

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра автоматизированных систем управления (АСУ)

Системный анализ процесса разработки программного обеспечения в
ИТ-компании
Курсовая работа
по дисциплине
«Теория систем и системный анализ»

Студент гр. 431-3

_____ Е.П. Бекиш

«__» _____ 2023 г.

Руководитель

Профессор кафедры АСУ,

д.т.н., доц.

_____ А.А. Захарова

оценка

«__» _____ 2023 г.

Томск 2023

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизированных систем управления (АСУ)

УТВЕРЖДАЮ

Зав. кафедрой АСУ

к.т.н., доцент

_____ В.В. Романенко

01.09.2023

Задание

на курсовую работу по дисциплине
«Теория систем и системный анализ»

Студенту группы 431-3 факультета систем управления

_____ Бекиш Егору Павловичу _____

1. Тема курсовой работы: Системный анализ процесса разработки программного обеспечения в IT-компании.
2. Срок сдачи: 26 декабря 2023 г.
3. Исходные данные: Теория систем и системный анализ: методические указания по выполнению курсовой работы по дисциплине «Теория систем и системный анализ» для студентов направления подготовки 09.03.01 Информатика и вычислительная техника / А. А. Захарова. – Томск: Томск. гос. ун-т систем упр. и радиоэлектроники, 2022. – 25 с. [Электронный ресурс] - Режим доступа: <https://asu.tusur.ru/learning/090301/d27/090301-d27-project.pdf>
4. Содержание работы (перечень вопросов, подлежащих разработке):

- 4.1. Изучение проблем, возникающих в организации типа «ИТ-компания» при осуществлении процесса разработки программного обеспечения.
- 4.2. Разработка базовых моделей системы.
- 4.3. Выявление причин возникновения выбранной проблемы процесса.
- 4.4. Постановка и оценивание целей системы для решения выбранной проблемы.
- 4.5 Разработка и оценка альтернатив решения проблемы выбора варианта управления по критерию Вальда (групповое оценивание).
- 4.6. Разработка программного продукта, реализующего выбор варианта управления по критерию Вальда (групповое оценивание).
5. Перечень графического материала:
 - 5.1. Дерево целей и результаты оценивания целей методом анализа иерархий.
 - 5.2. Алгоритм программы ЭВМ, реализующей выбор варианта управления по критерию Вальда (групповое оценивание).
6. Требования к оформлению работы: в соответствии с ОС ТУСУР 01-2021

Дата выдачи задания 01 сентября 2023 г.

Руководитель:

профессор кафедры АСУ, д.т.н., доцент Захарова А.А. _____

ПОДПИСЬ

Задание принял к исполнению: _____

01.09.2023

ПОДПИСЬ

Оглавление

Введение.....	5
1 КРАТКОЕ ОПИСАНИЕ ОБЪЕКТА ИССЛЕДОВАНИЯ	7
2 МОДЕЛИРОВАНИЕ СИСТЕМЫ.....	13
2.1 Модель взаимосвязи системы с окружающей средой.....	13
2.2 Модели состава и структуры системы.....	15
2.3 Построение дерева причин.....	19
3 ПОСТАНОВКА ЦЕЛЕЙ И ПОИСК РЕШЕНИЙ.....	22
3.2 Оценка целей методом анализа иерархий	23
3.2.1. Описание метода анализа иерархий.....	23
3.2.2. Построение матриц парных сравнений и расчёт локальных приоритетов.	26
3.2.3. Расчет глобальных приоритетов.	28
4 РАЗРАБОТКА И ОЦЕНКА АЛЬТЕРНАТИВ РЕШЕНИЯ ПРОБЛЕМЫ МЕТОДОМ КРИТЕРИЯ ВАЛЬДА (ГРУППОВОЕ ОЦЕНИВАНИЕ).....	31
4.1 Разработка альтернатив достижения цели	31
4.2 Описание и оценивание альтернатив методом критерия Вальда (групповое оценивание)	32
5 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА, РЕАЛИЗУЮЩЕГО МЕТОД КРИТЕРИЯ ВАЛЬДА (ГРУППОВОЕ ОЦЕНИВАНИЕ)	37
5.1 Описание алгоритма решения задачи	37
5.2 Описание средств разработки и тестирование программы	37
Заключение	41
Список использованных источников	43
Приложение А_(обязательное) Дерево целей и результаты оценивания целей методом анализа иерархий	45
Приложение Б_(обязательно) Алгоритм программы ЭВМ, реализующей выбор варианта управления по критерию Вальда (групповое оценивание).....	46

Введение

Объектом исследования является организация типа «ИТ-компания», предметом исследования – процесс разработки программного обеспечения в ИТ-компании.

Сопутствующие данные, которые в современном веке занимают большое количество памяти и их анализ, обработка требует значительное время, поэтому необходимо делать это быстро. Данной задачей занимаются ИТ-компании по разработке ПО. Внушающее количество данных с большим объемом порождает высокий спрос на разработку ПО. В связи с этим, чтобы оставаться на современном рынке, рассматриваемая организация должна сохранять высокий уровень разработки.

Для этого необходимо постоянно анализировать проходящие в организации процессы, выявлять возможные проблемы, актуализировать устаревшие решения и, при необходимости, разрабатывать новые.

Цель курсовой работы – выработка и оценка альтернатив решения проблемной ситуации в процессе разработки программного обеспечения в ИТ-компании на основе методов системного анализа и разработанного программного обеспечения.

Задачи:

- изучение проблем, возникающих в выбранной системе при осуществлении процесса разработки программного обеспечения в ИТ-компании;
- разработка базовых моделей системы;
- выявление причин возникновения выбранной проблемы процесса;
- постановка и оценивание целей системы для решения выбранной проблемы;
- разработка и оценка альтернатив решения проблемы методом выбора варианта управления по критерию Вальда (групповое оценивание);
- разработка программного продукта, реализующего заданный метод оценивания/сравнения альтернатив.

Компетенции, на формирование которых направлена данная курсовая работа:

- способность применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности;

- способность осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.

1 КРАТКОЕ ОПИСАНИЕ ОБЪЕКТА ИССЛЕДОВАНИЯ

В настоящий момент информационные технологии развиваются очень быстро. Данное явление можно заметить в существовании большого количество ИТ компаний. По данным сайта РИА Новости [1], с января 2020 года по май 2023 года, число компаний увеличилось с 178 тыс. до 201 тыс. организаций. Из всего множества ИТ-компаний можно выделить основные их деятельности:

- Разработка программного обеспечения — это процесс создания и модификации программного обеспечения, включающий в себя планирование, проектирование, разработку, тестирование и внедрение программного обеспечения для удовлетворения определенных потребностей пользователя. Целью разработки программного обеспечения является создание качественных, эффективных, безопасных и удобных в использовании программных продуктов.

- Управление потоками баз данных — создание баз данных и информационных ресурсов, формирование и ведение баз данных, включая сбор из одного или более источников, администрирование баз данных, поиск данных, их отбор, сортировка по запросам, предоставление отобранных данных пользователям.

- Обслуживание ИТ-инфраструктур — удаленное управление административными правами доступа, учетными записями пользователей, аппаратной платформой ИТ-инфраструктуры, баз данных, системой мониторинга. Выполнение системных настроек, получение обновлений, их установка и сопровождение установки, обеспечение сетевой доступности, резервное копирование, выполнение плановых и внеплановых восстановлений, осуществление переноса данных.

- Создание серверов — предоставление инфраструктуры для хостинга, web-хостинг, услуги потоковой передачи данных, предоставление прикладного хостинга, выделенный хостинг.

В работе приводится системный анализ IT-компаний, занимающейся разработкой программного обеспечения, используя заказную модель разработки.

Из всего следует, что целью данной системы является получение прибыли за предоставление услуг в разработке программного обеспечения.

Так как IT-компания является самостоятельной организацией, то она подвержена сильному влиянию со стороны внешней среды.

Получается, важным элементом внешней среды IT- компании является клиент, который предоставляет прямое финансирование. Далее клиент передает один из вариантов реализации проекта, это может быть как уже программный продукт, так и техническое задание, чтобы компания смогла, основываясь на данных, управлять процессами продукта. Тем самым более качественно и быстро будет выполнена работа, которая в свою очередь повлечет за собой повторное обращение клиента в IT-компанию и распространит информацию о ней, следовательно добавив клиентскую базу.

Так же не стоит забывать о еще одном влияющем компоненте на систему – конкуренты. То, какие услуги, расценки, клиентская база, известность и т.п. в других компаниях – это оказывает главное влияние на количество клиентов в данной организации [2]. Так же сама компания является конкурентом по отношению к другим, оказывая на остальных то же самое влияние.

Одним из важных компонентов, влияющим на систему, является государственные органы, которые контролируют деятельность компаний. Перечень необходимых услуг представлен на портале Министерства цифрового развития, связи и массовых коммуникаций РФ [3]. Данный орган контролирует, чтобы каждая компания была аккредитована, тем самым числилась в реестре аккредитованных организаций для осуществления деятельности в области информационных технологий. В свою очередь, организацию обеспечивает соблюдение налоговых отчислений в соответствующий государственный орган.

Поставщики тоже являются элементом, оказывающим влияние на систему. Для того, чтобы ИТ-компания существовала нужно помещение, техническое оборудование и т.п. Если рассматривать техническое оборудование, то как правило, фирма заключает контракт с официальным поставщиком по оптимальным для себя условиям и заказывает оборудование, оплачивая все необходимое. Поставщики в ответ отправляют необходимое оборудование, закупленное у производителя.

Для примера приведем ИТ-компанию по разработке программного обеспечения на территорию Томска:

- ИТ-компания “Веб-Киборг” – разработка интернет магазинов, сайтов, техническая поддержка имеющихся проектов [4].
- ИТ-компания “Sibedge” – разработка программного обеспечения для промышленного производства [5].
- ИТ-компания “alex-group” - создании бизнес сайтов, продвижении любых интернет-проектов обслуживание и поддержку сайтов [6].
- ИТ-компания “АВТОСНАБ - 70” – прокладывание и настройка локальной сети, настройка облачных сервисов, кибер-безопасность и защита данных, удаление вирусов и шпионского ПО [7].

