость будет сопоставима в качественным комплексным решением. Выбирать данную стратегию можно лишь в том случае, если нужно сэкономить средства и время на внедрение системы. Но все дальнейшие шаги к модернизации будут стоить значительно дороже и станут нерациональными и экономически невыгодными.

2) Другой вид автоматизации – по участкам. Она предполагает обновление отдельных производственных или управленческих линий, которые сопоставимы по функциональным особенностям. Данный метод применяется в том случае, если у компании нет лишних средств на автоматизацию в полном объеме, а выбранные участки дают хороший экономический рост, поэтому применяется она часто на производственных участках.

3) Третий вид – автоматизация по направлениям, которая предполагает полную автоматизацию отдельных видов деятельности компании. Основной отличительной особенностью от автоматизации по участкам является тот факт, что вся деятельность предполагает охват абсолютно всех подразделений и служб, задействованных в автоматизируемом направлении. Финальным этапом этого метода часто является полная автоматизация компании.

4) И последний вариант – полная автоматизация, предполагающая некую систему, состоящую из множества элементов разного уровня и значения. В такую систему обычно входят другие подсистемы, модули, функции, блоки управления, текущие задачи и т.д. Такое построение позволяет использовать один и тот же алгоритм для просчета разных задач, а также исключается дублирование данных из разных источников, поскольку все находится итак в одной системе.

Основным минусом полной автоматизации можно выделить необходимость большого количества затрат на реализацию такой схемы, а также высокий уровень первоначального планирования и расчетов.

5) Полная автоматизация управления компанией.

6) АСУП как система включает в себя большое количества элементов разных уровней и разного назначения. К ним можно отнести модули, задачи, подсистемы, управленческие процедуры, функции, операции, блоки управления и т. п. Стандартные системы типа ERP обычно представляют собой иерархические структуры, включающие в себя элементарные управленческие процедуры, используемые для включения в АСУП.

7) Внедрение включает в себя объединение и утверждение управленческих функций и процедур, чтобы в рамках управления компанией реализовывалась оптимизация его поведения.

8) Внедрение проявляется в каждой обеспечивающей и функциональной системе без исключения.

9) АСУП, разработанные на основе одной базовой системы ERP, выводят компанию на новый уровень объединения организационного обеспечения при помощи упрощения пользовательского интерфейса. Особенно ощутим подобный эффект в больших АСУП, в которых новая система приходит на смену нескольким локальным. Практическим результатом перехода к единой системе становится единый для всей компании стандарт на варианты взаимодействия пользователей с системой.

10) Но основная задача, для которой и внедряются автоматизированные системы в компании — это функциональная интеграция.

Проанализировав данные варианты стратегий, была выбрана стратегия по направлениям, так как это наиболее подходящий вариант для рассматриваемого предприятия.

1.3.3 Выбор и обоснование способа приобретения ИС для автоматизации комплекса задач

Способы приобретения ИС – это комплекс действий от определения и формализации решения о важности ИС до момента внедрения данной системы на предприятии. Выделяют несколько способов приобретения информационных систем, а именно:

- Самостоятельная или заказная разработка;
- Покупка готовой ИС (зарубежной или отечественной);
- Покупка + доработка (как самостоятельно, так и на заказ);
- Стандартная аренда.

А теперь внимательно рассмотрим каждый вариант на предмет его применения к автоматизации некоторой поставленной задачи.

Если выбирается покупка готового продукта, то необходимо выбрать такое решение, которое будет полностью удовлетворять всем бизнес-процессам компании. Сегодня это сделать очень сложно, поскольку практически все производители стараются угодить всем и просто перегружают свой софт совершенно ненужным функционалом, за который все равно нужно платить. Либо готовое решение представляет только каркас, на котором уже строится необходимый функционал для потребителя. Оба данных решения никак не подходят для реализации задачи, поскольку подразумевают большие вложения денежных средств в саму систему автоматизации.

Приобретаемая система не подойдет компании и из-за нехватки функциональности, в ней хуже реализовано будущее развитие ИС, будет необходим длительный срок адаптации под конкретные нужды компании, а также неудобства будут вызывать частое обращение в службу поддержки.

Другой вариант – заказ у сторонних разработчиков, который позволит получить ИС, подходящую под бизнес-правила компании, но это только на текущий момент. В случае каких-то изменений бизнес-процессов придется вновь преобразовывать систему, что выльется в лишние затраты на модернизацию ИС.

Вариант с покупкой и доработкой также возможен, но в случае, если требуется небольшая доработка без изменения самого ядра ИС, которое иногда проблематично настроить из-за неполной документации.

Вариант с арендой ИС имеет сразу несколько больших преимуществ:

- Открывает новые инновационные возможности за счет сотрудничества с поставщиками интеллектуальных услуг мирового уровня, а также дарит ценный опыт во взаимодействиях и партнерстве;
- Дает компании возможность быстро перестраиваться под изменяющиеся реалии рынка – всегда проще найти другого поставщика, чем закупать новое оборудование, перестраивать внутреннюю деятельность и ликвидировать старые мощности;
- Ускоряет получение новых навыков и ресурсов.

Однако в этом методе есть и свои нюансы – компания рискует вывести слишком много видов деятельности за свои пределы и в случае нестабильной ситуации может утратить то, что приносило ей популярность и успех на рынке.

Поэтому наиболее оптимальным решением становится то, что ИС будет разрабатываться в стенах компании собственными силами, поскольку компания располагает квалифицированными специалистами для разработки и внедрения подобного вида систем. Но все-таки разработка такого рода предполагает, что у компании имеются специалисты, способные провести анализ бизнес-процессов, а следом разработать и внедрить систему для их автоматизации. А поскольку автоматизация затрагивает одно достаточно большое направление, всем специалистам не составит труда провести работы по проектированию, разработке и финальному внедрению системы автоматизации.

Создаваемая ИС будет включать в себя весь необходимый функционал, а также будет учитывать особенности работы и пожелания сотрудников компании, следовательно будет являться уникальным продуктом, что позволит быстро изменить в случае чего функциональность по требованию заказчика.

1.4 Обоснование проектных решений

1.4.1 Обоснование проектных решений по информационному обеспечению

Зачастую для информационного обеспечения имеется совокупность требований, а именно:

- ИО обязано стать полноценным и адекватным для поддержки требуемого числа функций;
- В процессе кодирования данных важно использовать проверенные ID;
- В момент кодирования выходных или входных данных, используемых на высшем уровне управления, применяются только определители данного уровня;
- ИО зачастую полностью совместимо с реализуемой системой;
- Формат документов должен сочетаться со стандартами фирмы и построенной системой документации;
- Вид визуальных форм и всех документов обязан отвечать требованиям для любого рабочего места сотрудника;
- Все используемые графики, сокращения и сообщения важно использовать общепринятые, характерные для предметной области, утвержденные самим заказчиком;

База данных как основа информационного обеспечения обычно включает в себя:

- Справочники, поддерживаемые методом работы с программой клиентами системы, которые работают в ней;
- Созданные таблицы для работы системы в конкретных условиях, начальная загрузка которых связана с установкой системы, и данные, которые могут корректироваться администратором системы;
- Таблиц текущей и уже имеющейся информации, доступ к которым производится при помощи ПО системы с наличием требуемых прав.

Обмен данными между совокупностью задач, которые уже включены в систему, а также сторонними системами выполняется в границах функций, которые встроены в систему.

В проектном решении используются сразу несколько принятых у заказчика показателей: общегосударственный и местный. По первому классификатору проходят все ГОСТы, которые есть в базе системы. Важным требованием также становится генерация локальных показателей. При этом важно применять справочники: иерархические и линейные. В линейных справочниках все

компоненты, которые уже находятся в составе, выделены в виде классических списков, а в иерархических – представлены в виде списков с иерархической компоновкой.

Опишем требования к системам кодировки и классификации данных. Системы классификации (СК) часто делят на иерархические и многоаспектные. Первый тип СК состоит из разбиения полученного множества на разные подмножества, которые имеют отношения подчинения друг другу. Второй тип СК применяет изначально сразу несколько независимых описательных признаков, т.е. стартовое множество определено сразу в разных аспектах. [11] Т.к. данные не нужно разделять на несколько признаков, в проектном решении можно применять иерархическую систему.

Электронная форма документа – это некий шаблон с пустым пространством, которое оставлено для самостоятельного заполнения пользователем. Подобные формы включают в себя командные кнопки, меню, формы и поддерживают разнообразные виды входных данных. Важно понимать, что для реализации подобных форм необходимо узконаправленное ПО. Определим часть форм:

- Формы для сбора данных, переноса их в БД с возможностью обработки (перемещение и сбор по e-mail);
- Формы для получения и сбора данных внутри компании и за ее пределами (заказы на покупку, отчеты о продажах, счета).

Интегрированная ИБ – объединение взаимосвязанных данных с малой избыточностью, допускающей их применение оптимальным образом для большинства приложений.

СУБД поддерживает совместимость данных, минимизацию синтаксической и семантической избыточности, совмещенность данных и реального состояния объекта, разделённое хранение информации у пользователей. Нюансом можно отметить требование повышенного контроля вводимых данных и разделение доступа к данным.

Для приспособления экономических данных для улучшенного поиска, отправки и анализа по каналам связи, их переводят в цифровые символы. Поэтому

изначально их разделяют (классифицируют), затем переводят в код при помощи классификатора.

Классификатор – это подчиняющееся однозначному правилу создания цифровой или буквенный код, причитав который, можно определить, к какому виду относится информация, какой объект используется, в каком порядке обрабатывается и для чего предназначен.

Классификатор может действовать как в рамках определенной информационной системы (тогда он называется внутренним), так и в пределах отрасли или всего государства (тогда это отраслевой или государственный классификатор соответственно).

Для внемашиного информационного обеспечения основной его частью является система классификации используемой документации.

В состав системы включены перечень документов с указанием разделов и при необходимости подразделов, к которым может быть отнесен документ. При использовании четкой структуры документации значительно облегчается работа с ней, в том числе эксплуатация, поиск, обновление или редактирование.

Основная характеристика всего потока документов в современных автоматизированных информационных системах состоит в том, что документы обладают практически одинаковой формой, что часто может приводить к возникновению хаоса, большой трудоемкости достижения нужных результатов, в результате чего обработка такой документации производится очень долго и часто с небольшим успехом. Поэтому необходимо использовать единый стандарт организации документооборота, в том числе и при использовании электронных систем документооборота.

Основной составляющей внемашиного ИО становится система документации, которая применяется в рамках контроля экономического объекта. Сам документ представляет из себя сочетание сведений, которые используются в рамках решения экономических задач, и находится на отдельном носителе в рамках утвержденной фирмы.

Система документации становится сочетанием некоторых форм документов, используемых в рамках контроля экономическим объектом. Суть

системы экономической документации заключается в огромном числе видов документов.

Новейшие системы документации, которые применяются в ручных ЭИС, определены большой вариативностью форм документов; многочисленным потоком документов и их запутанностью; разделением данных в документах, и исходя из этого всего – малой актуальностью получаемых результатов. Анализ данных в таких ИС отнимает почти все свободное время сотрудника. Для упрощения системы документации часто используют ряд подходов:

- Использование единого стандарта для всех документов;
- Переход к ЭДО.

Входные документы для анализируемой задачи лучше получать через сеть, поэтому они обязаны иметь четко выверенную структуру. Для упрощения работы с подобными данными, они представляются в виде таблиц (создать это не так проблемно, т.к. документы уже содержат готовые структуры и могут быть изменены).

К внутримашинному ИО относят описание форм в электронном виде.

В рамках подготовки структуры экранных форм для передачи форм начальных документов с текущими данными лучше всего использовать комбинированную форму, которая походит на ту, что применялась для реализации самого документа. Местоположение полей должно идти так, как оно будет удовлетворять логической структуре документа и файлов с данными, которые минимизируют сложность операции загрузки в БД.

В рамках создания экранных форм для документов с конкретными данными важно понимать, что эти макеты используются для вывода и обновления записей ИБ, и в рамках их создания лучше применять анкетную форму реквизитного расположения, которая подойдет для данных операций.

Основной составляющей немашинного ИО становится система документации, которая применяется в рамках контроля экономического объекта. Сам документ представляет из себя сочетание сведений, которые используются в рамках решения экономических задач, и находится на отдельном носителе в рамках утвержденной фирмы.

Система документации становится сочетанием некоторых форм документов, используемых в рамках контроля экономическим объектом. Суть системы экономической документации заключается в огромном числе видов документов.

Новейшие системы документации, которые применяются в ручных ЭИС, определены большой вариативностью форм документов; многочисленным потоком документов и их запутанностью; разделением данных в документах, и исходя из этого всего – малой актуальностью получаемых результатов. Анализ данных в таких ИС отнимает почти все свободное время сотрудника. Для упрощения системы документации часто используют ряд подходов:

- Использование единого стандарта для всех документов;
- Переход к ЭДО.

Входные документы для анализируемой задачи лучше получать через сеть, поэтому они обязаны иметь четко выверенную структуру. Для упрощения работы с подобными данными, они представляются в виде таблиц (создать это не так проблемно, т.к. документы уже содержат готовые структуры и могут быть изменены).

Отраслевые классификаторы в основном используются для каких-то процедур, связанных с обработкой информации внутри конкретной отрасли между двумя и более организациями.

Локальные классификаторы используют в пределах отдельных предприятий.

В данном проекте должны быть спроектированы следующие локальные классификаторы:

- сотрудники компании;
- заказчики;
- проекты;
- задачи;
- специализации сотрудников.

Разрабатываемая система должна содержать в себе следующие модули:

- модуль авторизации;
- модуль работы со справочниками;

- модуль ввода данных;
- модуль получения отчетов;
- модуль поиска.

1.4.2 Обоснование проектных решений по программному обеспечению

Программное обеспечение (ПО) — совокупность программ для реализации целей и задач автоматизированной системы. [2]

Самую значимую часть информационного обеспечения, как правило, составляет база данных, которая так или иначе присутствует в любой информационной системе. Именно в базе данных хранится вся информация, связанная с работой, то есть вводом, обработкой, выводом информации в описываемой информационной системе. База данных должна иметь возможность ее изменения под возникающие нужды обработки информации, то есть должна иметь способность гибко меняться и обеспечивать необходимый уровень анализа хранящихся в ней данных и формирования необходимых отчетов с помощью запросов.

В состав базы данных входят:

Таблицы со справочной информацией, то есть такой, которая записывается один раз и далее используется уже в созданном виде. Этот факт не отрицает возможности изменения, дополнения, редактирования или удаления каких-либо данных из справочной таблицы.

Таблицы с начальной информацией, необходимые для старта или начала работы системы. Эти таблицы могут быть связаны с авторизацией пользователей, содержать параметры начальной настройки информационной системы и аналогичную информацию.

Таблицы с оперативной информацией, то есть такой информацией, которая постоянно используется в работе информационной системы и которые наиболее часто подвержены изменению хранящихся в них данных.

Доступ ко всем таблицам, как правило, регулируется настройкой прав доступа.

ПО делится на два вида: общее (операционные системы, операционные оболочки, компиляторы, интерпретаторы, программные среды для разработки прикладных программ, СУБД, сетевые программы и т.д.) и специальное (совокупность прикладных программ, разработанных для конкретных задач в рамках функциональных подсистем, и контрольные примеры). [2]

Программные средства общего назначения должны удовлетворять следующим требованиям [1] :

- представлять собой набор взаимосвязанных программ с нормализованными методическими указаниями и инструкциями по использованию обслуживающих программ на всех стадиях реализации расчётов комплексов задач;
- строиться по принципу открытого программного обеспечения с целью дальнейшего развития и совершенствования;
- управлять устройствами комплекса технических средств и организовать очередность решения задач;
- осуществлять работу программ самоконтроля и иметь группу диагностических тестов;
- обеспечивать защиту данных и программ пользователя друг от друга и от операционной системы;
- обеспечивать удобное хранение данных и обращение к любой совокупности данных;
- обеспечивать надежность хранения информации при сбоях ЭВМ;
- проводить анализ аварийных ситуаций, определять их причину, сообщать о ней и предлагать различные варианты действия пользователю.

Операционная система и базовое ПО.

Разрабатываемый программный продукт имеет клиент-серверную архитектуру.

Архитектура клиент-сервер основана на распределении функций между двумя типами независимых и автономных процессов: серверами и клиентами. Сеть связывает воедино серверы и клиенты, предоставляя средства связи.

Если вся обработка данных происходит на стороне сервера, а клиент выполняет только функции интерфейса с пользователем, то клиентское

приложение называют «тонким» клиентом. Если часть обработки данных происходит на стороне клиента — то «толстым» клиентом.

Архитектура клиент-сервер включает в себя три основных компонента:

Клиенты. Клиент представляет собой любой процесс компьютера, который запрашивает сервис от сервера. Клиент также называется интерфейсным приложением. Клиентский процесс, базируется на графическом интерфейсе пользователя.

Серверы. Сервер — это компьютерный процесс, предоставляющий сервис клиентам. Сервер также называют серверным приложением. Серверный процесс характеризуется независимостью от местоположения, оптимизацией использования ресурсов, масштабируемостью и способностью к взаимодействию с другими системами[12].

Для корректного взаимодействия компонентов клиент-серверной архитектуры между собой требуется их соответствие некоторым основным правилам. Эти правила должны в равной степени выполнять и клиенты, и серверы, и ППО.

Аналогично техническому обеспечению программное обеспечение отличается для клиентской и серверной части.

Выбор сетевой операционной системы (СОС) во многом зависит от технической платформы вычислительных средств.

В Таблице 1.4 приведена сравнительная таблица трех распространенных систем управления базами данных, конкурирующих на рынке программного обеспечения по основным показателям.

Таблица 1.4

Сравнение СУБД

Показатели	Microsoft SQL Server 2018	баллы	MySQL 8.0	баллы	PostgreSQL 14	баллы
------------	------------------------------	-------	-----------	-------	---------------	-------

Поддерживаемые операционные системы	Windows Desktop/Server	3	Windows Desktop/Server, Linux, Unix, Mac	5	Windows1 Desktop/Server, Linux, Unix, 2Mac	5
Поддержка даты и времени	Да	3	Да (но без временной зоны)	2	Да	3
Аутентификация	Средствами БД и Active Directory	3	Средствами БД	2	Много разных методов, включающих предыдущие	1
Производительность планировщика запросов для сложных запросов	Средняя (умеет параллельные запросы «из коробки»)	1	Очень хорошая	5	Плохая	1
Итого		10		14		10

На сегодняшний день разработаны десятки различных СУБД.

В качестве возможных вариантов рассматриваются MySQL против Firebird.

MySQL - сервер, который работает с MYISAM и InnoDB таблицами. Как известно, InnoDB, это научная работа Хейки Туури в Хельсинском универе (1994-2000) в области высокопроизводительных технологий БД.

В серверах на Linux, mysql это практически стандарт де факто. А Firebird, де факто, БД для Windows в небольших корпорациях стран СНГ. А вот за границей его применяют в основном как БД накопления и хранения данных в пользовательских приложениях удаленных сотрудников и малопользовательских CRM.

Оптимальным вариантом является MySQL. Данная программа не сильно нагружает систему. Данная СУБД широко распространена, поэтому будет не сложно найти того, кто сможет ее обслуживать

На основании вышеприведенных доводов решено в качестве СУБД MySQL.

При выборе программного обеспечения для разработки будем рассматривать следующие характеристики: надёжность; эффективность; стоимость; понятность пользователю; простота настройки; простота оформления экранных и печатных форм.

Проведем сравнение PHP и Perl

Преимуществами PHP по сравнению с популярным языком Perl, является простой синтаксис, что делает его легче в изучении, и более четкий и менее раздутый исходный код, что делает его легче при разработке модулей. PHP в большей степени ориентирован на web-программирование и не сложен в изучении. Недостатком PHP 3 является его низкая по сравнению с Perl производительность на сложных приложениях при обработке больших скриптов. Но в версиях PHP 4 и 5 эта проблема решена. PHP доступен для большинства операционных систем современных web-серверов.

В качестве языка программирования интерфейсов разрабатываемой системы наиболее подходящим представляется PHP. Основными факторами, послужившими в пользу выбора PHP, являются:

1. PHP «технически» выигрывает среди рассматриваемых языков программирования.
2. PHP имеет огромную популярность.
3. PHP имеет возможность функционирования с элементами систем: HTTP-протоколом, языком HTML и встроенной поддержки СУБД.
4. Для PHP характерна доступность, благодаря его невысокой стоимости.

1.4.3 Обоснование проектных решений по техническому обеспечению

Техническое обеспечение (ТО) является совокупностью технических средств (ТС), необходимых для работы ИС, а также нужная документация на эти средства и процессы производства.

Комплекс ТС включает в себя:

- ПК всех моделей;
- Механизмы сбора, анализа, отправки и хранения данных;
- Механизмы отправки данных и линии связи;

- Периферия и средства автоматического съема данных;
- Канцелярские принадлежности.

Для создаваемой ИС нужно такое аппаратное обеспечение:

- ПК для работы бухгалтера;
- ЛВС;
- Сервер БД;
- Принтеры (МФУ).

Сейчас параметры установленных ПК удовлетворяет требованиям к ПК ИС. Все ПК объединены в единую сеть, оснащены выходом в Интернет. На каждом рабочем месте имеется лазерный принтер.

2 Проектная часть

2.1 Разработка проекта автоматизации

2.1.1 Этапы жизненного цикла проекта автоматизации

ЖЦ является непрерывным процессом, который изначально начинается с момента выражения решения о необходимости его создания и заканчивается после факта изъятия из обращения.

Среди самых популярных стандартов часто выделяют такие:

- ГОСТ 34.601-90 – используется в АИС и устанавливает все этапы и стадии их создания. Также он описывает содержание работ для любого этапа. Стадии и этапы, закрепленные внутри стандарта, больше всего соответствуют каскадной модели ЖЦ;
- ISO/IEC 12207 – стандарт, указывающий процессы и организацию ЖЦ. Может использоваться в любом виде заказного ПО. Стандарт не имеет описания стадий и этапов;
- Custom Development Metho – технологический материал по созданию прикладных ИС, детализированный до уровня заготовок проектных решений, которыми пользуются в проектах с использованием средств Oracle. Применяется CDM для классической модели ЖЦ (есть все этапы и описанные задачи), а также в процессе «оперативной разработки» или «облегченного прохода», которые применяются в рамках малых проектов;
- Rational Unified Process (RUP) – применяет интерактивную модель разработки, включающую 4 фазы: старт, изучение, создание и внедрение. Каждая из фаз может делиться на этапы, которые в итоге создают версию для внутреннего или внешнего применения. Завершение всех 4 фаз – это цикл разработки, и по завершению одного цикла генерируется новая версия системы. Если по факту проект продолжается, то сам продукт тоже изменяется и проходит вновь эти 4 фазы. Суть работы при использовании RUP – создание и поддержка моделей на базе UML;
- Microsoft Solution Framework (MSF) – аналогичен RUP, тоже состоит из 4 фаз: изучение, разработка, реализация и проверка, считается итерационным

и включает применение объектно-ориентированных моделей. MSF в отличие от RUP чаще всего используется для создания бизнес ПО;

- Extreme Programming (XP) – экстремальное программирование (современная методология, создана в 1996 году). Ее суть составляют командная работа, постоянная коммуникация с заказчиком в рамках всего проектирования ИС, создание проекта с использованием последовательно оптимизируемых прототипов.

Перед началом самой разработки проекта внедрения универсальной методики контроля, важно понять цели внедрения. Фирмы, решающие воспользоваться подобными методиками, зачастую преследуют несколько целей: рост скорости реакции фирмы на изменение внешней среды, минимизация цены продукта, устранение различных удерживающих факторов в бизнес-процессах фирмы, рост эффективности командной работы и т.п.

Среди явных проблем процесса создания проекта можно выделить такие:

- Корректировка требований относительно процесса создания;
- Разделение ответственности за реализуемую работу и ее результат;
- Доступность потока мелких, срочных, приходящих требований, отвлекающих программистов и управленцев от базового направления работ.
- Срыв срока, повышение расходов, потери качественного уровня.

Чтобы реализовать успешную организацию процесса создания, подготовлена универсальная процедура создания проектов.

Модели, методики и подходы универсального управления проектами имеют свое развитие в рамках сложных технических проектов, которые заключаются в реализации больших программных комплексов. Универсальное управление можно рассматривать как некую платформу, которая находится в рамках нескольких методик контроля инновационными, а самое важно, ИТ проектами. Базовая суть универсального управления проектами описана в работах Дж. Хайсмита [1], Г. Аллемана [8], Г. Чина [11]. Сравнение разных прикладных методологий универсального управления новейшими проектами также есть в трудах К. Лармана [27] и П. Абрамсона и других [5]. Моментный подход к

определяю наилучшей методики универсального управления проектами указан в работе А.С.Коха [3].

Дж. Хайсмит выделяет базовое качество данного подхода так: «Гибкость (agility) - это возможность сразу и создавать, и отвечать на корректировки, создавая прибыль в изменчивых условиях экономики. Гибкость становится способностью держать баланс между стабильностью и хаосом» [3]. Также он пишет, что: «Часто многие склонны верить, что подвижность говорит об отсутствии структуры. Но такое отсутствие порождает хаос. Также, избыток некой упорядоченности несет за собой жесткость. Теория сложности описывает то, что процесс создания инновации методами, которые не определены заранее, часто реализован в точке соприкосновения порядка и хаоса, гибкости и стабильности. По словам ученых, реализация нового возможна на границе хаоса [2]. Но нахождение такого баланса между порядком и хаосом и будут направлены все возможности универсального управления проектами. Причем следует оно от упорядоченности процессов, инструментов и методик, которые есть в традиционных школах проектного управления в рамках деконструкции, деструктуризации множественных упорядоченных реализованных условий протекания проектов, к выработке понятных и логичных (направленных на персонал методов и инструментов, которые имеют возможность плавно приспособиться к изменчивым условиям). Если классическими параметрами управления проектами описывались планирование, улучшение и отслеживание, то универсальное управление проектами основными усилиями будет считать естественную эволюцию и приспособление.

В 2001 году основатели и последователи части близких по духу и сути методик управления ИТ-проектами создали свод правил универсального контроля проектами. Данный свод описал 4 базовые идеи и 12 параметров [17]. Создатели свода правил, почитатели методик экстремального программирования, скрама, адаптивного создания проектов и т.п., сознательно не стали сводить универсальное управление проектами к моделям, инструментарию и средствам, но отразили его в виде неких адаптационных идей и принципов, которые нужны для понятного творческого воплощения в контексте отдельной ситуации. Суть и

идеи универсального проектного управления не нужно сводить к сложившимся законам и правилам.

Базовая основа универсального управления заключена в:

- Персоналии и их работа важнее, чем инструмент и процесс;
- Применяемое ПО (в общем случае - ценный для клиента продукт или реализуемая услуга) важнее, чем собранное документальное сопровождение (или контроль плана бюджета);
- Работа с заказчиком важнее, чем обязательства по контракту;
- Реакция на коррективы важнее, чем простое следование плану.

Суть универсального управления проектами состоит в:

- Удовлетворенность клиента благодаря оперативной и надежной разработки продукта, который имеет значение для клиента;
- Позитивное отношение к корректировке требований к продукту, даже на финальной стадии, если это имеет значимую ценность для клиента и ведет к росту конкурентных качеств продукта;
- Создание и поставка отдельно работающих модулей или обновленных версий ПО (ежемесячно, еженедельно или чаще);
- Все участники проекта, в особенности создатели и вдохновители, должны без проблем поддерживать конкретный темп работы на требуемый срок;
- Рост технического мастерства исполнителей и улучшение самого продукта;
- Переход к простоте, чтобы не реализовывать лишнюю работу;
- Самоорганизация и автономия на уровне команды проекта, оптимальные тех. требования, дизайн и архитектура реализуются у лучше организованной команды;
- Доступность изменений при смене обстоятельств.

Принцип схемы универсального управления проектами признается циклическая модель ЖЦ проекта, которая разбивает проект на несколько процедур. «Каждая процедура выглядит как некий программный файл в миниатюре, и состоит из задач, которые нужны для реализации мини-прироста по скорости работы: изучение требований, составление плана, проектирование, написание кода, проверка, документирование. Каждая отдельная процедура зачастую не так оправдана для реализации обновлённой версии продукта, тут

понимается, что сам по себе проект готов к реализации по факту каждой процедуры. В рамках окончания каждой процедуры команда проводит переоценку основных задач разработки» [14].

Модель описывает, что проект проводится некоторое количество циклов (процедур), и любой цикл имеет 4 фазы – определение требований, подготовка проекта, реализация и оценивание. Любой следующий цикл ведет к корректировке требований, изучению содержания, оптимизации продукта и процессов создания проекта. В реальности процедурная природа универсального проекта сложнее, и предполагает доступность перехода с этапа на этап, что часто отмечается в виде названной хаотической модели ЖЦ проекта.

Такие понятия о ЖЦ проекта предполагают сторонний взгляд на то, что есть в проекте. Понятно, что в отличие от классического управления проектами с неизменно утверждёнными требованиями к итогу и границам содержания, универсальное управление включает требования и содержание к изменённым динамически [2]. Сама суть универсального управления проектами соответствуют концепциям развитых и доступных проектов [7;6]. Если классические терминальные проекты включают неизменное определение границ ЖЦ и сути проекта, при достижении которых проект завершается, то развивающиеся проекты всегда открыты для дальнейших корректив и изменений. Открытые проекты вообще отражают содержание лишь в рамках общих направлений и индексов, которые корректируют принцип выполнения.

Итерационная природа универсального проекта предполагает совместную работу многих этапов создания новой продукции и подготовительных работ, планирования и реализации продукции, что очень похоже по сути на параллельное проектирование [4]. Универсальное управление проектов основано на постоянно зацикленном параллельном проектировании, что включает применение неких функциональных команд и улучшенную коммуникацию между исполнителями проекта. И по сути «подвижное» проектное управление включает в себя многие подходы, которые есть в интегрированной методике создания продукции (IPD) [3].

Гибкость и открытость по структуре и составу работ несет в себе повышенный уровень свободы от типовых процедур и процессов. Универсальное

управление проектом всячески указывает важность уменьшения документальной работы, сторонних процедур и процессов, явно определенных методов и подходов к пониманию задачи управления. Типовые стандарты и процедуры описывают как серьезные гасители «подвижности» в рамках управления. Упор в управлении реализуется на методы и связи, личностные способности, мотивацию, лидерство, координацию участников проекта. Исходя из китайской поговорки: «в руках грамотного человека хорошим станет даже плохой метод, а в руках безграмотного даже хорошее становится плохим».

Создание видения проекта включает реализацию видения продукции проекта, подготовку и утверждение документации проекта, понимания по содержанию проекта и его ограничениях, суммарной оценке и контроле, методе построения продукции, сути командной работы. Данный этап выражает общие контуры и параметры реализуемой продукции, суть проекта, участников всей системы проекта, ценности и правила, которые отражают стиль и порядок взаимодействия команды проекта. Понимание итогов и проекта становится не рекламным заявлением, а в рамках реального инструмента управления, и гораздо более важного, чем планы детальные. По сути, видение отражает и сам проект, его совокупные части и связи на уровне конкуренции и вдохновляет всю работу команды.

Вслед за подготовкой видения реализуется проектирование концепции, что направлена на выделение ресурсов и возможностей для реализации ведения. Такой проект состоит из описания параметров создаваемой продукции и общих направлений работы по производству. Такое проектирование может показать приблизительное число версий продукции, периодичность их реализации, их суть взаимодействия друг с другом.

Понимание цикла создания версии необходимо для создания подробного плана внедрения последующей версии продукта, т. е. прототипа. На данном этапе реализуется переход их концепции в режим итерационной разработки и адаптации. Если нужно, подготовку концепции можно повторять несколько раз, тем самым изменяя и видение, и сам проект, и планы генерации версий.

В цикле создания и адаптации реализуются действия для наработки ТЗ и плана работ для кодировщиков и интернов, проводится исправление ошибок,

проверка, программирование, корректировка требований, внедрение, понимание итоговых результатов, возможные правки требований, изменение плана и проекта (также изменяется и видение), и цикличное повторение процедуры до того момента, пока не будет реализован законченный продукт. По итогу выполняется передача готовой работы заказчику и завершение проекта. Все коррективы в концептуальном проекте будут записаны для накопления опыта и обучения, который важен для последующих проектов.

Дж. Хайсмит описывает 8 принципов контроля проектами:

- Поддержание адаптивной культуры организации со стороны инвесторов и директоров;
- Поддержка и внедрение само-организуемых и автономных команд для проекта;
- Поощрение проявления надежности и последовательности в рамках допустимых для конкретного проекта значений;
- Доступность адаптации к постоянным коррективам;
- Поддержка и отражение прозрачности процессов;
- Внедрение организационного обучения;
- Использование практик, поддерживающих каждую отдельную стадию (специализация практики и способностей);
- Внедрение адекватного числа чек-поинтов в проекте.

В комплексе управления НИОКР компании базовым видом его деятельности станет достижение результативной реализации проектов НИОКР, получение обновлённых продуктов.

Оценка проекта – важная процедура в управлении НИОКР на стартовой стадии реализации проекта, которая включает сложный постоянный процесс с учетом нахождения сторонней информации. Базируется она на формализованном методе [5].

Выделим, что в рамках оценки проекта важно учесть инвестиционные возможности реализации проекта, совмещение одного проекта с другими в рамках НИОКР, а также влияние проекта на компанию и ее экономику в совокупности в случае успешного запуска проекта в той сфере, в которой реализуется работа человека (финансирование от государства или рынок услуг).

Для описываемого холдинга есть самые важные сейчас проблемы:

- Нет интереса от гос. заказчика в реализации проекта (согласование, конкурсная основа, подготовка документации);
- Нет быстрого реагирования на негативные факторы со стороны управленческого аппарата.

Есть ряд проблем формирования, планирования и мониторинга создания совокупности НИОКР:

- Проблематичность коммуникации в процессе подготовки плана и реализации проектов из-за огромного числа активных участников;
- Отклонение от сроков и плановых показателей, что ведет к коммуникационному разрыву между проектами;
- Несовершенная регламентированная стратегия создания проектов (верхний план, показывающий приоритет работы над проектами) из-за повышенной неопределенности;
- Нет доступности быстрого получения данных касательно статусов уже реализованных работ любого проекта.

При этом финишная совокупная структура связана с уровнем контролируемости этой совокупности со стороны директората и политики компании в рамках НИОКР.

Для адекватного контроля проектом в менеджменте НИОКР нужна некая информационная база. Часто для этого применяют:

- Параметры оценивания проектов;
- Допуски и оценки, на которых основаны решения по проектам;
- Контроль проекта;
- Процедура реализации проекта.

И по итогу, важным является оперативное обновление всех видов данных, которые идут из других отделов предприятия (маркетингового, финансового). Организационные структуры контроля типа матричной наилучшим образом этому соответствуют.

Опишем критерии, которые отвечают за оценку и определение проектов при выборе управленческих решений: цели компании, её стратегия, ценности и политика; маркетинг; контроль НИОКР; денежные отношения и производство [5].

Параметры, связанные с политикой и стратегией компании, включают стратегическое планирование, рискованные операции, нововведения, аспект времени.

Определение проекта с повышенным риском (часто и с большими вложениями) нежелателен там, где руководство не любит принимать решения с повышенным риском. Это зачастую характерно для коммерческого сектора, чем технического, т.к. последний поддается ограничению, например, минимизации сферы НИОКР.

Критерии маркетинга следующие:

- Определение потребностей;
- Уровень желаемых продаж;
- Время;
- Текущие продукты и воздействие на них;
- Образование цены;
- Конкуренция;
- Сфера сбыта;
- Начальные расходы.

Плюсы рыночной потребности при выборе явного проекта не означают, что его рыночный успех гарантирован. Учет перспективности рынка реализуется рейтинговыми оценками перспектив его роста, адекватности бизнеса, его стабильности.