Целью процесса разработки программного обеспечения в ИТ-компании является выполнение заказа клиента в поставленные сроки.

Предназначением процесса разработки программного обеспечения является результат необходимый заказчику, включая в себя требования заказчика, качество выполненной работы, как необходимо эксплуатировать и обслуживать ПО в дальнейшем, и положительный имидж и хорошая репутацию сервиса у клиентов.

Следовательно, показателями эффективности разработки проекта будут:

- Количество выполненных заказов на разработку ПО.
- Время разработки ПО.
- Качественно было выполнение работы.

– Размер полученной выручки за год.

Все это отражается на этапах разработки программного обеспечения.

Этапы разработки программного обеспечения:

1. Сбор требований и их анализ.

Самый ответственный, важный этап, от которого зависит успех программного обеспечения. Специалисты собирают первичные данные, что позволяет создать основу. Параллельно анализируются риски, связанные с проектом. Это определяет возможность использования разных технических подходов, в основе которых лежит минимизации финансовых расходов.

Специалисты создают макеты и прототипы, определяют требования к проекту. Следующим шагом данного этапа является документирование требований со стороны клиента. Это дает полную правовую защищенность обеих сторон - разработчика и заказчика. В документе прописываются требования, которым должен соответствовать софт.

Чаще всего для этого разрабатывается SRS (Software Requirement Specification) – документ, в котором содержатся основные требования, которые предъявляются к программному продукту. Разработчикам важно точно выявить желание клиента, определить сроки разработки проекта. Здесь главная проблема - многостраничный список требований. Для их решения необходимо тесное взаимодействие с заказчиком, акцент на высокоуровневых требованиях.

2. Разработка дизайна программного продукта.

Создаваемый софт должен быть не только функциональным, но также удобным, понятным для пользователя. Для этого требуется правильно разработать архитектуру, способ представления программы, его пользовательский интерфейс, графическое решение. Особое внимание нужно уделить дизайну, где ориентируются на Software Requirement Specification.

SRS – документ, в котором закрепляется перечень требований и свойств, которые предъявляются к правильной, корректной работе программы. Разработчиком и дизайнером нужно понять, в какой форме должен быть

представлен продукт. Сделать это непросто. Сам заказчик зачастую не знает этого, полагаясь на опыт, квалификацию программистов.

Для этого обычно каждый из разработчиков предлагает свой подход. После все документируется в Design Document Specification. Далее информация анализируется, выявляются требования и связи архитектурного модуля продукта с внешними модулями. Чтобы добиться успеха, важно иметь в команде грамотных лидеров, способных предложить оптимальную архитектуру на основе опыта выполнения аналогичных проектов.

3. Создание прототипа модели.

На этом этапе создается реальный прототип для поддержки знаний, полученных в процессе быстрого проектирования. Это небольшая низкоуровневая рабочая модель желаемой системы.

4. Разработка кода программного обеспечения.

После того, как разобрались с архитектурой, согласовали функционал, дизайн и концепцию, приступают к разработке софта. Пишут программный код, выполняют сборку, последовательно создают необходимые модули и фичи согласно утвержденному DDS. Практика доказывает, что чем более четкими являются требования в Design Document Specification, тем лучше происходит имплементация.

5. Тестирование полученного продукта.

После разработки софта, специалисты приступают к тестированию. Этот процесс затрагивает все этапы жизненного цикла. Все баги и недочеты фиксируются, регистрируются и отслеживаются. Недочеты исправляются, программный продукт тестируется заново. Это процесс происходит до тех пор, пока готовый информационный продукт не достигнет тех стандартов качества, которые прописаны в SRS.

Здесь начинает активно действовать команда разработчиков, тестировщиков. Главная сложность этого этапа – время, необходимое на выявление причин багов. Поиск ошибок в коде - сложная задача. Тестирование лучше

проводить параллельно с разработкой. Это позволит не возвращаться к ним после запуска ПО.

6. Внедрение и поддержка.

После устранения всех багов, ПО выходит в релиз. Начинается поэтапное внедрение программы согласно выбранной бизнес-стратегии. Изначально софт может быть выпущен в ограниченном сегменте, протестирован в конкретной бизнес-среде. Для этого выполняют тестирование User Acceptance Testing. В его основе - получение реальных отзывов со стороны.

Это позволяет проанализировать обратную связь, увидеть недочеты, произвести улучшение продукта. В дальнейшем, после выпуска продукта на рынок, к работе подключается команда специалистов поддержки.

Количество услуг будет формироваться в том случае, если клиент доверяет организации, в которую обратился и в нашем случае, что на каждом этапе разработки все будет идти в указанные сроки. Но из-за нехватки на рынке труба высоквалифицированных специалистов время разработки будет увеличиваться из чего получаем потерю клиентов, которые перестают финансировать компанию.

Из выше сказанного, можно выделить проблему несоответствия планируемых сроков и бюджета разработки программного обеспечения фактическим.

2 МОДЕЛИРОВАНИЕ СИСТЕМЫ

2.1 Модель взаимосвязи системы с окружающей средой

Для построения модели взаимосвязи системы с окружающей средой определим наименования элементов, входящих в неё:

- процесс разработки программного обеспечения в IT-компании – исследуемый процесс, представлен в данной модели как «чёрный ящик»;
- клиенты – потребители услуг, источник прибыли фирмы;
- поставщики – организации, которые поставляют фирме необходимое оборудование;
- руководство – основываясь на предпочтениях клиента, осуществляется управление компанией, устанавливает сроки выполнения работы и т.д.;
- конкурент – различные IT-компании в том-же населенном пункте, специализирующиеся на разработке программного обеспечения;
- контролирующий государственный орган – Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации, обеспечивающий соблюдение фирмой российского законодательства;

Построенная диаграмма взаимосвязи исследуемой системы с окружающей средой представлена на рисунке 2.1.

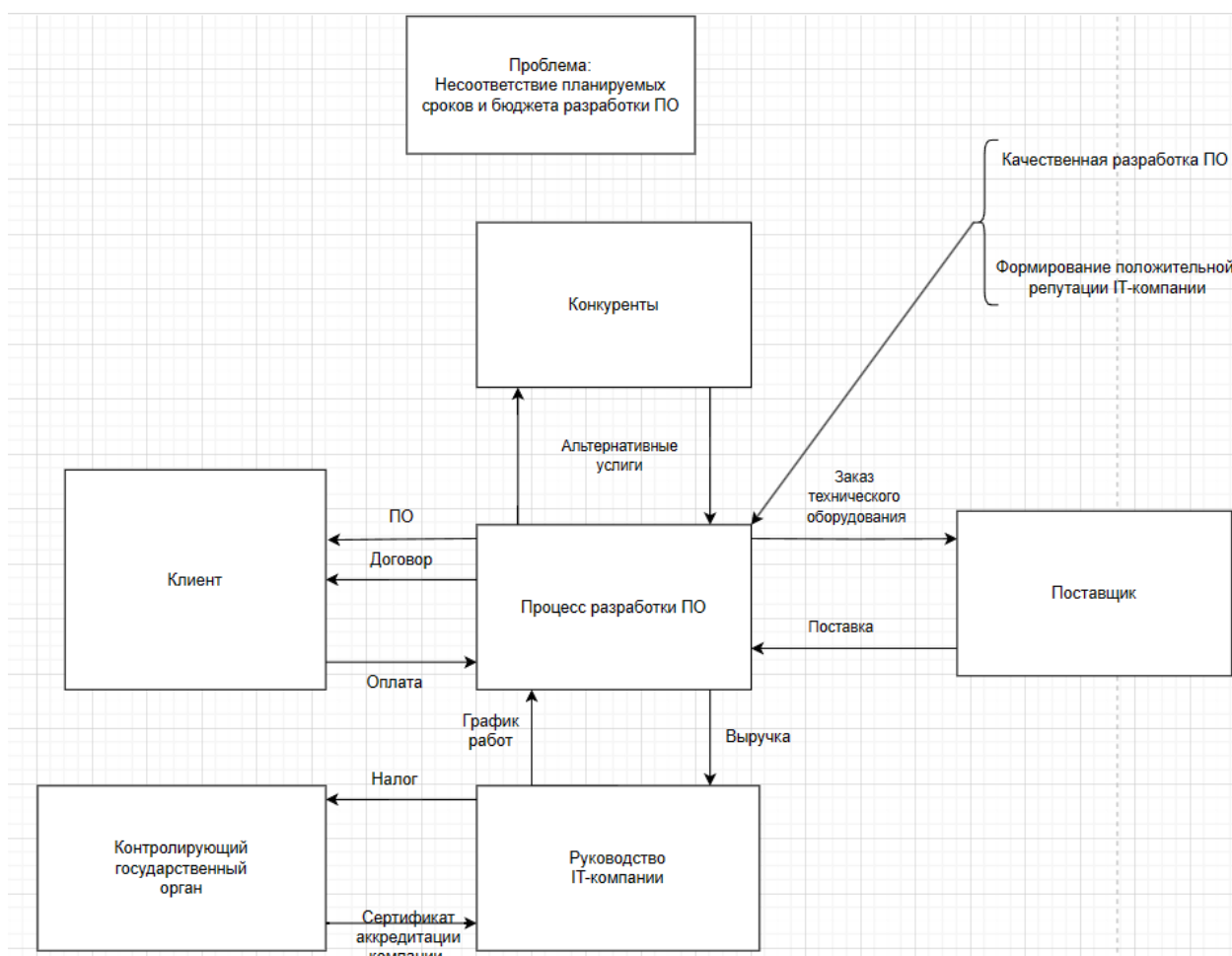


Рисунок 2.1 - Диаграмма взаимосвязи системы с окружающей средой

Построенная модель позволяет выявить заинтересованные стороны анализируемой системы. Таковыми являются непосредственно клиенты и сотрудники IT-компании.