Уровень продаж, доступный для нового продукта, важно рассчитывать по перспективам его роста, а также методами прогноза темпа потребления продукции и корректировки потенциала потребителя (в рамках коммерческой структуры).

Время рыночного плана, часто связано с требованиями балансировки конкретных видов бизнеса фирмы. В идеале нужно, чтобы все новые продукты шли на рынок в объеме и качестве, которое диктуется маркетингом. Новые продукты повышают ассортимент, повышают размеры реализации и доходность, являются двигателем роста компании.

Объём продаж, который выражается в реализуемости, становится функцией от цены предложения. Доходы от продаж единицы продукции – разница между ценой и расходами на производство. Если расходы фирма-производитель не

может контролировать, то при корректировке цены важно учесть уровень цен на товары конкурентов.

Рамки установки предприятием цены на новую продукцию состоят из начальной цены продукции как низшей планки стоимости, а верхний предел выражен ценой потребления, по которой продукт еще остается привлекательным на рынке. Другими словами, верхний предел цены выражен параметрами качества продукта (надёжностью, функциональностью, дизайном, расходами на производство). Цена для потребителя также становится инструментом оценивания результативности от применения продукции с учётом цены ее потребления (цена покупки + расходы по эксплуатации).

Каналы продаж - фактор, который не игнорируется при оценке проекта. Если появится потребность в новых каналах продаж, то затраты поползут вверх. Также важно оценить стартовые затраты для выхода на рынок вообще.

Для определения стандарта главным фактором выступает более подробное и полноценное описание работы на этапах и стадия АС.

Сама каскадная модель имеет множество преимуществ, но при условии использования ее в проекте, приемлемом для нее. Ниже представлены ее преимущества:

- Модель хорошо знакома потребителям, не имевшим никакого отношения к созданию и эксплуатации программ, а также конечным пользователям (часто используется другими компаниями для отслеживания проектов, которые не связаны с разработкой ПО);
- Она лучше справляется с трудностями и отлично срабатывает в тех проектах, где все достаточно понятно, но трудноразрешимо;
- Она очень доступна для понимания, т.к. преследует простую цель – выполнение необходимых действий;
- Она проста и удобно в использовании, т.к. процесс разработки идет поэтапно.

Но в случае, если каскадная модель используется в проекте, не предназначенном для нее, проявляются следующие ее недостатки:

- Основа модели – линейная последовательная структура, и в результате попытки вернуться назад на одну-две фазы для исправления проблемы или недостатка приходится жертвовать временем и срывать график работ и затрат;
- Она не может предотвращать итерация между фазами, которые очень часто встречаются при создании ПО, поскольку сама модель строиться согласно циклам аппаратного инжиниринга;
- Она не показывает главное свойство разработки ПО, которое направлено на решение задачи. Отдельные фазы связаны определенными действиями, что часто отличается от привычной работы коллектива или персонала;
- Она создает ошибочное впечатление о работе с проектом. Указание, что «35% выполнено» обычно не имеет какого-то смысла для менеджера проектов.

Исходя из недостатков каскадной модели, ее применение нужно ограничивать ситуациями, в которых все требования для их разработки очень точны и понятны.

Спиральная модель особое внимание уделяет начальным этапам разработки – подготовке стратегии, проектированию и анализу, где все применяемые технические решения проверяются и обосновываются методом создания прототипов. Каждый виток спирали означает создание компонента или версии ПО. В них можно уточнять цели и характеристики проекта, его качество, а также выражаются работы на следующем витке. Таким образом, углубляются и конкретизируются детали проекта, и в результате определяется обоснованный вариант, который и реализуется.

Стандарт ISO/IEP 12207 не включает полноценного описания работы на этапах и стадиях реализации АС.

Стандарт CDM оправдан при работе с проектами, включающими Oracle-технологий, которые в нашем случае не используются.

Стандарт MSF, как было описано выше, направлен на бизнес-сферу.

Стандарт XP предпочтителен для команды. Поэтому в нашем случае будет использоваться ГОСТ 34.601-90, т.к. именно он имеет описание работы на любом этапе создания АС.

Основные стадии создания АС включают в себя:

- 1) Определение требований к системе;
- 2) Подготовка концепции;
- 3) Подготовка ТЗ;
- 4) Реализация технического проекта;
- 5) Написание документации;
- 6) Установка и использование.

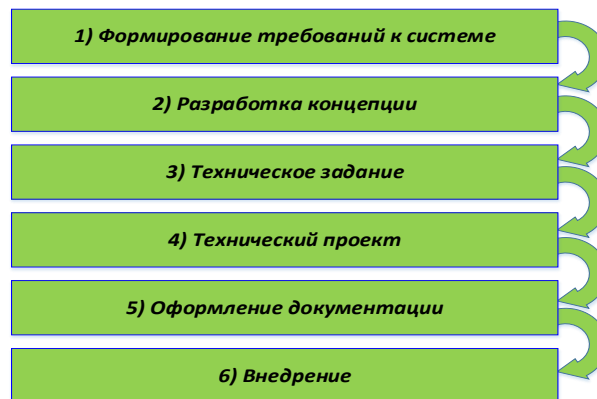


Рис. 2.1 Базовые стадии реализации ИС

В рамках этапа «Выведение требований к системе» реализуется следующее:

- Изучается сам объект;
- Готовятся требования пользователя;
- Указывается важность разработки.

В данном этапе используются такие участники, как: IT-менеджер, руководитель отдела производства. По факту создания всех задач готовится отчет о выполненной работе – описывается объект автоматизации, выделяются требования системе, отражаются расходы на создание, введение в работу и поддержку, указывается возможный эффект от реализации и отражаются условия для корректной работы системы.

По факту реализации этапа «Отражение требований к системе» готовятся виды концепций. Создается ряд доступных концепции и планов реализации, анализируют ресурсы, требуемые для реализации ИС и ее адекватной работы, изучают недостатки и преимущества всех методов, сверяют требования пользователей и показатели всех предлагаемых систем.

В рамках этапа «Подготовка концепции» используется только IT-менеджер. По факту завершения всех описанных работ определяется самый удачный из всех

приемлемых вариантов, который сможет полностью удовлетворить всем требованиям.

По факту завершения этапа «Подготовка концепции» выполняется ТЗ проекта автоматизации. По факту его подготовки нужно его согласовать и утвердить. В этом этапе принимают участие: IT-менеджер и руководитель отдела делопроизводства. По факту завершения этот пункт отражает - функции ИС и подсистем, состав совокупных и персональных задач, концепцию БД, состав СУБД, параметры и функции программных средств.

Этап установки состоит из: подготовки исследуемого объекта, тренинг сотрудников, проведение пуско-наладочных и монтажных работ, реализация испытаний, первый опытный запуск и приемочные испытания. На данном этапе задействованы: IT-менеджер, сисадмин, руководитель делопроизводства. По итогу происходит изучение итогов испытаний ИС, проверка соответствия ТЗ, устранения возможных неполадок и подпись всех актов.

Сейчас все чаще применяют такую следующую модель ЖЦ:

- Каскадная модель (рис. 2.2) включает в себя последовательную реализацию описанных этапов в порядке очередности. Переход на дальнейший этап отражает полную готовность на всех предыдущих.

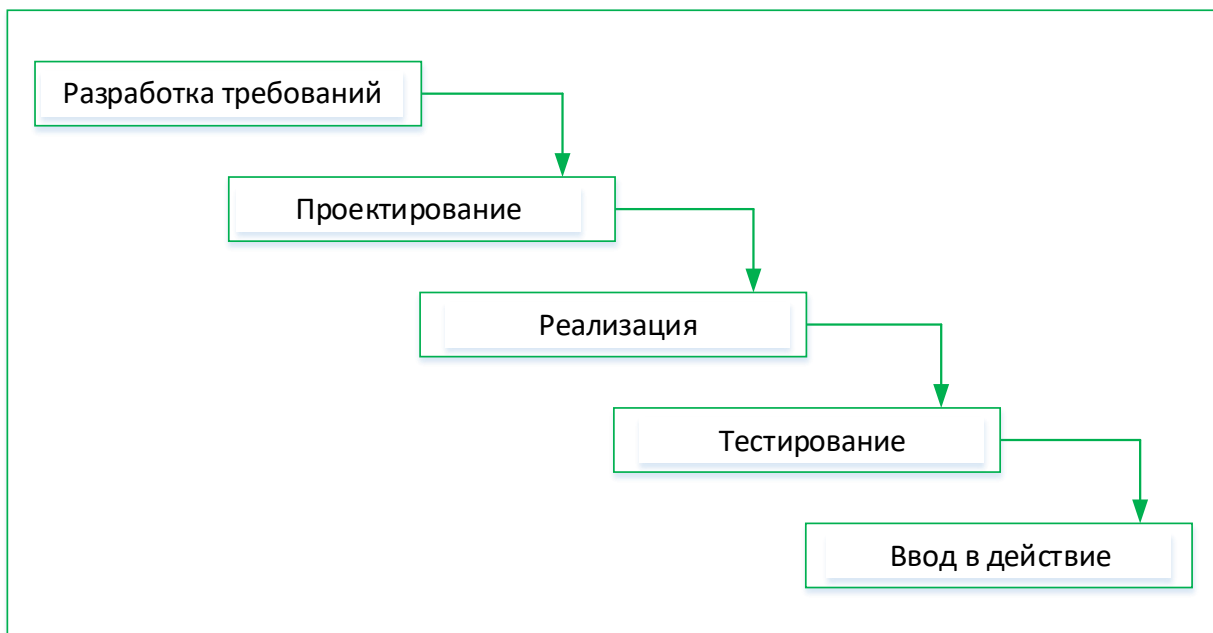


Рис. 2.2 - Каскадная модель ЖЦ ИС

Поэтапная модель с серединным контролем (Рис. 2.3). Создание ИС происходит итерациями с циклами обратной связи по каждому этапу.

Межэтапные проверки помогают учесть реально возникающие взаимовлияния итогов проектировки на различных этапах; время жизни любого из этапов равно совокупному периоду создания.

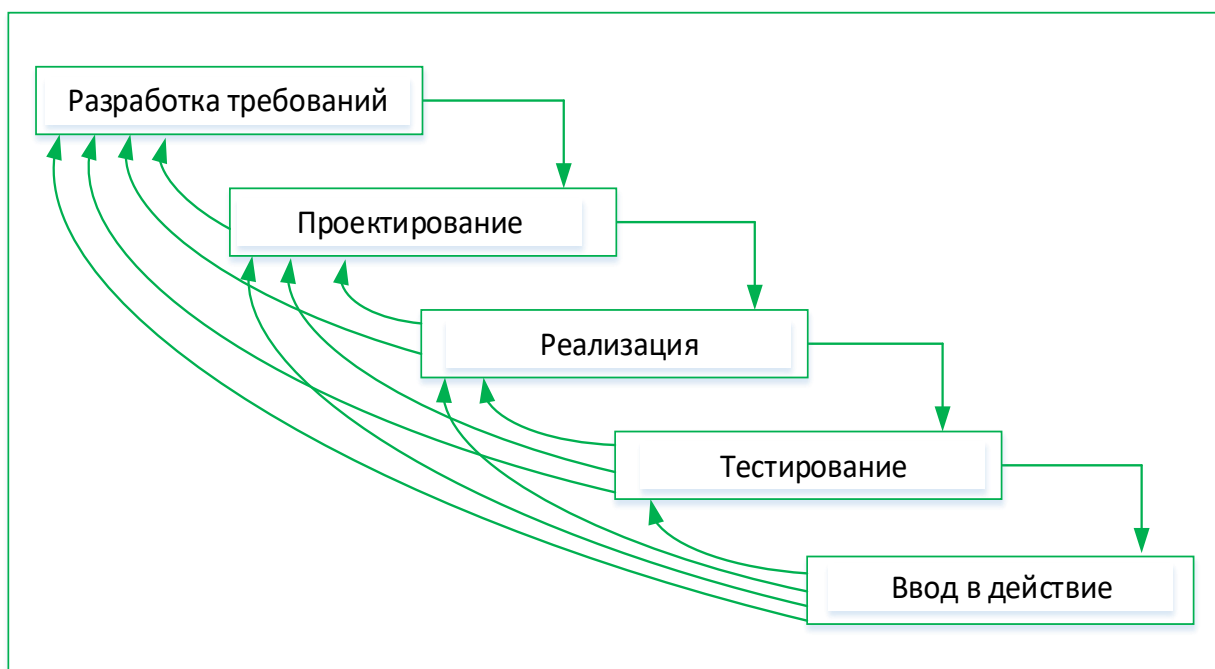


Рис. 2.3 - Поэтапная модель с срединным контролем

Спиральная модель (Рис. 2.4). Каждый цикл реализует создание другой версии продукта, утверждаются правила проекта, проверяется уровень его качество, рассматриваются работы будущего цикла.

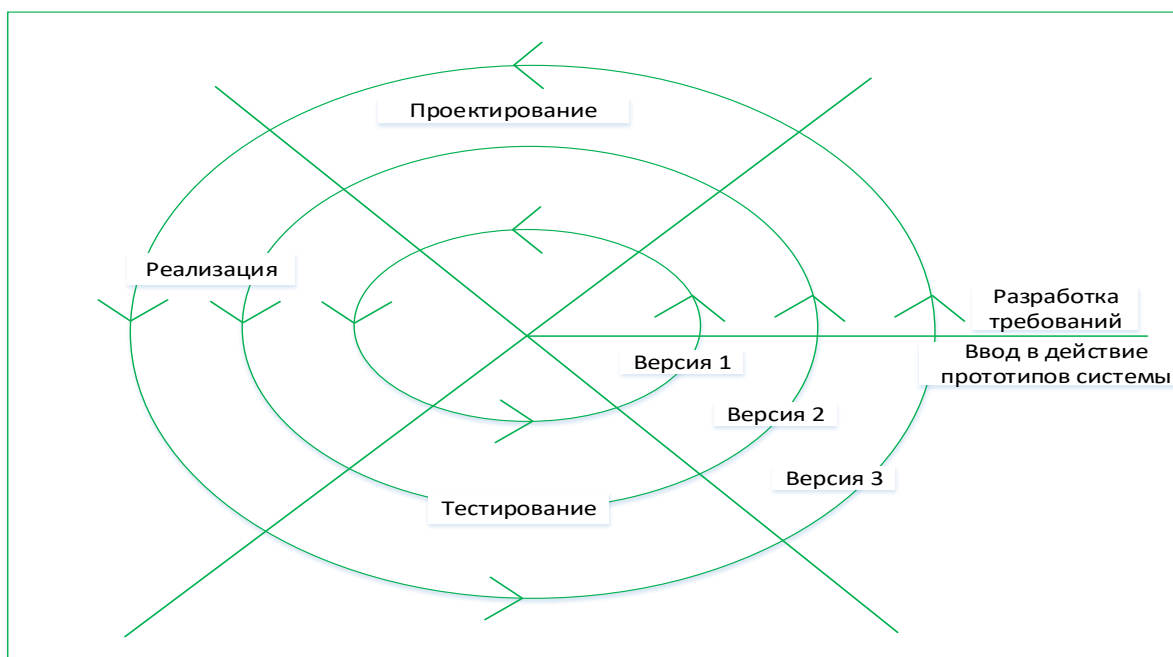


Рис. 2.4 - Спиральная модель ЖЦ ИС

Большое внимание уделено стартовым этапам разработки – разработке и изучению, где все технические решения корректируются и доказываются методом создания прототипов.

Для разрабатываемого проекта наиболее подойдет каскадная модель для разработки приложения из-за возможности контроля промежуточных фаз.

Далее произведем выбор стратегии внедрения разработанной системы. В настоящий момент выделяется четыре стратегии внедрения информационной системы:

- Параллельная стратегия - для случая, когда старую работающую систему необходимо заменить новой;
- Скачок – эта стратегия подразумевает резкий переход от одной системы автоматизации к другой;
- Опытная эксплуатация "пилотного проекта - это тактика "скачка", но применяемая к ограниченному числу изделий, наиболее успешна в малом участке деятельности;
- Узкое место - при внедрении "узкого места" план внедрения выполняется только для "узкого места" и для людей, работающих в нем.

Исходя из описания и условий деятельности компании, а также особенностей разрабатываемой информационной системы, в качестве стратегии внедрения была выбрана стратегия Опытная эксплуатация пилотного проекта, так как в этом случае внедрение системы произойдет наиболее безболезненно.

2.1.2 Ожидаемые риски на этапах жизненного цикла и их описание

В процессе разработки и внедрения информационной системы всегда могут присутствовать риски, то есть такие события, в результате реализации которых конечная цель проекта может быть не достигнута или достигнута с нарушением временных или финансовых рамок.

Такие события могут быть как определенными, так и неопределенными, то есть такими, которые могут как наступить, как и не наступить.

Для того, чтобы не допустить реализацию рисков, необходимо ими управлять. Для управления рисками нужно по возможности выявить все возможные риски и определить степень вероятности их возникновения.

Для управления рисками используется несколько стратегия, среди которых в качестве основных можно выделить избегание рисков, делегирование рисков и принятие рисков.

В результате избегания рисков вся работа строится таким образом, что риски стараются не допустить в принципе. Для этого все фазы проекта жестко формализуются и отклонение от них не допускается [29].

При использовании делегирования рисков ответственность за их возникновение несут специальные сотрудники из специального подразделения, в их же обязанности входит выполнение заранее спланированных мероприятий по снижению вреда от реализовавшихся рисков.

При принятии рисков планирование работы осуществляется с тем учетом, что все риски обязательно реализуются, в том числе корректируются и временные и стоимостные его показатели. В этом случае должен быть подготовлен план и порядок действий в результате возникновения незапланированных рисков.

Кроме того, риски могут быть как прямые, то есть такие, на которые сотрудники проекта могут повлиять, и косвенные, то есть те, которые зависят от сторонних причин.

Проект создания ИС управления проектами, как и все остальные проекты по созданию ПО, включает множество неопределенных моментов, которые могут повлечь за собой риски срыва реализации проекта.

Управление рисками состоит в их раннем выявлении и принятии мер, которые позволят либо 100% предотвратить их возникновение, либо значительно уменьшат последствия.