Клиенты являются основным потребителем услуг компании. С точки зрения поставленной проблемы, они могут влиять на качество разрабатываемого продукта несколькими способами. Большое количество клиентов, а, как следствие, большое количество поступающих заявок на разработку ПО, может увеличить срок ожидания их выполнения, т.к. каждая разработка занимает определённое время, вследствие чего может возникнуть очередь. Также на разработку влияет сложность функциональных требований продукта, иначе говоря, его масштабы решения каких-либо задач.

Сотрудники компании занимаются разработкой, непосредственно выполняют функциональные и нефункциональные требования, обеспечивая надлежащее качество ПО. Скорость выполнения работ зависит от сложности поставленных требований клиента, квалификации самих разработчиков.

Критерии, по которым каждый может оценить уровень системы в сравнении с его представлением об «идеале», а также соответствующие им системы оценивания приведены в таблице 2.1.

Таблица 2.1 – Критерии оценивания системы

Критерий	Ед. изм. и/или возможные значения	Система оценивания
1. Клиент 1.1 Время затраченное на разработку ПО	Месяц	1-2 – «отлично»; 3-4 – «хорошо»; 4-5 – «удовлетворительно»; более 6 – «неудовлетворительно».
1.2 Качество выполнения программного продукта	Балл, Удовлетворительное – неудовлетворительное	6 – 10 – «удовлетворительное»; менее 6 баллов – «неудовлетворительное».
2. Сотрудник компании 2.1 Размер заработной платы	Тыс. руб.	60-80 – «неудовлетворительный»; 80-100 – «приемлемый»; более 100 – «отличный».
2.2 Уровень сложности разработки ПО	Балл	1-5 – «высокий» 6-8 – «средний» 9-10 – «низкий»

2.2 Модели состава и структуры системы

Для построения модели состава системы выполним декомпозицию системы по двум основаниям.

Первым основанием является жизненный цикл разработки ПО. В данном случае рассматриваются все её этапы – от приёма заявки клиента на разработку ПО до его передачи клиенту. Далее каждый этап жизненного цикла декомпозируется по технологическим этапам.

Жизненный цикл разработки ПО состоит из трех последовательных этапов: приёма заявки клиента на разработку ПО, решение поставленных задач, эксплуатация ПО.

Этап приёма заявки клиента на разработку ПО состоит из процедур формирования команды и постановки задач. Основываясь на тех функциональных требованиях, которые хочет получить клиент, выбираем специалистов для их реализации. Как только мы сформировали команду необходимо распределить и поставить задачи. Данный этап может оказывать сильное влияние на поставленную проблему, т.к. контроль качества разработки ПО лежит исключительно на разработчиках и есть вероятность, что квалификация специалистов напрямую влияет на качество ПО.

На этапе решения поставленных задач пишется программный код, далее проводится тестирование всех функциональных требований, чтобы перейти к нефункциональным. Главная сложность этого этапа – время, необходимое на выявление причин багов. Поиск ошибок в коде - сложная задача. После тестирования проводим отладку полученных ошибок и багов, чтобы передать клиенту полностью работающий продукт.

Теперь, когда ПО полностью выполняет функционал его нужно запустить в работу, оценить качество проделанной работы разработчиков. То есть необходимо связаться с клиентом для передачи ПО. На данном этапе может представлять затруднение в случаях, когда не получается вовремя выйти с клиентом на связь и передать ему ПО.

Также, поскольку рассматривается ИТ-компания типа разработки ПО, уровень специалистов можно принять за одинаково высокий и убрать вышеописанные возможные негативные факторы на этапе решения поставленных задач.

Соответствующая графическая интерпретация модели состава представлена на рисунке 2.2.

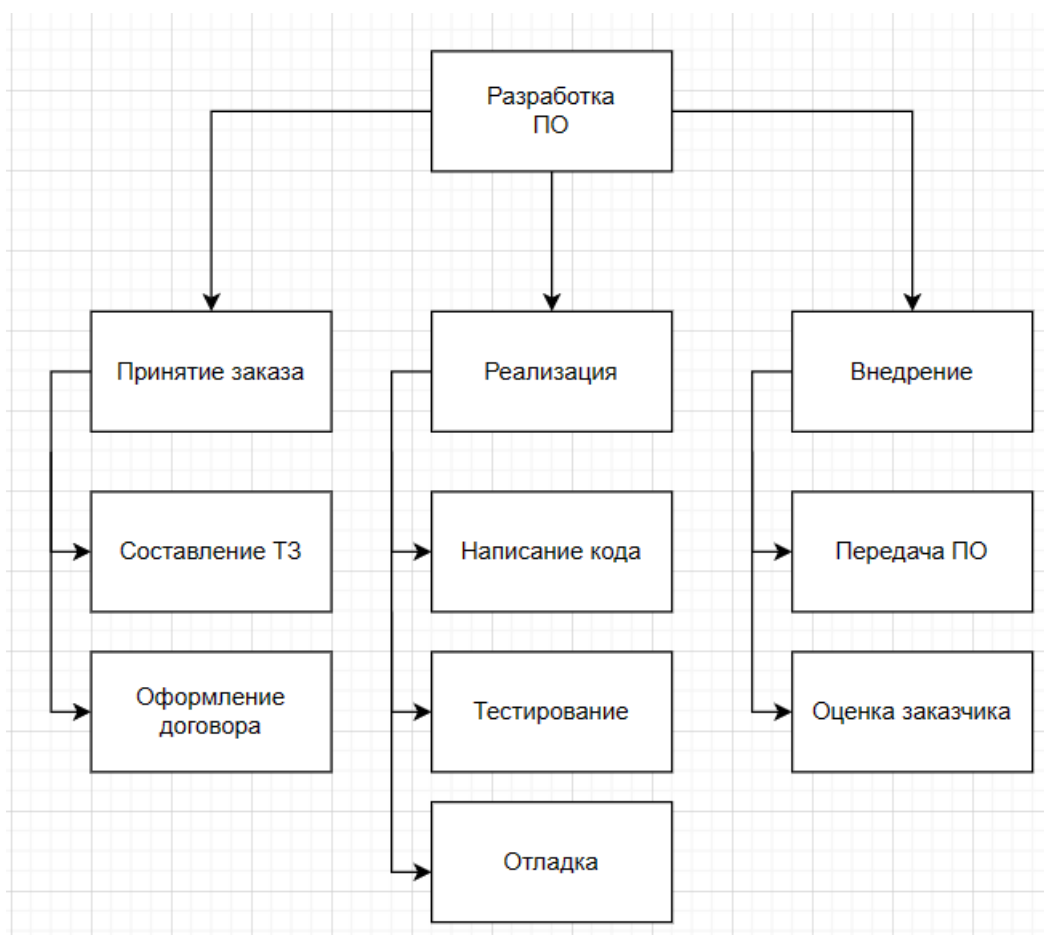


Рисунок 2.2 – Графическая интерпретация модели состава

В таблице 2.2 приведены структурные элементы деятельности рассматриваемой системы.

Таблица 2.2 – Структурные элементы системы

Подсистема	Предмет деятельности	Средство исполнения	Исполнитель	Регламент	Конечный продукт
Принятие заказа	Заказная модель	Компьютер, MS Word	Исполнитель	Предоставляемые услуги	Договор о разработке ПО
Реализация	Написание программы	Компьютер, ЯП	Разработчики	Функциональные требования	Разработанное ПО
Внедрение	Разработанное ПО	Компьютер	Разработчики	Описание работы ПО	Внедренное ПО, результаты оценки качества

Далее на рисунке 2.3 приведена диаграмма, отражающая связи между подсистемами, полученными в результате декомпозиции системы по

жизненному циклу. Соответственно, на рисунках 2.4, 2.5, 2.6 приведены аналогичные диаграммы в результате декомпозиции каждого из этапов второго уровня декомпозиции.

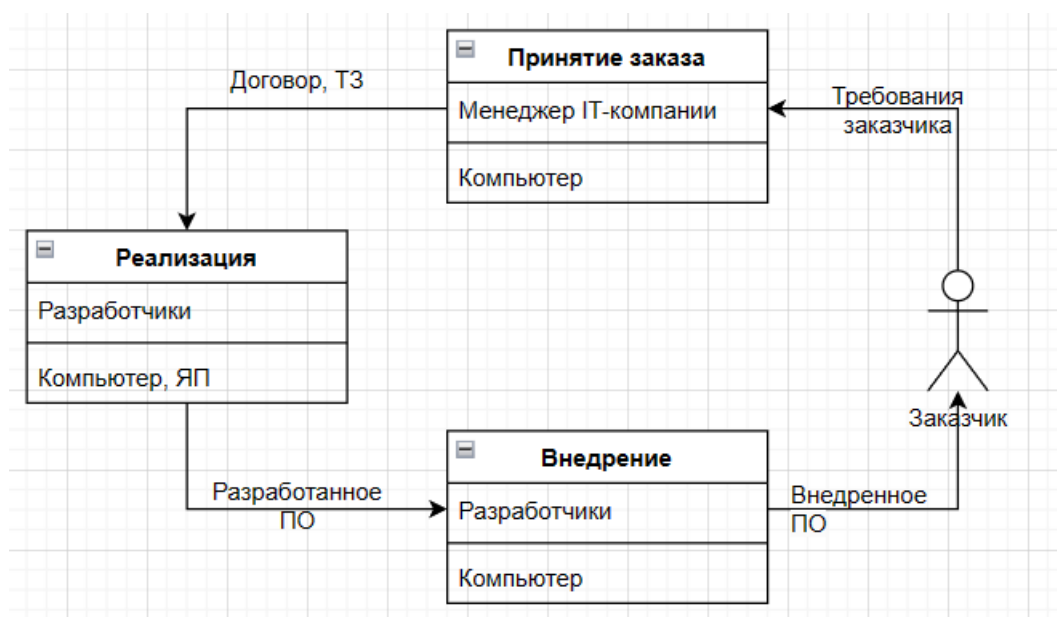


Рисунок 2.3 – Диаграмма взаимосвязи подсистем

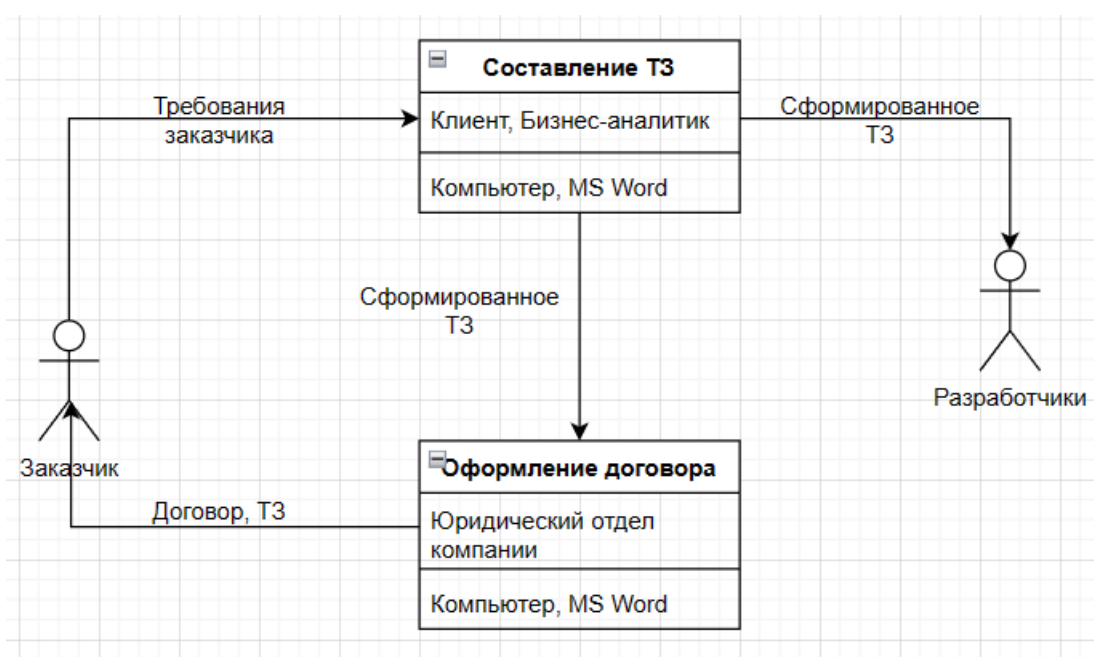


Рисунок 2.4 – Диаграмма взаимосвязи подсистемы «Принятие заказа»

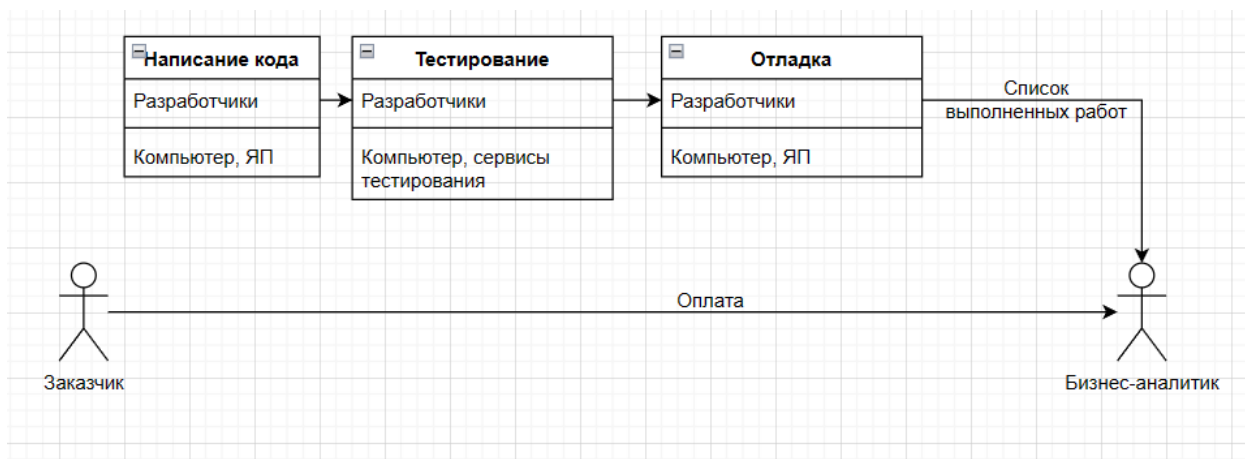


Рисунок 2.5 – Диаграмма взаимосвязи подсистемы «Реализация»

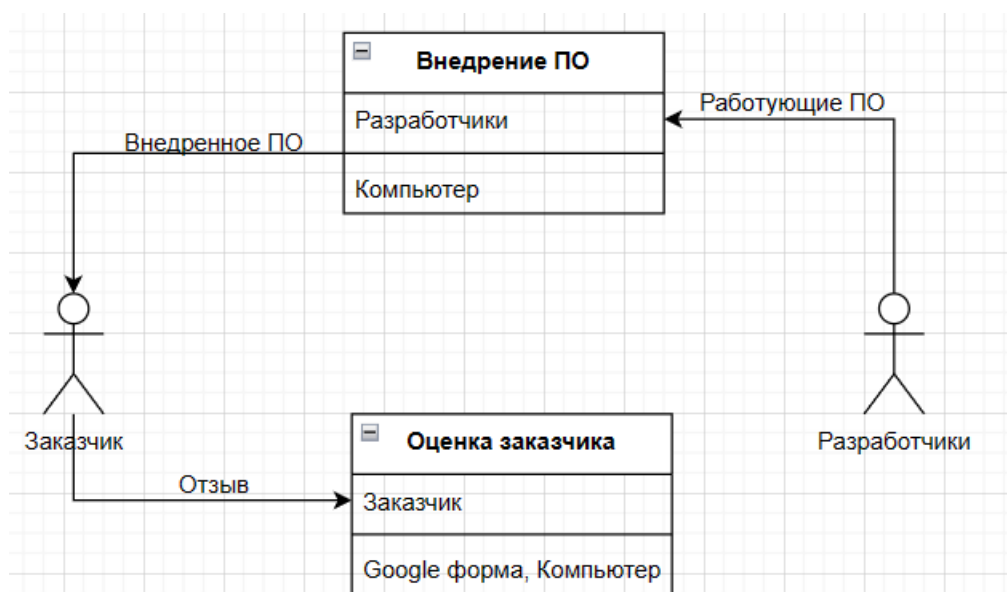


Рисунок 2.6 – Диаграмма взаимосвязи подсистемы «Внедрение ПО»

2.3 Построение дерева причин

Следующим этапом работы является выявление возможных причин проблемы длительного срока ожидания клиентами выполнения услуг.

В ходе анализа состояния системы и среды были выявлены следующая «подпроблема»: большое количество разрабатываемого ПО, которое задерживается из-за невыполнения поставленных в срок задач.

В свою очередь, кадровый дефицит может быть вызван в результате наплыва низкоквалифицированных специалистов. Дальнейшая проработка проблемы привела к следующим выводам:

- Некачественная коммуникация между IT-командой и клиентом. Например, клиент может долго не давать обратную связь, избыточно контролировать команду, не понимать цели проекта и постоянно менять требования.
- Неполная команда разработки на старте. Когда на проекте не хватает рук, приходится чем-то жертвовать, чтобы уложиться в сроки.
- Нехватка у IT-команды опыта на конкретном проекте. Если команда работает с чем-то новым, бывает сложно учесть все нюансы.

Тем самым влечет за собой большой кадровый дефицит специалистов, которые не в силах выполнять поставленные задачи. Анализ проблемы привел к таким результатам:

- Компании не предоставляют повышение квалификации, не проводят внутренние хакатоны, так же, заведомо, не отправляя на другие проводимые от других компаний или иных организаций.
- Компании нанимают заведомо некомпетентных людей. В целях привлечения компетентных кадров растут зарплаты.
- Требуется подготовка специалистов, обладающих современными знаниями и практическими навыками аналитических, статистических и эконометрических методов исследования, анализа социально-экономических явлений и процессов с применением цифровых технологий.

Построенное дерево причин изображено на рисунке 2.7.

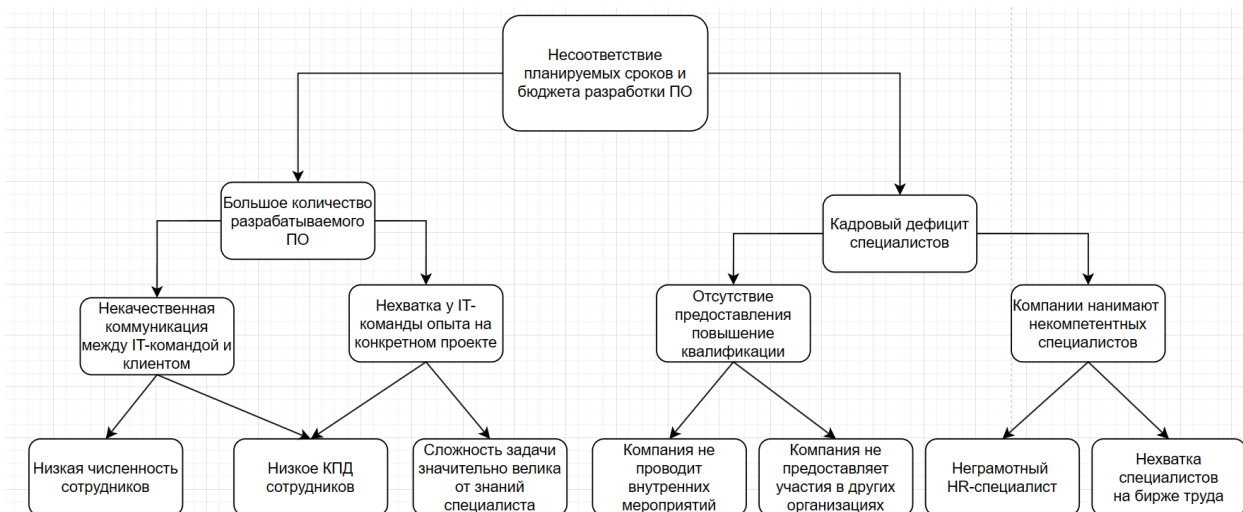


Рисунок 2.7 – Дерево причин

Для оценки важности причин воспользуемся методом ранжирования. Результаты оценивания приведены в таблице 2.3.