Сегодня существует три общепринятых стратегии управления рисками:

- Избегание рисков – проект строится так, чтобы исключить возможность появления любого риска;
- Делегирование рисков – проект строится так, чтобы передать все риски третьей стороне (инвесторам, банкам, заказчикам и т.п.);

- Принятие рисков – риски считаются неизбежной составляющей проекта, реализуется постоянный мониторинг симптомов их проявления, часто дорабатывается план действий в случае возникновения рисков.

Можно рассмотреть две базовые категории рисков – прямые и косвенные. На прямые риски проектная команда еще как-то можно повлиять, а вот косвенные риски нельзя проконтролировать в принципе.

Этап реализации концепции – есть риск подготовки концепции, которую не в силах будет реализовать. В концепции важно описать главные функции создаваемой ИС, выделить основу, и в дальнейшем уже улучшать созданную ИС.

Для минимизации рисков на этапе генерации концепции, нужно явно понимать свои возможности. Чтобы не переоценить свои силы, важно подготовить сначала общую концепцию, где уже будут иметься базовые функции системы. И в рамках продолжения реализации можно увеличивать и некие доп. функции.

Этап планирования может иметь риск неверной планировки, реализации завышенных планов проекта, когда фирма не сможет уложиться, что повлечет за собой рост длительности разработки, его удорожание. К этапу планирования важно отнестись очень внимательно, контролировать каждый шаг и понимать реализм результата.

Для уменьшения риска в рамках планирования важно заложить в график поправки на отдельные задержки в реализации конкретных работ. Так нужно создать такой гибкий график, который не изменялся бы из-за опережений и задержек.

Этап создания включает в себя риск того, что создание отдельного модуля будет связана со сложностями, а некая функция будет мешать ходу работ. В данном случае важно изначально понять сложный модуль или функцию и максимально ее упростить, поставить на ее место другую или удалить из проекта вообще.

Для минимизации риска разработки проблемного модуля, есть ряд решений: разбить модуль на несколько и решать все задачи в отдельном порядке, а также можно упростить модуль, если это становится единственным вариантом минимизации риска.

Этап тестирования включает в себя определение множества ошибок в программном коде, что ведет к глобальным расходам на доработку и устранение всех найденных ошибок. Нельзя заранее знать, сколько ошибок обнаружится и как много времени уйдет на их устранение.

Для уменьшения риска на этапе тестирования важно для данного этапа оставить больше всего времени, которое суммарно дается на реализацию системы, т.к. в зависимости от того, насколько грамотно будет создан продукт, будет зависеть, примет ли заказчик его или же нет.

Этап внедрения часто тоже бывает продолжителен, если заказчик не может сразу остаться довольным продуктом, да и сами сотрудники компании-заказчика могут с недоверием отнестись к новому ПО.

Для сокращения рисков в данной ситуации проводят качественное обучение сотрудников еще до периода эксплуатации, готовят отдел сопровождения и поддержки, понимают, что может произойти в процессе эксплуатации и как можно найти верное решение. Иметь возможность ответить на возникающие вопросы или открыть горячую линию для решения поступающих проблем.

Среди явных проблем процесса создания проекта можно выделить такие:

- Корректировка требований относительно процесса создания;
- Разделение ответственности за реализуемую работу и ее результат;
- Доступность потока мелких, срочных, приходящих требований, отвлекающих программистов и управленцев от базового направления работ.
- Срыв срока, повышение расходов, потери качественного уровня.

Чтобы реализовать успешную организацию процесса создания, подготовлена универсальная процедура создания проектов.

Модели, методики и подходы универсального управления проектами имеют свое развитие в рамках сложных технических проектов, которые заключаются в реализации больших программных комплексов. Универсальное управление можно рассматривать как некую платформу, которая находится в рамках нескольких методик контроля инновационными, а самое важно, ИТ проектами. Базовая суть универсального управления проектами описана в работах Дж. Хайсмита [1], Г. Аллемана [8], Г. Чина [11]. Сравнение разных прикладных

методологий универсального управления новейшими проектами также есть в трудах К. Лармана [27] и П. Абрамсона и других [5]. Моментный подход к определяю наилучшей методики универсального управления проектами указан в работе А.С.Коха [3].

Дж. Хайсмит выделяет базовое качество данного подхода так: «Гибкость (agility) - это возможность сразу и создавать, и отвечать на корректировки, создавая прибыль в изменчивых условиях экономики. Гибкость становится способностью держать баланс между стабильностью и хаосом» [16]. Также он пишет, что: «Часто многие склонны верить, что подвижность говорит об отсутствии структуры. Но такое отсутствие порождает хаос. Также, избыток некой упорядоченности несет за собой жесткость. Теория сложности описывает то, что процесс создания инновации методами, которые не определены заранее, часто реализован в точке соприкосновения порядка и хаоса, гибкости и стабильности. По словам ученых, реализация нового возможна на границе хаоса [16]. Но нахождение такого баланса между порядком и хаосом и будут направлены все возможности универсального управления проектами. Причем следует оно от упорядоченности процессов, инструментов и методик, которые есть в традиционных школах проектного управления в рамках деконструкции, деструктуризации множественных упорядоченных реализованных условий протекания проектов, к выработке понятных и логичных (направленных на персонал методов и инструментов, которые имеют возможность плавно приспособиться к изменчивым условиям). Если классическими параметрами управления проектами описывались планирование, улучшение и отслеживание, то универсальное управление проектами основными усилиями будет считать естественную эволюцию и приспособление.

В настоящее время используются три основных типа мероприятий по управлению рисками при разработке информационной системы:

Первая стратегия состоит в том, чтобы при разработке информационной системы вообще не допускать каких-либо рисков;

Суть второй стратегии заключается в том, что все возможные и предусмотренные риски необходимо делегировать третьей стороне, в качестве которой может выступать незаинтересованная сторона;

Третья стратегия заключается в принятии рисков как неотъемлемой части проекта и планировании всех этапов с учетом затрат на преодоление тех рисков, которые можно предусмотреть.

Риски можно разделить на прямые риски, которые непосредственно влияют на проект и опосредованные. На первые риски коллектив разработчиков может и обязан влиять, а вот вторая группа рисков такому влиянию неподвластна и поэтому такие риски наиболее опасны.

2.1.3 Организационно-правовые и программно-аппаратные средства обеспечения информационной безопасности и защиты информации

При эксплуатации разработанной информационной системы для обеспечения её безопасности от внешних и внутренних угроз используется комплекс мер по защите информации. В этот комплекс прежде всего входят средства, позволяющие ограничить доступ пользователей к различным модулям системы.

Для защиты от внутренних угроз в системе используется политика разделения прав доступа. Характеристика политики приведена в таблице 2.2.

Таблица 2.1

Разграничение прав пользователей

Группы пользователей	ПМ Авторизация	ПМ формирования проектов	ПМ формирования отчетов	ПМ работы со справочниками
Менеджер проектов	Чтение	Чтение	Чтение	Полный
Администратор системы	Полный	Полный	Полный	Полный
Клиенты	Чтение	Чтение	Чтение	Нет
Сотрудники	Чтение	Чтение	Чтение	Нет

В целях защиты информационной системы проводятся следующие мероприятия:

- обеспечение сетевой безопасности;

- обеспечение локальной безопасности;
- обеспечение физической безопасности.

Для обеспечения сетевой безопасности используются следующие средства:

- фильтрация трафика;
- ограничение доступа в интернет и во внутреннюю сеть;
- антивирусная фильтрация;
- система обнаружения атак;
- контроль содержания трафика;
- протоколирование и регулярный мониторинг доступа.

Локальная безопасность обеспечивается осуществлением следующих мероприятий:

- антивирусный контроль;
- аппаратная защита от несанкционированного доступа;
- криптографическая защита данных;
- защита персональным файрволом;
- резервирование данных;
- протоколирование доступа.

Для обеспечения физической информации файрвол, веб-сервер, сервера IDS и контроля за трафиком и все сервера данных находится в отдельном помещении, доступ в которое разрешен только администраторам, у которых есть ключ или магнитная карта к этой комнате (комната закрыта). Помещение оборудовано принудительной вентиляцией и пожарной защитой (полуавтоматической). Вход в офис компании должен осуществляться только по магнитным картам.

Электронный замок «Соболь»

Созданный и поставляемый ЗАО НИП «Информзащита», «Соболь» включает в себя следующие функций защиты:

- Определение и аутентификация пользователей;
- Отслеживание целостности файлов и физических секторов накопителя данных;
- Запрет загрузки ОС с внешних устройств;

- Запрет на вход в систему зарегистрированного пользователя при исчерпании им указанного количества неверных попыток входа;
- Запись событий, относящихся к безопасности системы.

Кроме указанной политики доступа, в разработанной информационной системе обеспечивается безопасность по таким направлениям, как сетевое, локальное и физическое.

Сетевая безопасность подразумевает такие мероприятия, как проверка трафика, использование ограничений в доступе к ресурсам, фильтрация трафика на антивирусной основе, запись всех событий при контроле трафика и последующий анализ этих событий.

Для обеспечения локальной безопасности осуществляются меры по обеспечению антивирусного контроля, контроля прав доступа, контроля за проникновениями и несанкционированным доступом, защита сетевыми средствами, контроль данных и их резервное копирование, а также шифрование данных при их передаче.

2.2 Информационное обеспечение задачи

2.2.1 Информационная модель и её описание

Предметная область информационной системы может быть описана с помощью информационной модели – графической схемы, на которой отображаются все потоки данных, в том числе входная информация и документы, на основе которых она формируется, экранные формы, в которые она вводится, порядок прохождения и обработки информационной системы, используемые хранилища данных, результатные документы во всех видах (в виде экранных форм, экспортируемых файлов или печатных форм).

Информационная модель представляет собой схему движения входных, промежуточных и результативных потоков и функций предметной области. Кроме того, она объясняет, на основе каких входных документов и какой нормативно-справочной информации происходит выполнение функций по обработке данных и формирование конкретных выходных документов. Информационная модель представлена на рис. 2.5.

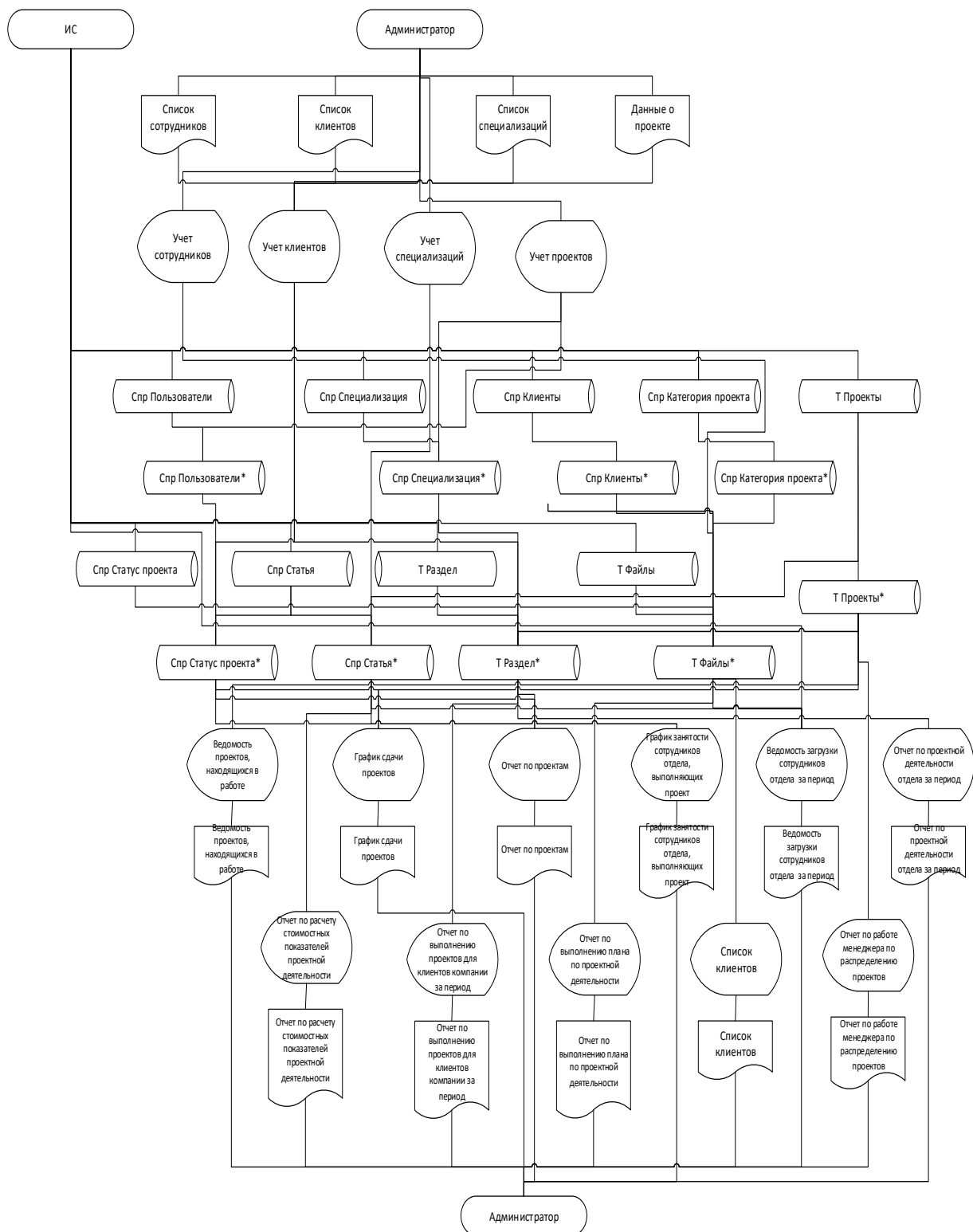


Рис. 2.5 Информационная модель системы

2.2.2 Характеристика нормативно-справочной, входной и оперативной информации

К нормативной информации относится та информация, которую необходимо использовать из-за норм и правил разработки, проектирования и использования информационных систем. К справочной информации относится информация, которая может быть записана в систему один раз и используется все время на протяжении эксплуатации информационной системы. При этом такая информация может периодически обновляться, но реже, чем информация, которая является несправочной.

К справочной информации относятся в первую очередь таблицы базы данных, которые относятся к справочникам системы.

Входной информацией для проектируемой системы являются проекты клиентов, сведения о сотрудниках компании, содержащиеся в штатном расписании, сведения о клиентах, а также сведения о категориях проектов, формируемые на основании деятельности компании. Эти данные поступают как в цифровом, так и в печатном виде.

С этой целью составляются перечень входной информации и состав реквизитов каждого вида входной информации, расположение реквизитов входной информации, описание полей (реквизитов) входных документов[13].

Значительное место в автоматизированной системе учета занимает нормативно-справочная информация. Нормативно-справочная информация — это условно-постоянная составляющая общей корпоративной информации, она используется при регламентации деятельности компании, обеспечивая целостность данных, сопровождающих бизнес-процессы компании [20].

По назначению нормативно-справочная информация используется для:

- идентификации единиц информации;
- информационно-поисковых целей;
- формирования характеристик объектов учета;
- обеспечения информационных взаимосвязей и расчетов.

Другими словами, нормативно-справочная информация — это ядро единого информационного пространства организации, включающее в себя набор справочников, словарей, классификаторов, стандартов, регламентов, используемых в деятельности организации [14].

Данные из входных документов вносятся в систему путём ручного ввода данных через веб-интерфейс.

Из штатного расписания в систему вводятся следующие данные:

- ФИО пользователя;
- специализация.

Сведения о категориях проектов содержат только наименования данных реквизитов.

Основным документом, вносимым в систему, является заявка на выполнение проекта.

Для обеспечения работы системы предусмотрены справочники, приведенные в таблице 2.3.

Таблица 2.2

Перечень используемых справочников

название справочника	ответственный за ведение	средний объём справочника в записях	среднюю частоту актуализации	средний объём актуализации, %
Специализации	Администратор	45	1 раз в месяц	10
Клиенты	Администратор	150	1 раз в год	10
Категории	Администратор	10	1 раз в месяц	10
Пользователи	Администратор	250	1 раз в год	10
Статусы пользователей	Администратор	5	1 раз в год	10

Справочник Специализации формируется на основании штатного расписания и содержит только наименование отдела. На основании этого же документа формируется и справочник Пользователи. Справочник Категории – на основании результатов анализа работы отдела. Справочник Пользователи формируется на основании штатного расписания предприятия, а также Справочника Статусы пользователей в системе.

В качестве входной информации для разрабатываемой ИС используются следующие документы:

- проект, который содержит перечень необходимых работ и документов,

- Нормативная стоимость проектных работ;
- Список сотрудников отдела с указанием специализации каждого;
- График загрузки каждого сотрудника отдела, в котором указано, какой проект каждый сотрудник выполняет в настоящее время и в какой срок должен закончить текущую работу;

- Список клиентов компании, заказывающих проектирование.

Клиент учитывается путем внесения данных в экранную форму «Учет клиентов».

Сотрудники также заносятся в систему с использованием экранной формы.

В данном проекте будут использованы следующие локальные классификаторы:

- 1) Классификатор статусов сотрудников

Структурная схема классификатора сотрудников представлена на рис. 2.10..

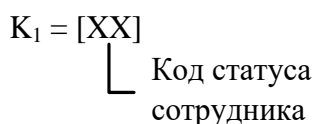


Рис. 2.6 Структурная схема классификатора статуса сотрудников

- 1) Классификатор сотрудников

Структурная схема классификатора сотрудников представлена на рис. 2.11..

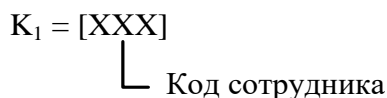


Рис. 2.7 Структурная схема классификатора сотрудников

- 2) Классификатор типов работ

Структурная схема классификатора представлена на рис. 2.12.

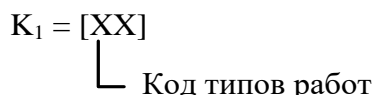


Рис. 2.8 Структурная схема классификатора типов работ

- 3) Классификатор работ

Структурная схема классификатора представлена на рис. 2.13.