Таблица 2.3 – Оценивание важности причин методом ранжирования

Причина	Ранг
Низкая численность сотрудников	5
Низкое КПД сотрудников	2
Сложность задачи значительно велика от знаний специалиста	7
Компания не проводит внутренних мероприятий	4
Компания не предоставляет участия в других организациях	3
Не грамотный HR-специалист	6
Нехватка специалистов на бирже труда	1

3 ПОСТАНОВКА ЦЕЛЕЙ И ПОИСК РЕШЕНИЙ

3.1 Построение дерева целей

Для построения дерева целей возьмём за основу дерево причин. Глобальной целью является разработать качественное программное обеспечение. Этого можно добиться двумя способами – сократить количества разрабатываемого ПО и повысить опыт и квалификацию сотрудников.

Сокращения большого количества разрабатываемого ПО, которое задерживается из-за невыполнения поставленных в срок задач можно добиться путём улучшения организации построения приоритетных этапов разработки. Или же можно повысить общую производительность сотрудников, чтобы за одно и то же время выполнялось больше задач на определённом этапе разработки.

Повышение производительности может заключаться в создании системы набора и подготовки персонала, либо в создании системы мотивации персонала, чтобы у работников появилась личная заинтересованность в повышении эффективности процесса.

Добиться сокращением времени разработки ПО можно, если создать такую систему, которая будет заниматься учётом и отслеживанием сроков на конкретном этапа разработки. Тем самым, эта задача не накладывает на разработчиков всю ответственность на выполненные этапы, за все будет полностью отвечать автоматизированная система, которая будет показывать, какой из выполняемых этапов сейчас в приоритете по времени. Следовательно, разработка ПО будет проводиться в срок. Еще подходит система подготовки сотрудников, т.к. при условии слаженности коллектива обеспечивается лучшее распределение выполняемых задач между сотрудниками компании – знание сильных и слабых сторон коллег позволяет распределять этапы разработки так, чтобы эффективность работы каждого была максимальна.

Отметим, что имея в компании опытных и квалифицированных специалистов, время на этапах разработки сокращается, приводя к выполнению проекта в поставленные сроки.

Согласно дереву причин, длительная разработка вызвана низкой эффективностью взаимодействия с клиентом либо отсутствием опытных специалистов. Соответственно, есть вариант нахождения альтернативных специалистов, которые будут иметь больше опыта, чтобы за поставленные временные рамки разработать ПО.

Увеличение квалифицированных сотрудников. Данный вариант может повлечь определённые расходы, т.к. разработкой будут заниматься сотрудники, привлеченные из других компаний. Также здесь подходит вариант поиска альтернативных компаний, которые могут предоставлять свои услуги в более короткий период и с заранее оговорённым сроком.

Построенное по вышеописанной логике дерево целей представлено на рисунке 3.1.



Рисунок 3.1 – Дерево целей

3.2 Оценка целей методом анализа иерархий

3.2.1. Описание метода анализа иерархий.

Метод анализа иерархий (МАИ), предложенный Томасом Саати, использует методологию дерева целей, т.е. также основан на формировании иерархии целей и средств по типу слоёв. Данный метод предназначен для выбора

средств решения сложной многофакторной проблемы и состоит в декомпозиции цели на всё более простые составляющие [8].

Метод состоит из пяти основных этапов:

- 1) иерархическое представление проблемы;
- 2) построение множества матриц парных сравнений;
- 3) определение векторов локальных приоритетов;
- 4) проверка согласованности полученных результатов;
- 5) вычисление глобальных приоритетов.

Этап иерархического представления на данный момент уже пройден – дерево целей построено. Для последующих шагов кратко пронумеруем каждый элемент иерархии. Результат представлен на рисунке 3.2.

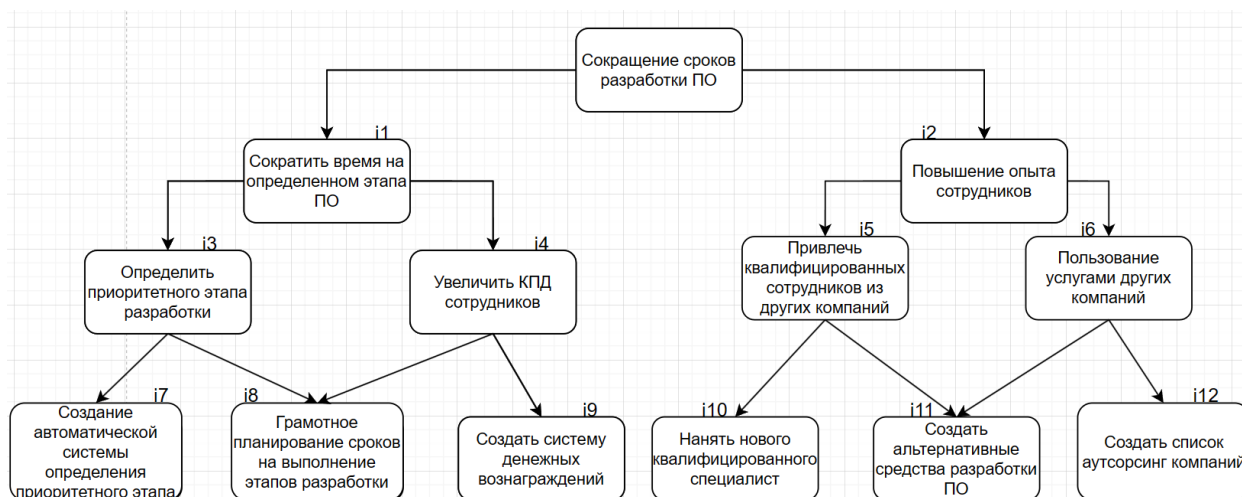


Рисунок 3.2 – Иерархия целей

Следующим этапом идёт построение матриц парных сравнений для каждой совокупности элементов, связанных с одним вышестоящим элементом. Для полученного дерева целей потребуется построить: одну матрицу, соответствующую второму уровню иерархии, для сравнения двух основных направлений возможного решения проблемы; две матрицы, соответствующие третьему уровню, для детального сравнения каждого из направлений в отдельности, и четыре матрицы, соответствующие четвёртому

уровню иерархии, для оценки влияния сценариев развития сервиса на каждое из «поднаправлений». При оценивании каждой сравниваемой пары ставится балльная оценка от 1 до 9 в зависимости от того, какой элемент пары доминирующий.

На основании каждой матрицы парных сравнений формируются наборы локальных приоритетов. Они отражают ценность, важность, силу влияния сравниваемых элементов по отношению к направляющему элементу – элементу вышестоящего уровня. Чтобы вычислить эти приоритеты, нужно для каждой матрицы вычислить множество собственных векторов и нормализовать результат к единице – так получится вектор приоритетов. Так как задача вычисления собственных векторов довольно трудоёмкая, мы воспользуемся приближённым методом – вычислением геометрического среднего.

Для определения приоритетных целей по методу анализа иерархий строятся матрицы парных сравнений для каждого уровня, после чего определяются локальные приоритеты для каждого элемента дерева целей.

Значения в матрице парных сравнений определяются субъективно по превосходству одного объекта относительно другого по 9-бальной шкале, где значение 1 представляет собой равнозначность, а 9 – максимальное превосходство.

Значение элемента собственного вектора V_i вычисляется по формуле:

$$V_i = \sqrt[n]{\prod_{j=1}^n a_{ij}} \quad (3.1)$$

где a_{ij} – значение элемента матрицы;

$i = \overline{1, n}, j = \overline{1, n}$ – соответственно номера строк и столбцов матрицы;

n – количество элементов.

Вектор локальных приоритетов рассчитывается путем нормирования значений элементов собственного вектора по формуле:

$$V_i^{\text{норм}} = \frac{V_i}{\sum_{i=1}^n V_i}, \quad (3.2)$$

где $V_i^{\text{норм}}$ – i -тый элемент вектора локальных приоритетов.

Индекс согласованности (ИС) определяется по формуле:

$$\text{ИС} = (\lambda_{\max} - n) / (n - 1), \quad (3.3)$$

где λ_{\max} – наибольшее собственное значение матрицы, рассчитывается по формулам:

$$\lambda_{\max} = \sum_{i=1}^n \lambda_i \quad (3.4)$$

$$\lambda_i = \sum_{j=1}^n a_{ij} * V_j^{\text{норм}} \quad (3.5)$$

Отношение согласованности (ОС) по формуле:

$$\text{ОС} = \frac{\text{ИС}}{\text{СС}}, \quad (3.6)$$

где СС – случайная согласованность матрицы (определяется по таблице для заданного значения n).

Для проверки согласованности таким путём, нужно сравнить ИС с индексом, вычисленным для абсолютно несогласованной матрице, полученной при случайном выборе суждений. Для получения конечного результата ИС нужно разделить на число, соответствующее случайной согласованности матрицы того же порядка – получится отношение согласованности (ОС).

Финальный шаг – вычисление глобальных приоритетов. Они рассчитываются, начиная со второго уровня вниз. Локальные приоритеты элементов умножаются на глобальные приоритеты элементов вышестоящего уровня - направляемых. Если направляемых несколько, то находится сумма взвешенных приоритетов по всем направляемым элементам. Процедура продолжается до нижнего уровня.

3.2.2. Построение матриц парных сравнений и расчёт локальных приоритетов.

Сравнив полученные элементы иерархии целей методом попарных сравнений, были рассчитаны локальные приоритеты для каждого из элементов иерархии относительно каждого родительского (направляющего элемента).

Индекс согласованности и отношение согласованности были также рассчитаны. Результаты приведены в таблицах 3.1 – 3.7.