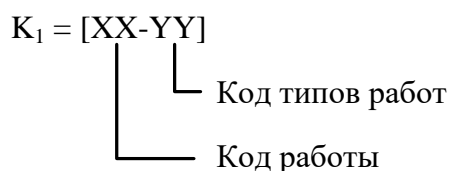


Рис. 2.9 Структурная схема классификатора работ

4) Классификатор клиентов

Структурная схема классификатора представлена на рис. 2.14.



Рис. 2.10 Структурная схема классификатора клиентов

5) Классификатор проектов

Структурная схема классификатора представлена на рис. 2.15.

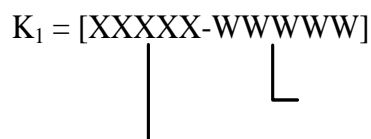


Рис. 2.11 Структурная схема классификатора билетов

Сводное описание всех вышеописанных классификаторов представлено в таблице 2.2.

Таблица 2.3

Сводная таблица используемых классификаторов и систем кодирования

№ п/ п	Наименование кодируемого множества	Значимость кода	Система кодирования	Мощность кода	Система классифи- кации	Вид классифи- катора
1	Классификатор статусов сотрудника	2	порядковая	99	отсутствует	локальный
2	Классификатор сотрудников	5	разрядная	99999	отсутствует	локальный
3	Классификатор типов работ	2	разрядная	99	отсутствует	локальный
4	Классификатор работ;	4	разрядная	9999	отсутствует	локальный
5	Классификатор клиентов	5	разрядная	99999	отсутствует	локальный

6	Классификатор проектов	10	разрядная	999999999 9	отсутствует	локальный
---	------------------------	----	-----------	----------------	-------------	-----------

Входной информацией называется такая информация, которая поступает в систему из разных источников и в разном виде, необходимая для функционирования самой системы. Такая информация может поступать как в виде документов, информация с которых вводится в экранные формы системы, в виде различных файлов, импортируемых в систему, в виде скриптов и элементов базы данных, а так же в любом другом виде. Как правило, именно на основании входной информации в результате ее обработки формируется результатная информации.

Главным качеством входной информации является достоверность, то есть такая информация должна быть правдивой, корректно заполненной, иметь все необходимые реквизиты, а также быть полной, то есть в таком виде, в котором ее достаточно для работы системы.

Оперативной информацией называется такая информация, которая постоянно используется в системе и постоянно и в режиме реального времени, то есть оперативно. Именно такая информация поступает в систему в течение всего жизненного цикла информационной системы, поддерживая ее функциональность.

2.2.3 Характеристика результатной информации

Результатная информация является итогом деятельности информационной системы, такая информация может быть реализована в виде экранных форм, в виде документов или файлов или какой-либо другой форме, которая может быть удобна для восприятия именно в данной информационной системе.

В системе учета кадров в первую очередь к результатной информации может быть отнесены информация о сотруднике в виде личной карточки, а также все документы, формирующиеся в результате работы системы – приказы отдела кадров, информация об отпусках, больничных, отчеты в органы государственной власти.

В результате работы системы формируются следующие выходные документы:

- 1) Ведомость проектов, находящихся в работе;
- 2) График сдачи проектов;
- 3) Отчет по проектам;
- 4) График занятости сотрудников отдела, выполняющих проекты;
- 5) Ведомость загрузки сотрудников отдела за период;
- 6) Отчет по проектной деятельности отдела за период;
- 7) Отчет по расчету стоимостных показателей проектной деятельности;
- 8) Отчет по выполнению проектов для клиентов компании за период;
- 9) Отчет по выполнению плана по проектной деятельности;
- 10) Отчет по работе менеджера по распределению проектов.

Ведомость проектов, находящихся в работе, содержит код проекта, название проекта, сотрудника, ответственного за выполнение проектов, даты начала и окончания работ, стоимость проекта и его статус.

2.3 Программное обеспечение задачи

2.3.1 Общие положения (дерево функций и сценарий диалога)

Функциональность разработанной системы представлена в виде рисунка дерева функций, на котором в иерархическом виде представлены все функции разработанной информационной системы. Все функции можно разделить на служебные (авторизация, сохранение данных, выход из системы) и основные, то есть такие, которые описывают порядок выполнения основного назначения информационной системы. Кроме того, присутствуют функции по составлению и получению отчетов по деятельности информационной системы.

Диалогом называется процесс обмена данными между пользователем и ПО, реализуемый посредством интерактивного терминала при условии выполнения нескольких правил [12]. Общение ПК и пользователя в процессе работы с ИС создано на базе диалогового режима, который контролирует сама программа.

Интерфейс пользователя создается по технологии WIMP (Window – окно, Image – картинка, Menu – меню, Pointer - указатель). Тут могут применяться и элементы графического интерфейса: окна диалогов и приложений, контекстное и иерархическое меню, средства вывода и ввода данных, значки панели

инструментов, программные указатели и т.п. Такие графические элементы выполняют как служебные, так и базовые программные функции. Сценарий диалога является структурной схемой, где в виде дерева отражены действия пользователя в процессе взаимодействия с отдельной формой и описаны функциональные возможности панели инструментов. Любая форма включает иерархическое меню, которое отражает все права пользователя в процессе работы с конкретной формой.

Дерево функций пользователя разработанной ИС представлено на рисунке 2.17.

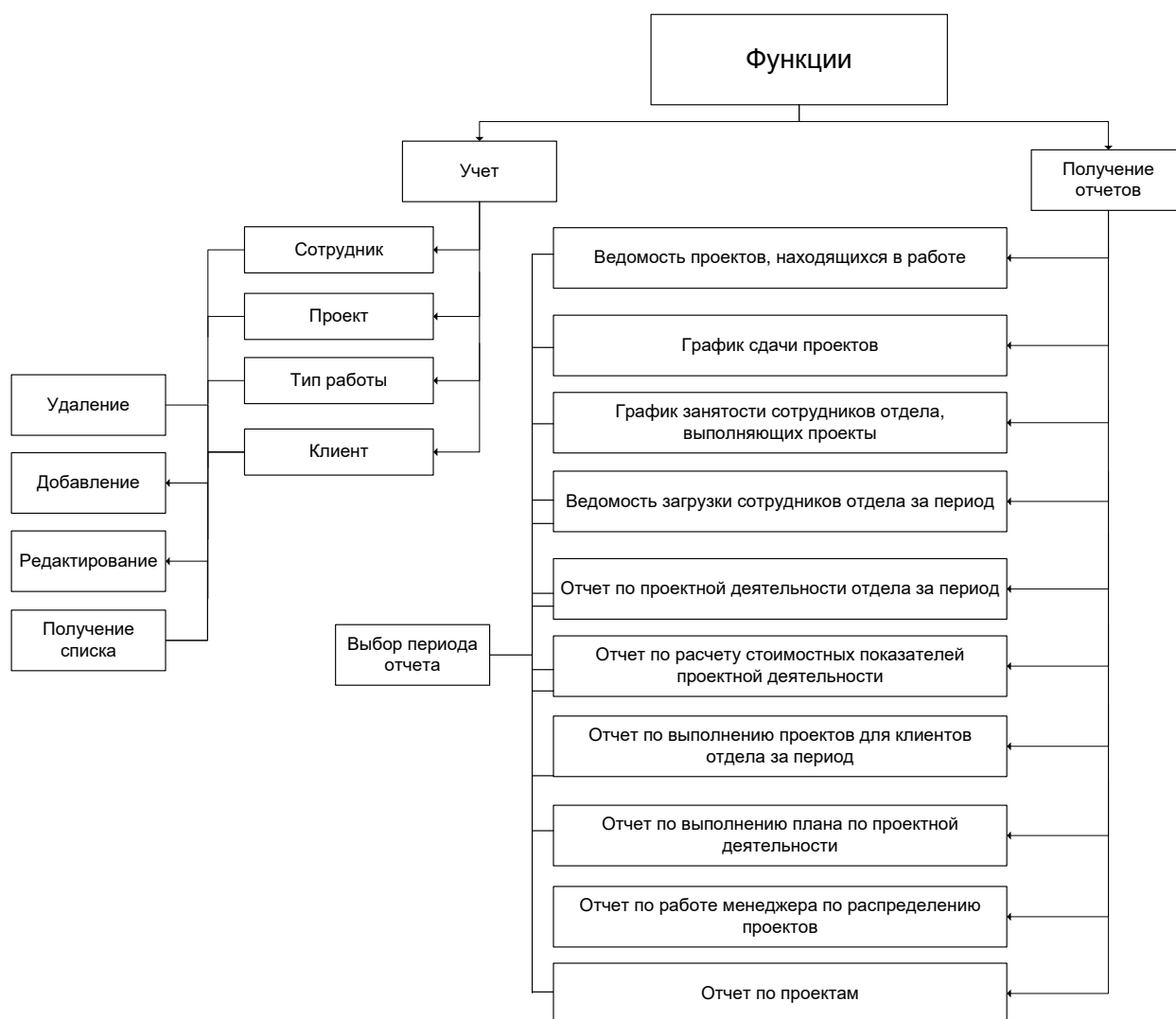


Рис. 2.12 Дерево функций пользователя ИС

Сценарий диалога формируется на основе дерева функций. В разработанной системе сценарий построен по иерархическому принципу. Работа

начинается с вызова главной экранной формы, на которой присутствует 2 вкладки – формы и отчеты.

Вкладка Формы содержит 5 пунктов меню с вызовом экранных форм для учета проектов, работ, клиентов и сотрудников.

Вкладка Отчеты содержит меню для выбора периода отчета и кнопки для вызова отчетов.

Сценарий диалога приведен на рисунке 2.18.

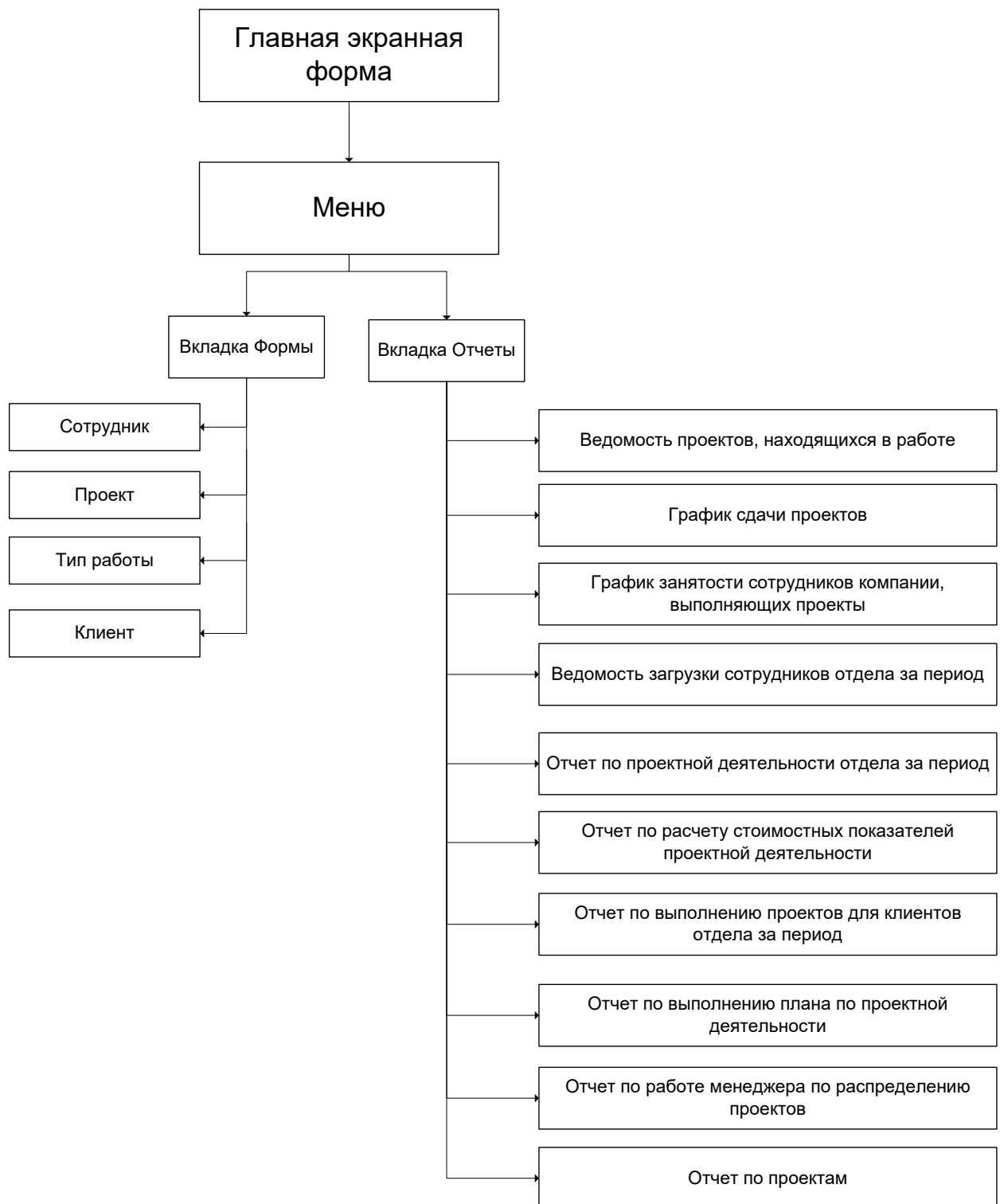


Рис. 2.13 Сценарий диалога системы

2.3.2 Характеристика базы данных

В рамках создания ИС следует понимать функциональную и обеспечивающие части системы. Обеспечивающие подсистемы ИС становятся

общими для всей ИС независимо от отдельных функциональных подсистем, где используются те или иные виды обеспечения. Состав обеспечивающих подсистем не связан с выбранной предметной области [11]. Число обеспечивающих подсистем включает в себя:

- Информационное — совокупность методик и средств по установке в компании информации, имеющей системы классификации и кодирования документов, показателей, данных;
- Техническое — совокупность технических средств, материалов и персонала, которые помогают обслуживать технику;
- Программное — некое число программных средств для проведения задач управления с применением вычислительной техники.
- Правовое — набор правовых норм, которые описывают генерацию и разработку ИС (договор между заказчиком и создателем ИС и т.п.);
- Лингвистическое — набор языковых средств, применяемых на разных уровнях создания и обработки информации для человеко-машинного взаимодействия.
- Организационное — набор методов и средств, которые помогают улучшить организационную структуру объектов, подготовить должностные инструкции, направленные на новые ИТ.

Обеспечивающие подсистемы реализуют информационные процессы в системе и отвечают за их подготовку и организацию. Также важно описать самые важные подсистемы новой ИС.

Само ИО в нашей ИС включает в себя 2 компонента: немашинное ИО, включающее в себя параметры технико-экономических данных, методических предписаний, а также внутримашинное ИО, включающее макеты, интерфейса пользователя для передачи начальной информации в ЭВМ или вывода итоговой структуры ИБ: входных, выходных файлов, БД. Само ИО в любой ИС часто понимается как некий набор совокупной классификации, информационной базы и некой системы классификации.

Самой информационной моделью ИС станет ER-диаграмма применяемой БД.

В качестве информационной модели АРМ разработаем ER-диаграмму используемой базы данных.

Для Хранения данных в базе данных необходимо предусмотреть следующие сущности и атрибуты:

1. Проекты:
2. Категория проекта
3. Клиенты
4. Пользователи
5. Специализация
6. Статус проекта
7. Статья
8. Раздел
9. Файлы

Инфологическая схема базы данных представлена на рисунке 2.19.

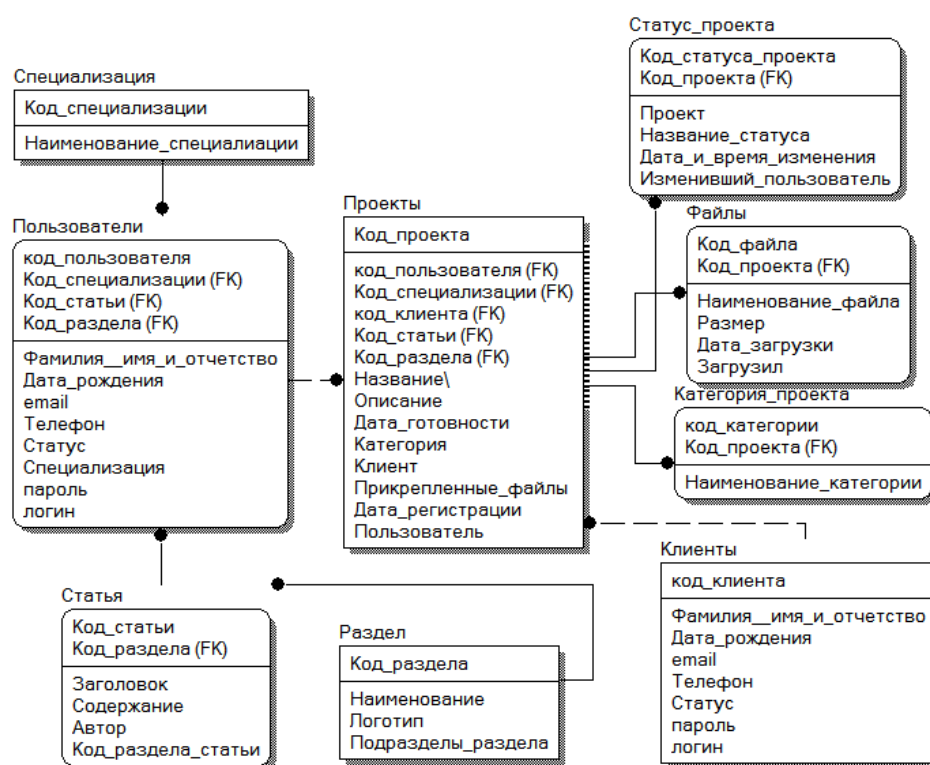


Рис. 2.14 Инфологическая схема базы данных

ER-модель разработанной базы данных представлена на рисунке 2.20.

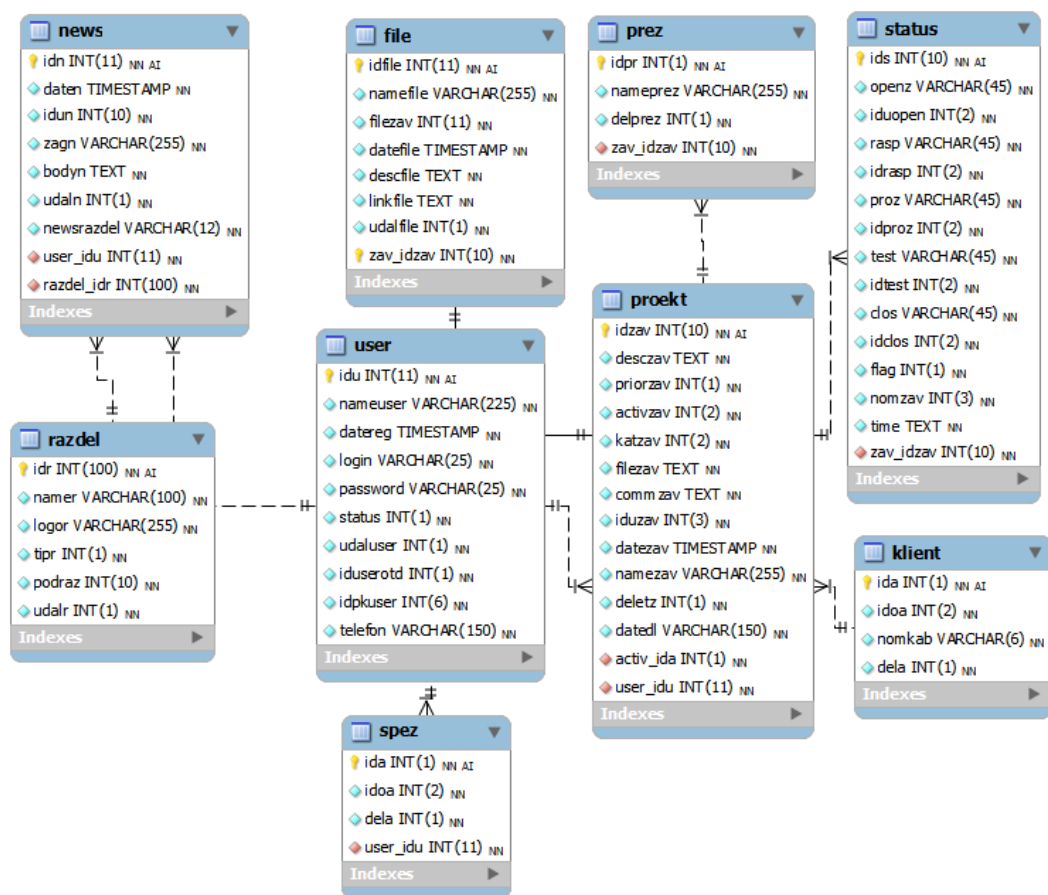


Рис. 2.15 Даталогическая модель разработанной базы данных

Далее определим для каждой таблицы тип поля и формат содержащихся в нем данных.

Таблица 2.4

Характеристика и описание полей таблицы Proekts

№	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код записи	idzav	int(10)	auto_increment
2.	Описание проекта	desczav	text	
3.	Приоритет	priorzav	int(1)	
4.	Актив	activzav	int(7)	
5.	Категория	katzav	int(7)	
6.	Ссылка на прикрепленный файл	filezav	text	
7.	комментарий	commzav	text	
8.	Код добавившего клиента	iduzav	int(3)	
9.	Дата формирования	datezav	timestamp	
10.	Наименование	namezav	varchar(255)	

11.	Дата окончания преокта	Datedl	date	
12.	Признак удаления	deletz	int(1)	

Таблица 2.5

Характеристика и описание полей таблицы Klient

№	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код записи	ida	int(1)	auto_increment
2.	Код компании	idoa	int(7)	
3.	ФИО	nomkab	varchar(6)	
4.	Признак удаления	dela	int(1)	

Таблица 2.6

Характеристика и описание полей таблицы status

№	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код записи	ids	int(10)	auto_increment
2.	Дата и время открытия	openz	varchar(45)	
3.	Код пользователя	iduopen	int(7)	
4.	Дата и время распределения	rasp	varchar(45)	
5.	Код пользователя	idrasp	int(7)	
6.	Дата и время приема	proz	varchar(45)	
7.	Код пользователя	idproz	int(7)	
8.	Дата и время проверки	test	varchar(45)	
9.	Код пользователя	idtest	int(7)	
10.	Дата и время закрытия	clos	varchar(45)	
11.	Код пользователя	idclos	int(7)	
12.	Признак удаления	flag	int(1)	
13.	Номер проекта	nomzav	int(3)	

Таблица 2.7

Характеристика и описание полей таблицы user

№	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код сотрудника	idu	int(11)	auto_increment
2.	ФИО сотрудника	nameuser	varchar(25)	
3.	Дата регистрации	datereg	timestamp	
4.	Логин для доступа в систему	login	varchar(25)	
5.	Пароль для доступа в систему	password	varchar(25)	
6.	Дата рождения	status	int(1)	
7.	Признак удаления	udaluser	int(1)	

Таблица 2.8

Характеристика и описание полей таблицы news

№	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код записи	idn	int(11)	auto_increment
2.	Дата	daten	timestamp	
3.	Код пользователя	idun	int(10)	
4.	Заголовок	zagn	varchar(255)	
5.	Тело записи	bodyn	text	
6.	Признак удаления	udaln	int(1)	
7.	Номер раздела	newsrazdel	varchar(12)	

Таблица 2.9

Характеристика и описание полей таблицы razdel

№	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код раздела	idr	int(100)	auto_increment
2.	Наименование	namer	varchar(100)	
3.	Логотип	logor	varchar(255)	
4.	Тип раздела	tipr	int(1)	1 – основной, 2 - подраздел
5.	Номер основного раздела	podraz	int(10)	При наличии
6.	Признак удаления	udalr	int(1)	

Таблица 2.10

Характеристика и описание полей таблицы Spez

№ пп	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код	ido	int(1)	
2.	Наименование	nameo	varchar(255)	
3.	Признак удаления	delo	int(1)	

Таблица 2.11

Характеристика и описание полей таблицы Kategorija

№ пп	Название столбца поля	Идентификатор	Тип	Примечание
1.	Код категории	idpr	int(1)	
2.	Наименование	nameprez	varchar(255)	
3.	Признак удаления	delprez	int(1)	

Таблица 2.12

Характеристика и описание полей таблицы File

№	Название столбца поля	Идентификатор	Тип	Примечание
---	-----------------------	---------------	-----	------------

1.	Код записи	namefile	varchar(255)	auto_increment
2.	Название файла	filezav	int(11)	
3.	Дата и время добавления	datefile	timestamp	
4.	Описание	descfile	text	
5.	Ссылка	linkfile	text	
6.	Признак удаления	udalfile	int(1)	

2.3.3 Структурная схема пакета (дерево вызова программных модулей)

Система реализована в виде совокупности программных модулей, каждый из которых выполняет определенную функцию или группу функций [21].

При этом программа снабжена навигацией, которая позволяет быстро и удобно перемещаться между модулями и соответственно выполнять функции программы в необходимом порядке.

Для отображения информации используются экранные формы в виде таблиц, так как в таком виде восприятие информации наиболее удобно. При этом поля и сортировка полей могут настраиваться и отображаться в разном порядке, по желанию пользователей. Некоторые поля с информацией, которая является служебной (например, код записи), скрыт от пользователя.

При вводе информации используются маски полей, не позволяющие ввести информацию неправильного вида в определенные поля (например, в поле ввода номера телефона нельзя ввести фамилию или наоборот).

Используется буферизация информации для избежание одновременного я информации различными пользователями. То есть при работе с массивом информации одним пользователем другому пользователю недоступно изменение такой информации.

Редактирование информации производится с помощью стандартного меню удалить, редактировать, добавить, изменяемая информация обновляется в базе данных, но в тоже время для сохранения целостности удаляемая информация не удаляется из базы данных, а только помечается таковой.

Зачастую автоматизация компании позволяет увеличить эффективность ее конкурентоспособности и стратегической направленности всех сторон бизнеса. Она также часто помогает оптимизировать бизнес-процессы, в том числе позволяет управлять деятельностью, трудовыми и технологическими ресурсами для комплексного подхода к улучшению результатов.

Сегодня модно наблюдать стремительное развитие глобальной информатизации всего общества. При таком подходе меняется и вся информационная среда, новые информационные технологии быстрее проникают практически во все сферы деятельности и быстро становятся неразрывной частью новой приобретенной информационной культуры человечества [20].

Поэтому последнее время стало так важно и актуально правильно формировать информационные технологии как самостоятельную дисциплину, изучающую новые методы создания эффективных ИТ (как в узком понимании термина, так и в обычном), т.е. своеобразную методологию и теорию проектирования различных информационных систем.

Структурная схема пакета ИС приведена на рисунке 2.21.

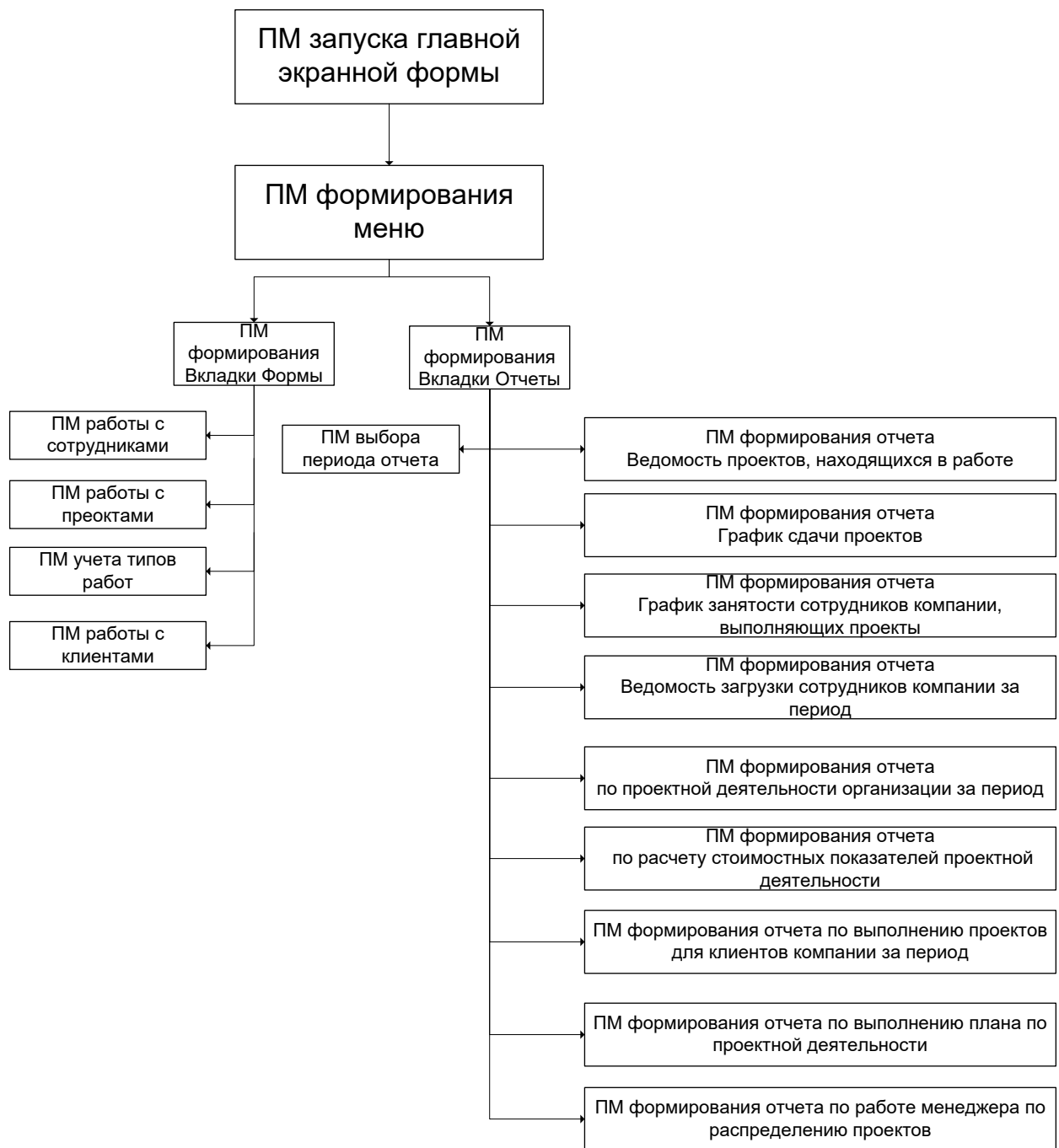


Рис. 2.16 Структурная схема пакета ИС

Представим краткое описание программных модулей, приведенных выше.

Программный модуль формирования главного меню отвечает за вывод на экран окна с двумя вкладками – Формы и Отчеты;

Программный модуль формирования вкладки Формы отвечает за вывод на экран формы с кнопками вызова экранных форм работы с данными;

Программный модуль формирования вкладки Отчеты отвечает за вывод на экран формы с кнопками вызова экранных форм работы с отчетами;

Программный модуль выбора периода отчета позволяет выбрать начальную и конечную дату получения отчета;

Программный модуль получения отчета позволяет вывести на экран Ведомость проектов, находящихся в работе в результате использования SQL запроса.

Остальные модули по формированию отчетов работают аналогично и предназначены для вывода на экран каждого отдельного отчета.

2.3.4 Описание программных модулей

Схемы технологического процесса сбора, передачи, обработки и выдачи информации для разработанной системы представлены на рисунке 2.22.

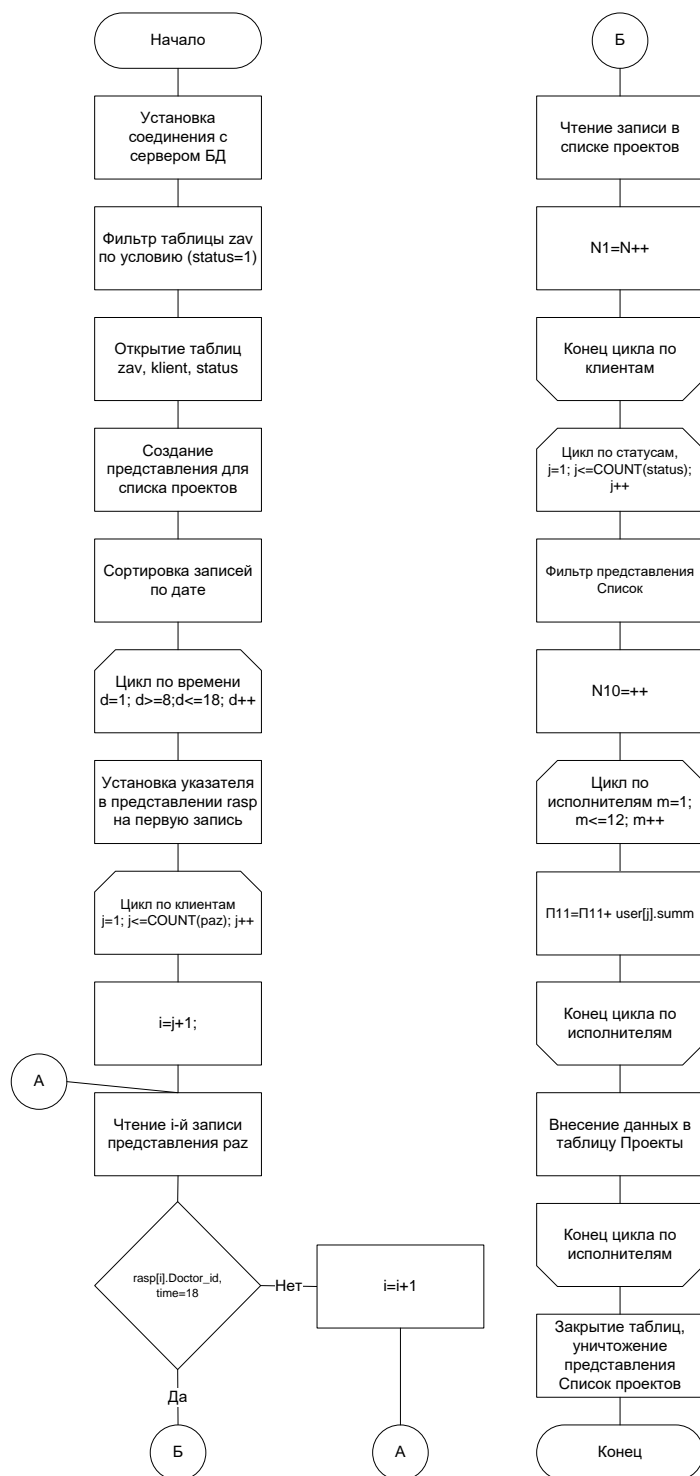


Рис. 2.17 Начало работы с системой (главное меню)

После авторизации сотрудник в главной форме выбирает пункт меню, которые соответствует необходимым действиям – отчеты или учет.

2.4 Контрольный пример реализации проекта и его описание

Основой разработки любой информационной системы являются методологии ее проектирования, к которым относятся средства проектирования и его стандарты, в том числе стандарты описания бизнес-процессов. Конкретный

метод проектирования информационной системы может быть реализована через стандарты, средства проектирования, другие средства, обеспечивающие выполнение каждого из этапов жизненного цикла информационной системы.

Самым сложным этапом в разработке и проектировании информационной системы является, как правило, этап анализа предметной области, для которого используются CASE средства. Эти средства используются, так как позволяют визуализировать необходимую информацию и на основе результатов анализа строить различные диаграммы, позволяющие принимать решения о этапах проектирования информационных систем.

Такие средства позволяют четко определить область разрабатываемой информационной системы, при необходимости изменить ее описание в реальном времени, провести проверку всех правил, механизмов, потоков информации как входной, так и результатной, изменить текущие ограничения.

При этом такие средства могут быть использованы как в относительно недорогих персональных компьютерах, так и специфичных компьютерных системах, что позволяет использовать такие средства для разработки информационной системы любого масштаба и размера.

Как правило, движение потоков данных в разрабатываемой информационной системе, назначение основных хранилищ данных и пути формирования результатных документов как в виде экранных форм, так и в виде печатных документов представляется на этапе проектирования в виде информационной модели. В этой схеме показывается, как информация попадает в систему, где и как хранится и обрабатывается, где формируется и какие пользователи имеют возможности ее использовать, в том числе получать отчеты.