Таблица 3.1 – Матрица парных сравнений для цели «Сокращение сроков разработки ПО»

	Уменьшить очередь разработки ПО	Повышение опыта сотрудников	Локальный приоритет
Уменьшить очередь разработки ПО	1,00	0,20	0,167
Повышение опыта Сотрудников	5,00	1,00	0,833
Индекс согласованности			0,00
Отношение согласованности			0,00

Таблица 3.2 – Матрица парных сравнений для цели «Уменьшить очередь разработки ПО»

	Определить очередь разработки	Увеличить КПД сотрудников	Локальный приоритет
Определить очередь разработки	1,00	0,50	0,333
Увеличить КПД сотрудников	2,00	1,00	0,667
Индекс согласованности			0,00
Отношение согласованности			0,00

Таблица 3.3 – Матрица парных сравнений для цели «Повышение опыта сотрудников»

	Найм квалифицированных сотрудников из других компаний	Пользование услугами других компаний	Локальный приоритет
Найм квалифицированных сотрудников из других компаний	1,00	9,00	0,9
Пользование услугами других компаний	0,11	1,00	0,1
Индекс согласованности			0,00
Отношение согласованности			0,00

Таблица 3.4 – Матрица парных сравнений для цели «Определить очередь разработки»

	Создание автоматической системы сроков разработки	Повышение HR-отдела	Локальный приоритет
Создание автоматической системы сроков разработки	1,00	0,17	0,143
Повышение HR-отдела	6,00	1,00	0,857

Окончание таблицы 3.4

Индекс согласованности	0,00
Отношение согласованности	0,00

Таблица 3.5 – Матрица парных сравнений для цели «Увеличить КПД сотрудников»

	Повышение HR-отдела	Система денежных вознаграждений	Локальный приоритет
Повышение HR-отдела	1,00	0,14	0,125
Система денежных вознаграждений	7,00	1,00	0,875
Индекс согласованности			0,00
Отношение согласованности			0,00

Таблица 3.6 – Матрица парных сравнений для цели «Найм квалифицированных сотрудников из других компаний»

	Найти нового специалиста	Альтернативные средства разработки ПО	Локальный приоритет
Найти нового специалиста	1,00	9,00	0,9
Альтернативные средства разработки ПО	0,11	1,00	0,1
Индекс согласованности			0,00
Отношение согласованности			0,00

Таблица 3.7 – Матрица парных сравнений для цели «Пользование услугами других компаний»

	Альтернативные средства разработки ПО	Создать систему пользования услуг	Локальный приоритет
Альтернативные средства разработки ПО	1,00	6,00	0,857
Создать систему пользования услуг	1,67	1,00	0,143
Индекс согласованности			0,00
Отношение согласованности			0,00

3.2.3. Расчет глобальных приоритетов.

Рассчитаем глобальные приоритеты всех элементов иерархии согласно правилам, описанным в пункте 3.2.1. Результаты представлены в таблице 3.8.

Таблица 3.8 – Значения глобальных приоритетов

Уровень иерархии	Наименование элемента иерархии	Глобальный приоритет
1	Сокращение сроков разработки ПО	1,00
2	Уменьшить очередь разработки ПО	0,167
2	Повышение опыта сотрудников	0,833
3	Определить очередь разработки	0,056
3	Увеличить КПД сотрудников	0,111
3	Найм квалифицированных сотрудников из других компаний	0,750
3	Пользование услугами других компаний	0,083
4	Создание автоматической системы сроков разработки	0,008
4	Повышение HR-отдела	0,062
4	Система денежных вознаграждений	0,097
4	Найти нового специалиста	0,675
4	Альтернативные средства разработки ПО	0,146
4	Список аутсорсинговых компаний	0,012

В результате совершённых действий можно установить, что для устранения исходной проблемной ситуации следует двигаться в направлении повышения опыта сотрудников путем найма, квалифицированных специалистом. Его и выберем в качестве приоритетного.

Для простоты понимания отобразим все полученные приоритеты на иерархии целей, как показано на рисунке 3.3. Глобальные приоритеты обозначены чёрным цветом над каждым элементом.

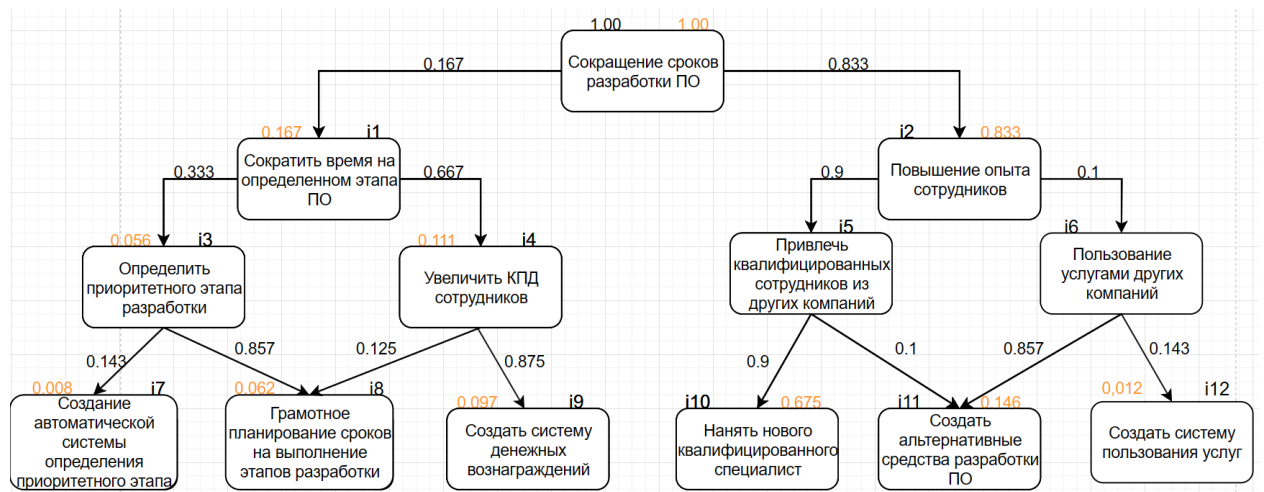


Рисунок 3.3 – Результат выявления приоритетов

4 РАЗРАБОТКА И ОЦЕНКА АЛЬТЕРНАТИВ РЕШЕНИЯ ПРОБЛЕМЫ МЕТОДОМ КРИТЕРИЯ ВАЛЬДА (ГРУППОВОЕ ОЦЕНИВАНИЕ)

4.1 Разработка альтернатив достижения цели

Выбранную приоритетную цель – «Найм квалифицированного сотрудника» можно достичь следующими путями:

1. **Хэдхантинг** – это процесс активного поиска и привлечения талантливых специалистов из других компаний. Создадим предложение, которое будет привлекательным для кандидатов. Оно должно включать хорошую зарплату, бонусы, возможности для развития и карьерного роста. Обратимся к своим контактам в других компаниях, чтобы узнать о потенциальных кандидатах. По итогам собеседования оценим их необходимые навыки для наших задач. После чего выбираем кандидата, который наилучшим образом соответствует требованиям компании и предлагает наилучшие перспективы для вашей компании [9].

2. **Партнерство** – сотрудничество с компаниями, у которых есть нужные вам специалисты. Можно заключить соглашения о сотрудничестве с другими компаниями, которые могут предоставить вам доступ к их базе данных кандидатов или помочь в поиске кандидатов [10].

3. **Нетворкинг** – участие в хакатонах, конференциях и других мероприятиях может показать сильные стороны тех специалистов, которые решают задачи необходимы компании [11].

4. **Организация стажировки для студентов или выпускников ВУЗов** – это позволит привлечь молодых специалистов, которые еще не имеют опыта работы, но обладают свежими знаниями и готовы учиться. Основываясь на разработки ПО можно разработать программу, которая будет интересной и полезной для студентов. Программа должна включать обучение, наставничество и оценку работы стажеров. С этим учетом у студента, который проявит себя

хорошо в дальнейшем будет место, где он может себя реализовать, а у компании молодой и перспективный сотрудник [12].

5. Рекрутинговые агентства и социальные сети – Можно обратиться в рекрутинговые агентства как оффлайн, так и онлайн для поиска кандидатов. Агентства имеют большую базу данных и могут предложить кандидатов, соответствующих требованиям компании. Так же существуют фрилансеры, которые находятся в активном поиске работы в социальных сетях, что потребует от нанимаемой компании создания HR-отдела для размещений вакансий в социальных сетях, где кандидат может заполнить определенную форму, чтобы компания смогла ее рассмотреть и пригласить на собеседование [13].

Для того, чтобы выбрать наилучший вариант найма, воспользуемся заданным по варианту критерием Вальда в контексте группового оценивания. Так как каждая альтернатива обладает разными эффективностью и сложностью обслуживания, каждый участник группы должен будет провести комплексную оценку каждой альтернативы. В качестве состояний среды будут использоваться возможные внешние условия – нехватка специалистов на рынке труда, соответствие и переизбыток.

4.2 Описание и оценивание альтернатив методом критерия Вальда (групповое оценивание)

Каждый член группы оценивает альтернативы по следующему параметру: затраты.

Экспертная группа состоит из следующих участников:

1. Руководитель IT-компании.
2. IT-специалист.
3. HR-менеджер.

В таблицах 4.1 – 4.3 отображены экспертные оценки вышеописанных параметров.

Обобщенная оценка считается в тыс. рублей и высчитывается по формуле:

$$O = A_j * V_{ij} , \quad (4.1)$$

где A_j – компетенция эксперта;

V_{ij} – значение шкалы;

O – обобщенная оценка.

Таблица 4.1 – Оценка затрат при нехватке специалистов на рынке труда

Варианты реализации	Руководитель ИТ-компании	ИТ-специалист	HR-менеджер	Обобщенная оценка (тыс.руб)
Хэдхантинг	800	800	700	790
Партнерство	600	800	800	720
Нетворкинг	700	600	800	660
Организация стажировки для студентов или выпускников ВУЗов	400	500	300	440
Рекрутинговые агентства и социальные сети	300	400	200	340
Компетентность экспертов	0,4	0,5	0,1	

Таблица 4.2 – Оценка затрат при соответствии специалистов на рынке труда

Варианты реализации	Руководитель ИТ-компании	Тимлид	HR-менеджер	Обобщенная оценка (тыс.руб)
Хэдхантинг	600	700	600	650
Партнерство	200	600	800	460
Нетворкинг	600	400	600	500
Организация стажировки для студентов или выпускников ВУЗов	700	600	700	650
Рекрутинговые агентства и социальные сети	700	800	400	720
Компетентность экспертов	0,4	0,5	0,1	

Таблица 4.3 – Оценка затрат при переизбытке специалистов на рынке труда

Варианты реализации	Руководитель ИТ-компаний	Тимлид	HR-менеджер	Обобщенная оценка (тыс.руб)
Хэдхантинг	500	600	500	550
Партнерство	100	400	500	290
Нетворкинг	300	100	500	220
Организация стажировки для студентов или выпускников ВУЗов	900	600	800	740
Рекрутинговые агентства и социальные сети	800	800	800	800
Компетентность экспертов	0,4	0,5	0,1	

Для получения коэффициента эффективности рабочего процесса необходимо добавить бюджет, который компания готова затратить на найм сотрудников.

Таблица 4.4 – Исходные данные для расчёта коэффициента эффективности

	Бюджет	Нехватка	Соответствие	Переизбыток
Хэдхантинг	1000	790	650	550
Партнерство		720	460	290
Нетворкинг		660	500	220
Организация стажировки для студентов или выпускников ВУЗов		440	650	740
Рекрутинговые агентства и социальные сети		340	720	800

Рассчитывать коэффициент эффективности будем по формуле:

$$k = M - Z_{ij}, \quad (4.2)$$

где M – бюджет;

Z_{ij} – затраты альтернативы для определенного состояния среды.

Таблица 4.4 содержит данные о прогнозируемых коэффициентах эффективности.

Таблица 4.4 – прогнозируемые коэффициенты эффективности

Варианты	Нехватка	Соответствие	Переизбыток
Хэдхантинг	210	350	450
Партнерство	280	540	710
Нетворкинг	340	500	780
Организация стажировки для студентов или выпускников ВУЗов	560	350	260
Рекрутинговые агентства и социальные сети	660	280	200

Оцениваем альтернативы, согласно критерию Вальда, необходимо выбрать из каждой альтернативы минимальное значение. Исходя из этого, альтернатива оценивается по следующей формуле [14]:

$$K(u_i) = \min(K(u_i)), i = 1, \dots, n \quad (4.3)$$

где $K(u_i)$ – значение критерия для альтернативы u_i ;

n – количество альтернатив.

Наилучшим считается вариант, имеющий максимальную эффективность. В таблице 4.6 приведены результаты оценивания по критерию Вальда.

Таблица 4.6. Оценка эффективности альтернатив критерием Вальда

Варианты	Значение критерия Вальда
Хэдхантинг	210
Партнерство	280

Окончание таблицы 4.6

Варианты	Значение критерия Вальда
Нетворкинг	340
Организация стажировки для студентов или выпускников ВУЗов	260
Рекрутинговые агентства и социальные сети	200

Полученные результаты оценивания и рассчитанные значения критерия Вальда для каждой из альтернатив позволили заключить, что наиболее приемлемым вариантом является выбор альтернативы найма сотрудника через нетворкинг.

5 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА, РЕАЛИЗУЮЩЕГО МЕТОД КРИТЕРИЯ ВАЛЬДА (ГРУППОВОЕ ОЦЕНИВАНИЕ)

5.1 Описание алгоритма решения задачи

Разработанный алгоритм для решения задачи выбора варианта управления (альтернативы) по критерию Вальда (с применением группового оценивания) представлен на рисунке 5.1. Далее приведено описание его этапов.

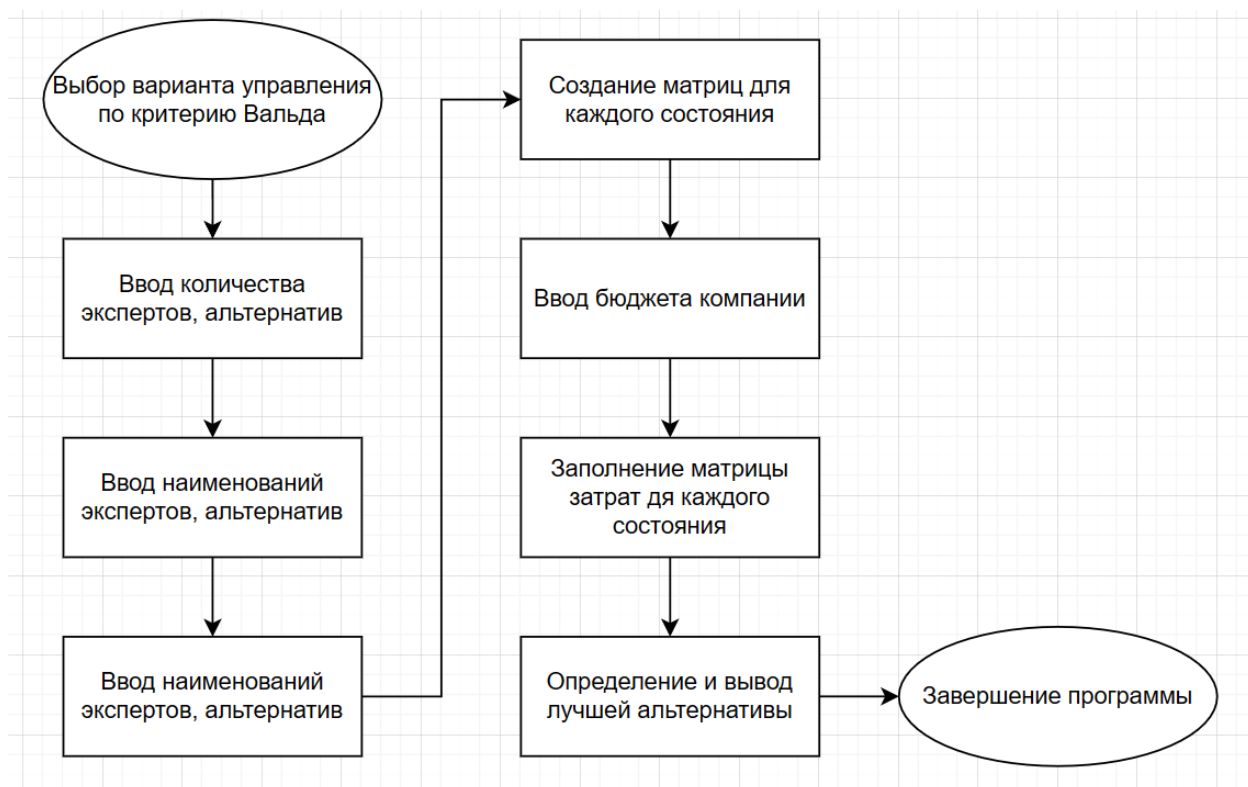


Рисунок 5.1 – Алгоритм решения задачи

5.2 Описание средств разработки и тестирование программы

Решение начинается с определения количества экспертов, которые будут оценивать альтернативы и так же количество альтернатив. Результат представлен на рисунке 5.2.

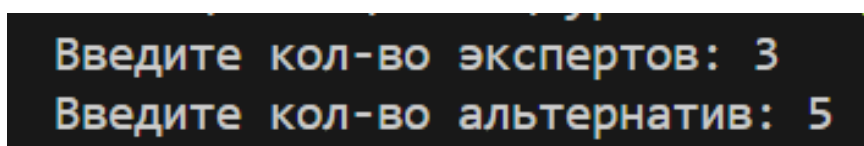


Рисунок 5.2 – Результат ввода количества экспертов и альтернатив

Далее вводим наименование экспертов и альтернатив (рисунок 5.3).

```

Введите эксперта №1: Руководитель
Введите эксперта №2: IT-специалист
Введите эксперта №3: HR-менеджер

Введите альтернативу №1: Хэдхантинг
Введите альтернативу №2: Партнерство
Введите альтернативу №3: Нетворкинг
Введите альтернативу №4: Организация стажировки для студентов или выпускников ВУЗов
Введите альтернативу №5: Рекрутинговые агентства и социальные сети
    
```

Рисунок 5.3 – Результат ввода наименования экспертов и альтернатив

После вводим значение компетенции, чтобы их суммарное значение была равным единице. Если же данные будут введены некорректно, то пользователю нужно ввести их заново как показано на рисунке 5.4.

```

Введите интервал шкалы
Левая граница: 1
Правая граница: 10
Введите коэффициенты компетентности экспертов
P.S. Значение одного эксперта принимает значение от 0 до 1
не включительно и сумма должна быть равна 1

Коэффициенты: 0.4 0.5 0.1
    
```

Рисунок 5.4 – Результат ввода компетенции экспертов

Для дальнейшего расчета коэффициента оптимизации создаем матрицы для каждого состояния. Результат можно увидеть на рисунках 5.5 – 5.8.