Далее опишем порядок использования разработанного программного обеспечения.

Создаем модель (или открываем сохраненную модель из файла). При создании модели вводятся критерии и факторы (внешние и внутренние), на основании которых осуществляется оценка качества сайта (рисунок 3.8).

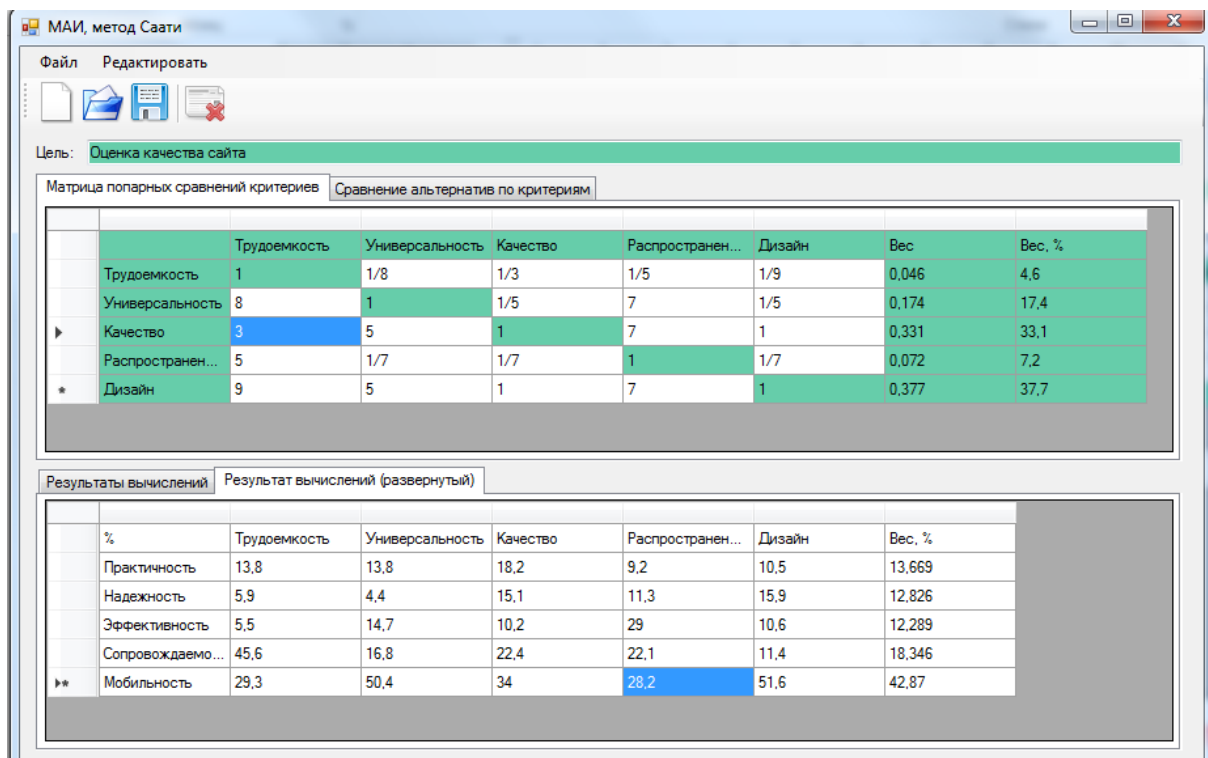


Рисунок 2.18 Главная форма программы

Далее, с помощью меню Редактировать, определяем перечень конкретных альтернатив и критериев оценки качества сайта:

В данном случае внутренними критериями и альтернативами для оценки качества сайта являются:

- K1 = «Надежность»;
- K2 = «Практичность»;
- K3 = «Эффективность»;
- K4 = «Сопровожаемость»;
- K5 = «Мобильность»;
- K6 = «Трудоемкость реализации»;
- K7 = «Универсальность»;
- K8 = «Качество реализации»;
- K9 = «Распространенность»;

В программе также можно указать количество критериев и альтернатив:

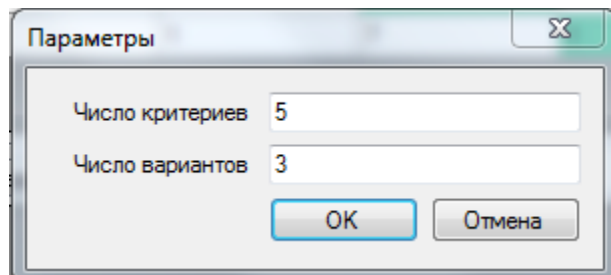


Рисунок 2.19 Количество критериев и альтернатив

Далее заполняем матрицу попарных сравнений критериев (для каждого критерия и альтернативы):

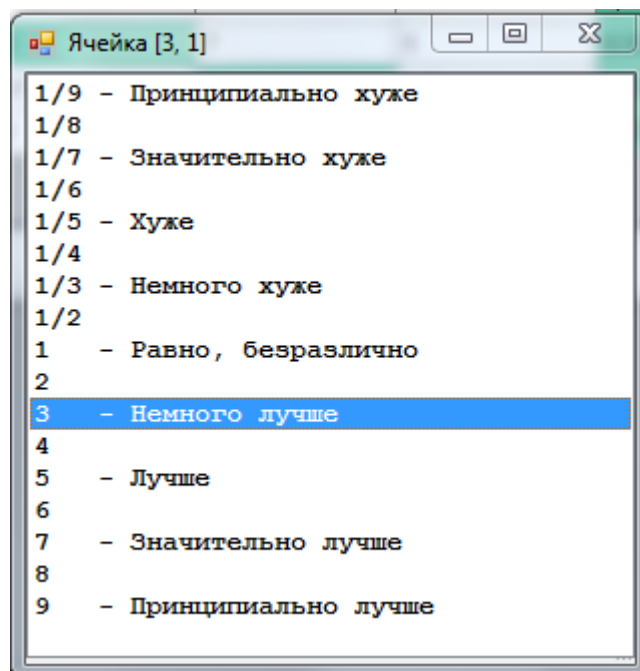


Рисунок 2.20 Матрица попарных сравнений

И получаем развернутый результат расчетов:

MAI, метод Саати

Файл Редактировать

Цель: Оценка качества сайта

Матрица попарных сравнений критериев Сравнение альтернатив по критериям

Критерий 1 Критерий 2 Критерий 3 Критерий 4 Критерий 5

		Практичность	Надежность	Эффективность	Сопровождается...	Мобильность	Вес	Вес, %
Практичность	1		1/3	9	1/8	1/9	0,105	10,5
Надежность	3		1	7	3	1/7	0,159	15,9
Эффективность	1/9		1/7	1	7	1/8	0,106	10,6
Сопровождается...	8		1/3	1/7	1	1/7	0,114	11,4
Мобильность	9		7	8	7	1	0,516	51,6

Результаты вычислений Результат вычислений (развернутый)

	%	Трудоемкость	Универсальность	Качество	Распространен...	Дизайн	Вес, %
Практичность	13,8	13,8	18,2	9,2	10,5	13,669	
Надежность	5,9	4,4	15,1	11,3	15,9	12,826	
Эффективность	5,5	14,7	10,2	29	10,6	12,289	
Сопровождается...	45,6	16,8	22,4	22,1	11,4	18,346	
Мобильность	29,3	50,4	34	28,2	51,6	42,87	

Рисунок 2.21 Развернутый результат расчетов (сравнение альтернатив)

Вносим изменения в матрицу попарных сравнений:

MAI, метод Саати

Файл Редактировать

Цель: Оценка качества сайта

Матрица попарных сравнений критериев Сравнение альтернатив по критериям

Критерий 1 Критерий 2 Критерий 3 Критерий 4 Критерий 5

		Практичность	Надежность	Эффективность	Сопровождается...	Мобильность	Вес	Вес, %
Практичность	1		3	5	1/5	1/9	0,138	13,8
Надежность	1/3		1	1/3	1/3	1/8	0,044	4,4
Эффективность	1/5		3	1	3	1/7	0,147	14,7
Сопровождается...	5		3	1/3	1	1/3	0,168	16,8
Мобильность	9		8	7	3	1	0,504	50,4

Результаты вычислений Результат вычислений (развернутый)

	%	Вес, %
Практичность	13,669	
Надежность	12,826	
Эффективность	12,289	
Сопровождается...	18,346	
Мобильность	42,87	

Рисунок 2.22 Матрица попарных сравнений

Эта матрица представляет собой сравнение критериев друг относительно друга. Меню редактирования ячейки матрицы вызывается

двойным кликом мыши. Редактировать можно только ячейки с белым фоном.

Вторая матрица представляет собой сравнение альтернатив друг относительно друга по определенному критерию. Меню редактирования ячейки матрицы вызывается двойным кликом мыши. Редактировать можно только ячейки с белым фоном.

1. После заполнения всех матриц получаем результат

В кратком виде:

The screenshot shows the 'MAI, метод Саати' application window. The 'Цель: Оценка качества сайта' (Goal: Website quality evaluation) is set. The 'Матрица попарных сравнений критериев' (Criteria pairwise comparison matrix) tab is active, displaying a 5x5 matrix for criteria: Практичность (Practicality), Надежность (Reliability), Эффективность (Effectiveness), Сопровождается... (Accompanied by...), and Мобильность (Mobility). The matrix values are: Practicality (1, 3, 1/7, 7, 1/9), Reliability (1/3, 1, 7, 3, 1/9), Effectiveness (7, 1/7, 1, 1/3, 1/7), Accompanied by... (1/7, 1/3, 3, 1, 7), and Mobility (9, 9, 7, 1/7, 1). The 'Результаты вычислений' (Calculation results) tab is also visible, showing a table with the following data:

	%	Вес, %
Практичность	13,669	
Надежность	12,826	
Эффективность	12,289	
Сопровождается...	18,346	
* Мобильность	42,87	

Рисунок 2.23 Результат сравнения альтернатив

В развернутом виде:

МАИ, метод Саати

Файл Редактировать

Цель: Оценка качества сайта

Матрица попарных сравнений критериев Сравнение альтернатив по критериям

Критерий 1	Критерий 2	Критерий 3	Критерий 4	Критерий 5
	Практичность	Надежность	Эффективность	Сопровожаемо...
Практичность	1	3	1/7	7
Надежность	1/3	1	7	3
Эффективность	7	1/7	1	1/3
Сопровожаемо...	1/7	1/3	3	1
Мобильность	9	9	7	1/7

Результаты вычислений

%	Трудоемкость	Универсальность	Качество	Распространен...	Дизайн	Вес, %
Практичность	13,8	13,8	18,2	9,2	10,5	13,669
Надежность	5,9	4,4	15,1	11,3	15,9	12,826
Эффективность	5,5	14,7	10,2	29	10,6	12,289
Сопровожаемо...	45,6	16,8	22,4	22,1	11,4	18,346
Мобильность	29,3	50,4	34	28,2	51,6	42,87

Рисунок 2.24 Результат сравнения альтернатив в развернутом виде

В результат работы программы мы получаем, что оцениваемый сайт имеет следующие результаты оценки:

Практичность – 13 пунктов из 100;

Надежность – 12 пунктов из 100;

Эффективность – 12 пунктов из 100;

Сопровожаемость - 18 пунктов из 100;

Мобильность – 42 пункта из 100.

Таким образом, разработанная методика позволяет группе экспертов оценить сайт или любой другой программный продукт по нескольким критериям, в результате чего получаем комплексную оценку качества сайта.

3 Обоснование экономической эффективности проекта

3.1 Выбор и обоснование методики расчета экономической эффективности

За период работы и создания различных ИС было решено выделять часть общих принципов их экономического обоснования и реализации подобных процессов. В любом случае оценка результативности подобной ИС обязана быть комплексной и включать все социальные, экологические и экономические, а также многие другие особенности принимаемых решений.

Базовой целью экономического описания новой ИС станут количественное и качественное обоснование экономической рентабельности разработки или обновления ИС, а также выявление основных экономических и организационных условий для последующей спокойной работы.

При создании ряда компонентов или элементов ИС, итоги экономического анализа включаются непосредственно по ним. Важно понимать, что в любом процессе подготовки ИС нужно учесть несколько аспектов экономического обоснования.

ПО стоит на особом месте в составе комплексного инструмента для автоматизации. Оно также имеет внутри себя программные модули или некоторые пакеты программ, созданных для применения сторонней ИС (заимствованные), или реализуемые для обширного круга использования в разных ИС (универсальные), или классические для различных сфер использования ПК. Также такого типа ПО почти в каждой ИС включает оригинальные, созданные на заказ программы и алгоритмы, требуемые для нормальной работы данной ИС. Важно всегда отражать процесс создания ПО из общей совокупности процессов удобно определять его с двух позиций:

- Прикладной, т.к. реализуемые программные продукты можно применять независимо от базовой системы;
- Системной, т.к. создано пиратское ПО.

Для корректного планирования процесса разработки ПО и расчета итоговой цены важно определить эффективность работы программных средств. Деление труда в процессе внедрения ПО всегда граничит с рядом трудностей, так же, как и любое нормирование творческой деятельности, элементы которого могут

объединится только после проведения экспертных оценок главных программистов, либо сложными и нереальными сроками подготовки всего проекта, что заставляет IT-специалиста выбирать решение. Творческая часть работы программистов можно нормировать, но точность этих норм всегда очень большая и зависит от ряда факторов. Только в рамках опытно-статистических данных от многих компаний, которые программируют или занимаются им, Открыто несколько методов расчета сложности создания ПО.

В рамках подсчета экономической эффективности этого диплома лежит на сравнении двух вариантов для единой задачи нового и текущего. Базовый вариант помогает автоматизировать некую часть процессов учета, а описываемая модель реализует полноценную автоматизацию процесса.

Совокупная эффективность часто зависит от пары составляющих:

- Косвенный эффект: определяется при повышении точности и верности данных, ростом максимальных объемом анализируемых данных;
- Прямой эффекта позволяющий минимизировать время на обработку и получение данных, Минимизацией трудоёмкости процесса работы и траты времени на описание проведенных операций.

Для расчет прямого эффекта, опишем параметры выходных данных и стоимость затрат.

Экономическая эффективность разработанной системы выражается в соотношении полезного конечного результата ее функционирования к затраченным ресурсам. Положительная экономическая эффективность достигается в случае, когда результаты деятельности организации превышают затраты. В рамках данного выпускной квалификационной работы будет произведен расчет прямого и косвенного эффекта от внедрения проекта автоматизации. Для расчета прямого эффекта будет использована методика сравнительного анализа, которая сводится к сравнению трудовых и стоимостных затрат до и после автоматизации и основывается на расчете следующих показателей:

К трудовым показателям можно отнести следующие:

- Абсолютное уменьшение трудовых затрат (ΔT) (час), которое рассчитывается по формуле:

$$\Delta T = T_0 - T_1, \quad (3.1)$$

где T_0 – базовый вариант трудозатрат на обработку данных (час),

T_1 – расчет трудозатрат на обработку данных по предлагаемому варианту (час);

- Параметр относительного уменьшения трудозатрат (K_T) (%), для подсчета которого используется формула:

$$K_T = \Delta T / T_0 * 100\%; \quad (3.2)$$

- Индекс повышения производительности или уменьшения трудозатрат (Y_T), который можно высчитать по формуле:

$$I_T = T_0 / T_1. \quad (3.3)$$

К стоимостным параметрам обычно относят:

- Абсолютное уменьшение стоимости затрат (ΔC) (руб.), выводимое по формуле:

$$\Delta C = C_0 - C_1, \quad (3.4)$$

где C_0 равно стоимостным затратам (в рублях) по базовому варианту обработку данных,

C_1 выражает стоимостные затраты (в рублях) на реализацию обработки данных по предлагаемому варианту;

- Коэффициент относительного уменьшения стоимостных расходов (K_C) (%), выражающийся формулой:

$$K_C = \Delta C / C_0 * 100\%; \quad (3.5)$$

- Индекс повышения производительности или уменьшения трудозатрат (Y_C), который можно высчитать по формуле:

$$I_C = C_0 / C_1. \quad (3.6)$$

Подсчет вышеперечисленных показателей позволяет предположить вычисление частных показателей, например:

- Трудоемкость отдельно взятой i -ой операции (T_i):

$$T_i = V_i / N_i, \quad (3.7)$$

где V_i выражает объем выполняемых на i -ой операции работ (действий или символов),

N_i равняется норме выполнения объема работ на каждой i -ой операции в один час;

- Стоимостные затраты на реализацию каждой i -ой операции (C_i):

$$C_i = C_{з/п} + C_{нр} = T_i * P_i * (1 + K_{нр}), \quad (3.8)$$

где $C_{з/п}$ является стоимостными расходами на заработную плату сотруднику, выполнившему каждую i -ую операцию (руб.);

$C_{нр}$ выражает сумму накладных расходов (руб.);

P_i рассчитывает часовую тарифную ставку сотрудника, реализующего i -ую операцию (руб.);

$K_{нр}$ выражает коэффициент финансовых потерь, возникающих при выполнении i -ой операции;

- Суммарная стоимость машинной работы по времени:

$$C_{\text{машвр}} = \sum_{m=1}^M (C_m^{\text{маш.вр}} * t_m^{\text{раб.маш}}), \quad (3.9)$$

где $C_m^{\text{маш.вр}}$ является стоимостью работы за каждую единицу времени на m -ом компьютере;

$t_m^{\text{раб.маш}}$ выражает общую продолжительность работы m -того компьютера.

Для расчета трудозатрат на обработку данных по базовому варианту (T_0) (час) используют следующую формулу:

$$T_0 = \sum T_{0i}, \quad (3.10)$$

где T_{0i} определяет трудоемкость реализации i -ой операции при базовом варианте решения задачи.

Для подсчета трудозатрат на обработку данных по предлагаемому варианту (T_1) (час) нужно пользоваться следующим алгоритмом:

$$T_1 = \sum T_{1j}, \quad (3.12)$$

где T_{1j} является трудоемкостью исполнения j -ой операции для предлагаемого варианта решения.

Чтобы просчитать стоимостные затраты на обработку данных по проектному варианту (C_1) (руб.), используется формула:

$$C_1 = \sum C_{1j} + C_{1 \text{ машвр}} + K_{\text{эклс}}, \quad (3.13)$$

где C_{1j} отражает стоимостные затраты на реализацию j -ой операции для предлагаемого варианта решения.

К главным обобщающим показателям экономической эффективности можно отнести: годовой экономический эффект, расчетный параметр эффективности капитальных вложений, время окупаемости затрат на запуск рассматриваемого проекта.

Годовой экономический эффект от создания и запуска информационной системы ($\mathcal{E}_Г$) определяется разностью абсолютного снижения стоимостных затрат и предполагаемой прибылью:

$$\mathcal{E}_Г = \Delta C - K_{\Pi} * E_{\Pi}, \quad (3.14)$$

где K_{Π} отражает расходы на реализацию проекта автоматизированной обработки данных, руб.;

E_{Π} является прописанным параметром эффективности капитальных вложений.

В данном случае произведение K_{Π} и E_{Π} можно рассматривать как норму прибыли, которая получается при внедрении системы. Параметр E_{Π} обычно равен 0,15 для любой отрасли народного хозяйства. E_{Π} также можно рассматривать как минимальную эффективность капитальных вложений, ниже которой они уже нерентабельны.

Вычисленное значение показателя годового экономического эффекта ($\mathcal{E}_Г$) в данном случае способствует сопоставлению экономических итогов автоматизации обработки информации с результативностью капитальных вложений в остальные направления улучшения производства и управления им.

Расчетный параметр эффективности капитальных затрат (E_p) обозначает отношение абсолютного снижения затрат стоимостных к затратам на реализацию и проектирование системы:

$$E_p = \Delta C / K_{\Pi}. \quad (3.15)$$

Время окупаемости (T_{OK}) обычно является неким отношением капитальных затрат на создание и установку ЭИС к максимальному снижению стоимостных затрат:

$$T_{OK} = K_{\Pi} / \Delta C. \quad (3.16)$$

Расходы на реализацию проекта автоматизированной обработки информации (K_{Π}) (руб.) считается как некая сумма расходов по категориям:

$$K_{\Pi} = K_{\text{предпр}} + K_{\text{проект}} + K_{\text{разр}} + K_{\text{внедр}} + K_{\text{эксп}} \quad (3.17)$$

$K^{\text{предпр}}$ является затратами на начальное обследование (руб.);

$K^{\text{проект}}$ содержит затраты на подготовку проекта данных (руб.);

$K^{\text{разр}}$ включает расходы на разработку, тестирование, написание всех пояснительных документов (руб.);

$K^{\text{внедр}}$ несет затраты на установку и отладку (руб.).

$K^{\text{эксп}}$ показывает стоимость на эксплуатацию и сопровождение (руб.)

3.2 Расчёт показателей экономической эффективности проекта

В соответствии с выбранной выше методикой произведем расчет основных показателей экономической эффективности разработанной информационной системы.

Для этого необходимо оценить объемы обрабатываемой информации в базовом и проектном варианте. Для расчета стоимостных показателей учтем, что заработная плата сотрудника составляет около 40000 рублей. Так как в рабочей неделе содержится 5 рабочих дней по 8 часов, рабочих дней в месяце в среднем – 22, стоимость одного часа работы составит:

$$40000/(22*8)=238 \text{ рублей в час.}$$

Как было указано выше, основным документом в автоматизированной задаче является проект. Объем данного документа в бумажном эквиваленте составляет около 20 документострок. В рабочий день оформляется около 10 документов, что в год составляет:

$$20*10*20*12=48000 \text{ документострок.}$$

Кроме того, при работе сотрудника необходимо учитывать следующие документы (таблица 3.1):

Таблица 3.1

Список и объем обрабатываемых документов

Наименование	Объем в день, д/с	Объем в год, д/с
заявка на выполнение проекта	64	16896
данные клиента	32	8448
список специализаций	80	21120
список пользователей	160	42240
список проектов клиентов	400	105600

отчет по поступившим, обработанным и закрытым проектам за произвольный период	320	84480
отчет по проектам, выполненным определенным сотрудником за произвольный период	320	84480
отчет по проектам, поданным определенным клиентом за произвольный период	320	84480

Тогда операции технологического процесса при базовом и проектном варианте за год и их характеристики представлены в таблице 3.2 и таблице 3.3

Таблица 3.2

Базовый вариант

№ п/п	Наименование операций технологического процесса решения комплекса задач	Оборудо- вание	Ед. Изм.	Объем работы в год	Норма выра- ботки / (опер/в час.)	Трудоемкость	Среднечасовая зарплата специалиста (руб.)	Часовая стоимость накладных расходов (руб.)	Стоимостные затраты для ручных операций
1.	заявка на выполнение проекта	нет	д/с	16896	70	241,37	255,68	9	63886,63
2.	данные клиента	нет	д/с	8448	70	120,69	255,68	9	31943,31
3.	список специализаций	нет	д/с	21120	70	301,71	255,68	9	79858,29
4.	список пользователей	нет	д/с	42240	70	603,43	255,68	9	159716,57
5.	список проектов клиентов		д/с	105600	70	1508,57	255,68	9	399291,43
6.	отчет по поступившим, обработанным и закрытым проектам за произвольный период		д/с	84480	70	1206,86	255,68	9	319433,14
7.	отчет по проектам, выполненным определенным сотрудником за произвольный период		д/с	84480	70	1206,86	255,68	9	319433,14
8.	отчет по проектам, поданным определенным клиентом за произвольный период		д/с	84480	70	1206,86	255,68	9	319433,14

Итого:					6396,34			1692995,66
--------	--	--	--	--	---------	--	--	------------

Таблица 3.3

Проектный вариант

№ п/п	Наименование операций технологического процесса решения комплекса задач	Оборудование	Ед. Изм.	Объем работы в год	Норма выработки (опер/в час.)	Трудоемкость	Средне-часовая зарплата специалиста (руб.)	Часовая стоимость накладных расходов (руб.)	Стоимостные затраты для ручных операций
1.	заявка на выполнение проекта	ПК	д/с	16896	205	82,4195122	255,68	2,5	21279,22
2.	данные клиента	ПК	д/с	8448	205	41,21	255,68	2,5	10639,61
3.	список специализаций	ПК	д/с	21120	205	103,02	255,68	2,5	26599,02
4.	список пользователей	ПК	д/с	42240	205	206,05	255,68	2,5	53198,05
5.	список проектов клиентов	ПК	д/с	105600	205	515,12	255,68	2,5	132995,12
6.	отчет по поступившим, обработанным и закрытым проектам за произвольный период	ПК	д/с	84480	205	412,10	255,68	2,5	106396,10
7.	отчет по проектам, выполненным определенным сотрудником за произвольный период	ПК	д/с	84480	205	412,10	255,68	2,5	106396,10
8.	отчет по проектам, поданным определенным	ПК	д/с	84480	205	412,10	255,68	2,5	106396,10

	клиентом за произвольный период								
	Итого:					2184,12			563899,32

Абсолютный показатель снижения трудовых затрат на обработку информации:

$$\Delta T = 6396,34 - 2184,12 = 4212,23 \text{ часов}$$

Показатель снижения стоимостных затрат

$$\Delta C = 1692995,66 - 563899,32 = 1129096,34 \text{ рублей}$$

Рассчитаем затраты на внедрение и проектирование системы, для чего учтем заработную плату разработчиков. При средней заработной плате 57000, привлечении двух разработчиков и общем сроке разработки 20 рабочих дней (месяц), принимаем стоимость разработки – 104000 рублей.

Срок окупаемости затрат на внедрение проекта машинной обработки информации:

$$T_{\text{ок}} = 104000 / 1129096 = 0,09 \text{ года или около 2 месяцев.}$$

Рассчитаем расчетный коэффициент эффективности капитальных затрат:

$$E_p = 1 / T_{\text{ок}} = 1 / 0,11 = 0,7$$

На рис. 3.1. приведена диаграмма сравнения базового и проектного вариантов трудовых затрат, на рис. 3.2 – стоимостных затрат.

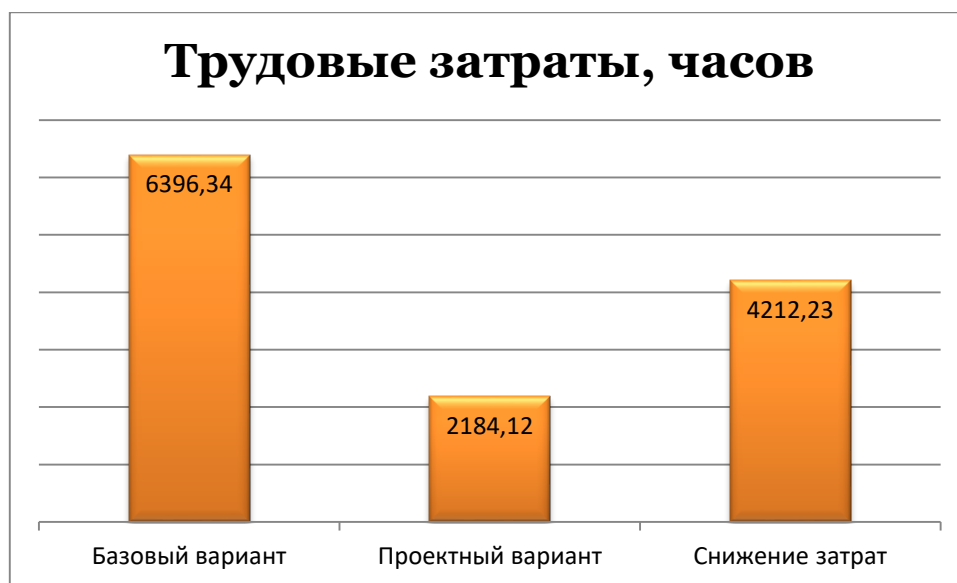


Рис. 3.1 Диаграмма сравнения базового и проектного варианта трудовых затрат обработки информации.

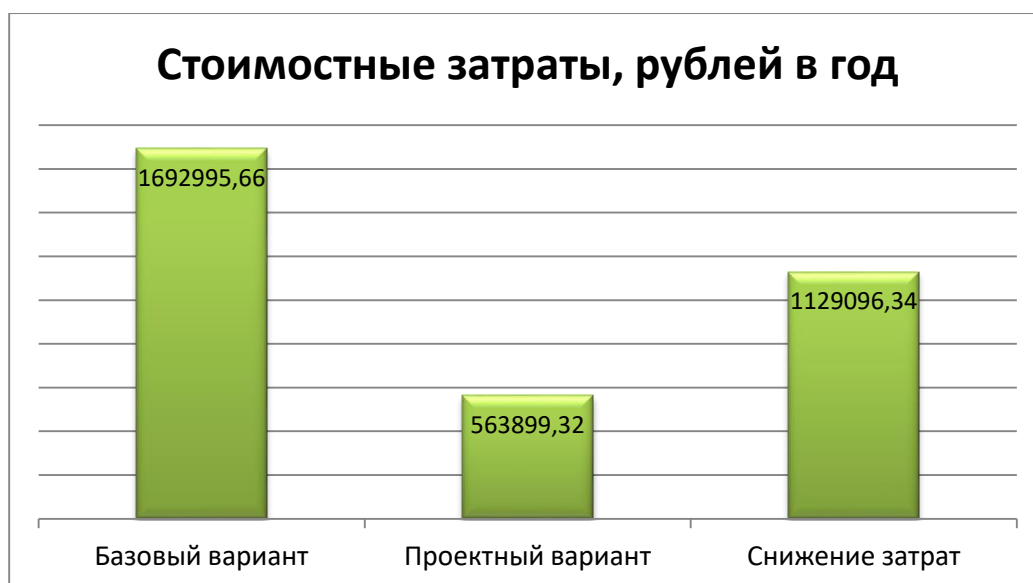


Рис. 3.2 Диаграмма сравнения базового и проектного варианта

Таким образом, данная диаграмма показывает эффективность внедрения системы тестирования веб-приложений.

В третьей главе выпускной квалификационной работы выбрана методика расчета экономической эффективности проекта и определены основные показатели снижения трудовых и стоимостных затрат в результате внедрения разработанной информационной системы.

Заключение

Проектируемая информационная система позволит компании осуществить автоматизацию процесса разработки веб-приложений сотрудниками компании, в том числе удаленными.

Основные задачи выпускной квалификационной работы — анализ деятельности предприятия, выявление существующих недостатков в текущей технологии управления предприятия, разработка автоматизированной системы тестирования веб-приложений в компании ООО «Аутсорсинг Контракт», выполнены.

В ходе выпускной квалификационной работы на основании анализа текущей технологии управления были сформированы цели и назначение разрабатываемого автоматизированного варианта решения данной задачи, произведено сравнение с аналогичными системами решения данной задачи, выбраны способ приобретения и стратегия автоматизации, а также принят ряд проектных решений по техническому, информационному и программному обеспечению.

Для проектируемой системы была построена информационная модель, построены физическая и логическая схема базы данных.

На основании анализа предметной области и данных, полученных путем испытания программного решения, была рассчитана ожидаемая экономическая эффективность проекта. Ее уровень позволяет говорить о рентабельности данного проекта. Было отмечено снижение как стоимостных, так и трудовых затрат. Срок окупаемости затрат на разработку и внедрении системы составит чуть меньше полутора лет.

Разработанная система разработки веб-приложений в компании ООО «Аутсорсинг Контракт» позволяет менеджеру автоматизировать свою работу, в том числе такие операции, как учет проектов, учет их выполнения, мониторинг процесса выполнения, получение отчетов по результатам работы.

Внедрение системы разработки веб-приложений в компании ООО «Аутсорсинг Контракт» позволит компании значительно снизить затраты на

оплату труда персонала, увеличить скорость обслуживания клиентов, а также точность и достоверность получаемых данных.

Разработанная информационная система подлежит интеграции с уже существующими на предприятии программными комплексами, кроме того, несмотря на то, что разрабатывалась для конкретного предприятия, может быть использована и на других, аналогичных фирмах за счет своей универсальности с минимальными изменениями.

В дальнейшем планируется написание программного кода на языке Python.

Список использованных источников

1. Федеральный закон «Об информации, информационных технологиях и о защите информации» N 149-ФЗ
2. ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
3. Алистер Коберн, Современные методы описания функциональных требований к системам, М. , Лори, 2017 г., 288 с.
4. Антониоу, Грос, Хоекстра: Семантический веб, М, ДМК-Пресс, 2016 г., 240 с.
5. Баодин Лю, Теория и практика неопределенного программирования, М., Бином. Лаборатория знаний, 2017 г, 416 с.
6. Барт Бэзинс, Эйми Бэкил, Зеппе Ванден Бруке, Java для начинающих. Объектно-ориентированный подход, М., Питер, 2018 г, 688 с.
7. Березин С.А., Березин Б.А., Начальный курс С и С++, М., Диалог-МИФИ, 2017 г., 288 с.
8. Брайан У. Керниган, Деннис М. Ритчи, Язык программирования С, М., Вильямс, 2017 г, 288 с.
9. Брайан У. Керниган, Роб Пайк, Практика программирования, М., Вильямс, 2017 г., 288 с.
10. Бунаков П.А., Лопатин А. В, Практикум по решению задач на ЭВМ в среде Delphi. Учебное пособие, М., Инфра-М, 2018 г, 304 с.
11. Буравченко О.О. Разработка программного обеспечения: теория и практика. – М.: Академия, 2018. – 281 с.
12. Бэнкс, Порселло: React и Redux. Функциональная веб-разработка, М. Питер, 2018 г., 336 с.
13. Васильев Р.А, Калянов Г.А., Левочкина Г.А., Стратегическое управление информационными системами, М, Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2017 г, 512 с.
14. Волкова В.Н., Системный анализ информационных комплексов. Учебное пособие для ВО, 3-е изд., стер. – М.: Лань, 2020. – 336 с.

15. Вон Вернон, Реализация методов предметно-ориентированного проектирования, М, Вильямс, 2017 г, 688 с.
16. Выговский Л.С., Водяхо А.И. Архитектурные решения информационных систем. Учебное пособие. – М.: Лань, 2017. – 356 с.
17. Грацианова Т.А., Программирование в примерах и задачах, М., Лаборатория знаний, 2018 г, 368 с.
18. Джон Скит, С# для профессионалов. Тонкости программирования, М., Вильямс, 2017 г, 608 с.
19. Джоэл Грас, Data Science. Наука о данных с нуля, М, БХВ-Петербург, 2018, 336 с.
20. Иванова Г.А., Технология программирования, М, КноРус, 2018 г, 336 с.
21. Информационные системы в экономике: учеб. пособие для вузов под ред. Д.В. Чистова. – М.: ИИФРА-М, 2017. - 234 с.
22. Исаев Г.А, Теоретико-методологические основы качества информационных систем, М. Инфра-М, 2018 г. 258 с.
23. Исаев Г.А. Проектирование информационных систем. Учебное пособие. – М.: Омега-Л, 2017. – 424 с.
24. Коноплева В. С., Коноплева И. А. Информационные системы в экономике. Учебное пособие. – М.: Проспект, 2018. – 112 с.
25. Ли Атчисон, Масштабирование приложений. Выращивание сложных систем, М, Питер, 2018 г, 256 с.
26. Основы информационных технологий: учебное пособие / С. В. Назаров, С. Н. Белоусова, И. А. Бессонова [и др.]. - 3-е изд. - Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. - 530 с.
27. Остроух А.В., Интеллектуальные информационные системы и технологии. Монография – М.: Лань, 2018. – 308 с.
28. Панюкова Т.А. Панюков А.С., Языки и методы программирования. Путеводитель по языку С++, М., Ленанд, 2018 г, 216 с.
29. Подбельский В.А, Курс программирования на языке Си, М., ДМК Пресс, 2018 г, 384 с.

30. Стенли Б. Липпман, Жози Лажойе, Заря Э. Му, Язык программирования C++. Базовый курс, М, Вильямс, 2017 г, 1120 с.

31. Федотова Е. Л., Информационные технологии и системы. Учебное пособие. – М.: Форум, 2018. – 352 с.