Оценивание при плохих условиях			
Альтернатива: Хэдхантинг	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	800	800	700
Альтернатива: Партнерство	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	600	800	800
Альтернатива: Нетворкинг	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	700	600	800
Альтернатива: Организация стажировки для студентов или выпускников ВУЗов	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	400	500	300
Альтернатива: Рекрутинговые агентства и социальные сети	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	300	400	200

Рисунок 5.5 – Результат ввода оценивание при нехватке

Оценивание при обычных условиях			
Альтернатива: Хэдхантинг	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	600	700	600
Альтернатива: Партнерство	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	200	600	800
Альтернатива: Нетворкинг	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	600	400	600
Альтернатива: Организация стажировки для студентов или выпускников ВУЗов	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	700	600	700
Альтернатива: Рекрутинговые агентства и социальные сети	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	700	800	400

Рисунок 5.6 – Результат ввода оценивание при соответствии

Оценивание при наилучших условиях			
Альтернатива: Хэдхантинг	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	500	600	500
Альтернатива: Партнерство	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	100	400	500
Альтернатива: Нетворкинг	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	300	100	500
Альтернатива: Организация стажировки для студентов или выпускников ВУЗов	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	900	600	800
Альтернатива: Рекрутинговые агентства и социальные сети	Эксперты: Руководитель	IT-специалист	HR-менеджер
Оценка:	800	800	800

Рисунок 5.7 – Результат ввода оценивание при переизбытке

Хэдхантинг	Обобщенная оценка
Партнерство	790.0
Нетворкинг	720.0
Организация стажировки для студентов или выпускников ВУЗов	660.0
Рекрутинговые агентства и социальные сети	440.0
Хэдхантинг	Обобщенная оценка
Партнерство	650.0
Нетворкинг	460.0
Организация стажировки для студентов или выпускников ВУЗов	500.0
Рекрутинговые агентства и социальные сети	650.0
Хэдхантинг	Обобщенная оценка
Партнерство	550.0
Нетворкинг	290.0
Организация стажировки для студентов или выпускников ВУЗов	220.0
Рекрутинговые агентства и социальные сети	740.0
	800.0

Рисунок 5.8 – Результат обобщенной оценки для каждого состояния

Потом формируем единую матрицу с бюджетом IT-компании, который задаем (рисунок 5.9.), где в дальнейшем получаем матрицу эффективности (рисунок 5.10).

Введите бюджет, который вы планируете выделить: 1000

Рисунок 5.9 – Результат ввода бюджета IT-компании

	Нехватка специалистов	Соответствие специалистов	Переизбыток специалистов
Хэдхантинг	210.0	350.0	450.0
Партнерство	280.0	540.0	710.0
Нетворкинг	340.0	500.0	780.0
Организация стажировки для студентов или выпускников ВУЗов	560.0	350.0	260.0
Рекрутинговые агентства и социальные сети	660.0	280.0	200.0

Рисунок 5.10 – Результат посчитанных затрат для каждого состояния

Из полученный матрицы выбираем для каждой альтернативы минимальное значение, среди которых выбираем максимальное значение – определяет самый приемлемый вариант как показано на рисунке 5.11.

Хэдхантинг	210.0
Партнерство	280.0
Нетворкинг	340.0
Организация стажировки для студентов или выпускников ВУЗов	260.0
Рекрутинговые агентства и социальные сети	200.0
Самая лучшая альтернатива: Нетворкинг	

Рисунок 5.11 – Результат выборки самой приемлемой альтернативы

Заключение

В процессе выполнения курсовой работы был исследован процесс разработки программного обеспечения в IT-компании. Была выявлена проблема несоответствия планируемых сроков и бюджета разработки программного обеспечения фактическим.

Были построены такие модели системы, как: модель «чёрного ящика», модели состава и структуры системы. В результате построения дерева причин было получено, что основная причина проблемы лежит в неквалифицированных специалистах.

На основе построенного дерева причин было сформировано дерево целей системы для решения проблемы. Эти цели были оценены методом анализа иерархий. В результате оценивания, наиболее приоритетной оказалась цель нанять квалифицированного сотрудника из другой компании. После этого были разработаны альтернативные пути достижения этой цели.

Было разработано пять альтернатив, которые включали в себя способы привлечения квалифицированных сотрудников. Альтернативы были оценены между собой методом выбора варианта управления по критерию Вальда с применением группового оценивания. Был разработан показатель эффективности для сравнения альтернатив между собой. Наилучшей альтернативой Нетворкинг.

Далее, используя среду разработки Visual Studio Code [15] и язык программирования Python [16] был разработан программный продукт, реализующий вышеописанный метод оценки альтернатив. Продукт представляет собой консольное приложение с простым и понятным интерфейсом. В результате взаимодействия с пользователем, на основе статистических данных фирмы, а также на основе оценок экспертной группы выбирается наилучшая альтернатива. Процесс сопровождается выводом на экран показателей эффективности для каждой альтернативы.

На этапе тестирования, в результате ввода в программу данных рассматриваемой в работе фирмы, удалось получить те же значения, что и при проведении непосредственных расчётов.

Полученные результаты можно использовать в контексте рассматриваемой компании, осуществив реализацию соответствующей альтернативы. Также получившийся программный продукт может быть использован для других организаций такого же типа, в которых наблюдается рассматриваемая проблема.

Список использованных источников

1. Число IT-компаний в России [Электронный ресурс]: РИА Новости. URL: <https://ria.ru/20230614/kompanii-1877930925.html> (дата обращения: 01.11.2023).
2. Афендульева Д.Ю. Анализ конкурентоспособности предприятия, основанный на компьютерных технологиях искусственного интеллекта. Механико-математические науки: докл. Саратовская науч.-практ. конф. Саратов 2017. С. 4-7.
3. Аккредитация IT-компаний [Электронный ресурс]: Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации. URL: <https://digital.gov.ru/ru/activity/govservices/1/> (дата обращения 01.11.2023).
4. Веб-студия ВЕБ-Киборг – веб-разработка для решения задач бизнеса [Электронный ресурс]: официальный сайт IT-компании Веб-Киборг. URL: <https://web-kiborg.ru/> (дата обращения 04.11.2023).
5. Разработка ПО для цифровой трансформации бизнеса [Электронный ресурс]: официальный сайт IT-компании Sibedge. URL: <https://sibedge.team> (дата обращения 04.11.2023).
6. Создание сайтов в Томске [Электронный ресурс]: официальный сайт IT-компании Alex-group. URL: <https://ag70.ru> (дата обращения 04.11.2023).
7. ООО "АВТОСНАБ-70" [Электронный ресурс]: официальный сайт IT-компании Autosnab-70. URL: <https://autosnab70.ru> (дата обращения 04.11.2023).
8. Метод анализа иерархий Т. Саати [Электронный ресурс]: онлайн ресурс википедии. URL: https://en.wikipedia.org/wiki/Analytic_hierarchy_process (дата обращения 23.11.2023).
9. Хедхантинг [Электронный ресурс]: онлайн ресурс википедии. URL: <https://en.wikipedia.org/wiki/%D0%A5%D0%B5%D0%B4%D1%85%D0%B0%D0%BD%D1%82%D0%B8%D0%BD%D0%B3> (дата обращения 30.11.2023).

10. Партнерство [Электронный ресурс]: онлайн ресурс википедии. URL: <https://en.wikipedia.org/wiki/Partnership> (дата обращения 30.11.2023).

11. Нетворкинг [Электронный ресурс]: онлайн ресурс. URL: <https://trends.rbc.ru/trends/education/60be769b9a79472d355122af> (дата обращения 30.11.2023).

12. Стажировка для студентов [Электронный ресурс]: онлайн ресурс. URL: <https://edunews.ru/students/primeneniye/stazhirovka-studentov.html?ysclid=lq4tr6sxej61134903> (дата обращения 30.11.2023).

13. Рекрутинговые агентства [Электронный ресурс]: онлайн ресурс википедии. URL: https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D0%B4%D1%80%D0%BE%D0%B2%D0%BE%D0%B5_%D0%B0%D0%B3%D0%B5%D0%BD%D1%82%D1%81%D1%82%D0%B2%D0%BE (дата обращения 30.11.2023).

14. Критерий Вальда [Электронный ресурс]: онлайн ресурс википедии. URL: https://en.wikipedia.org/wiki/Wald%27s_maximin_model (дата обращения 30.11.2023).

15. Visual Studio Code [Электронный ресурс]: онлайн ресурс википедии. URL: https://ru.wikipedia.org/wiki/Visual_Studio_Code (дата обращения 05.12.2023).

16. Python [Электронный ресурс]: онлайн ресурс википедии. URL: <https://ru.wikipedia.org/wiki/Python> (дата обращения 05.12.2023).

Приложение А

(обязательное)

Дерево целей и результаты оценивания целей методом анализа иерархий



Рисунок А.1 – Дерево целей

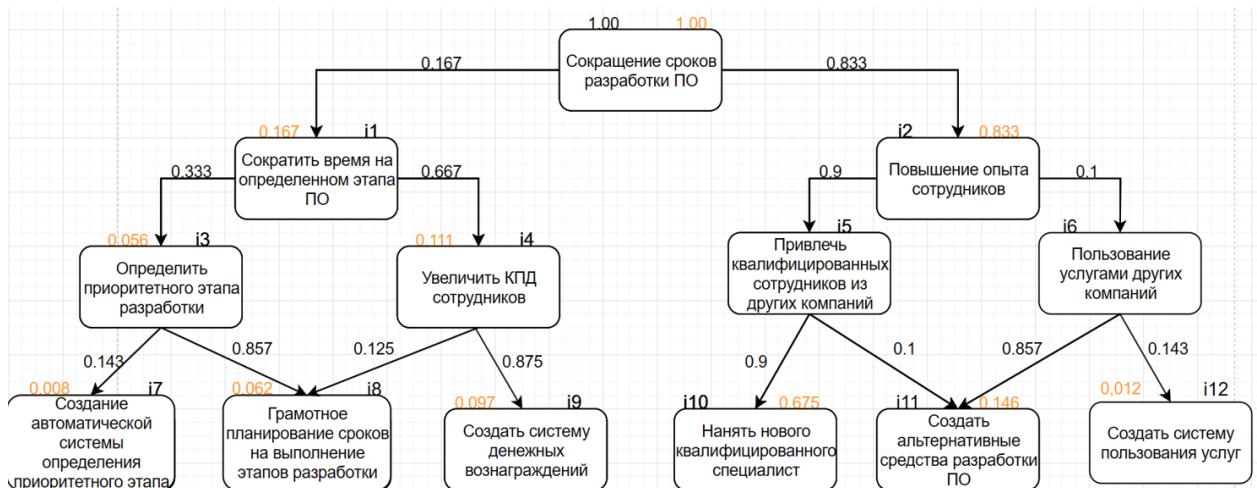


Рисунок А.2 – Результаты выявления приоритетов

Приложение Б

(обязательно)

Алгоритм программы ЭВМ, реализующей выбор варианта управления по критерию Вальда (групповое оценивание)



Рисунок Б.1 – Алгоритм решения задачи