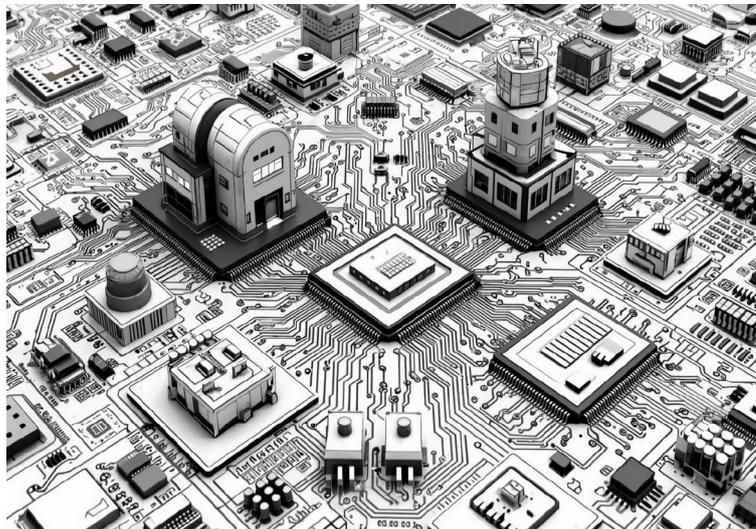


ИТМО

**Д.А. Зубок, Т.Ю. Ивановская, А.В. Маятин,
А.С. Станкевич, С.А. Штумпф**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ПО ВЫПОЛНЕНИЮ НАУЧНО-
ИССЛЕДОВАТЕЛЬСКИХ РАБОТ,
ПРОХОЖДЕНИЮ ПРОИЗВОДСТВЕННОЙ
ПРАКТИКИ И ПОДГОТОВКЕ К
ГОСУДАРСТВЕННОЙ ИТОГОВОЙ
АТТЕСТАЦИИ**



**Санкт-Петербург
2025**

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

УНИВЕРСИТЕТ ИТМО

**Д.А. Зубок, Т.Ю. Ивановская, А.В. Маягин,
А.С. Станкевич, С.А. Штумпф**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО
ВЫПОЛНЕНИЮ НАУЧНО-
ИССЛЕДОВАТЕЛЬСКИХ РАБОТ,
ПРОХОЖДЕНИЮ ПРОИЗВОДСТВЕННОЙ
ПРАКТИКИ И ПОДГОТОВКЕ К
ГОСУДАРСТВЕННОЙ ИТОГОВОЙ
АТТЕСТАЦИИ**

УЧЕБНО-МЕТОДИЧЕСКОЕ ПОСОБИЕ

РЕКОМЕНДОВАНО К ИСПОЛЬЗОВАНИЮ В УНИВЕРСИТЕТЕ
ИТМО

по направлению подготовки 09.04.02, 09.04.03
в качестве Учебно-методического пособия для реализации основных
профессиональных образовательных программ высшего образования
магистратуры

ИТМО

Санкт-Петербург
2025

Зубок Д.А., Ивановская Т.Ю., Маятин А.В., Станкевич А.С., Штумпф С.А., Методические указания по выполнению научно-исследовательских работ, прохождению производственной практики и подготовке к государственной итоговой аттестации – СПб: Университет ИТМО, 2025. – 74 с.

Рецензент(ы): Хлопотов Максим Валерьевич, кандидат технических наук, заведующий лабораторией факультета прикладной информатики ИТМО.

Пособие разработано для магистрантов образовательных программ «Программирование и интернет-технологии» и «Информационные системы бизнеса» факультета информационных технологий и программирования, для помощи в освоении практических навыков профессиональной деятельности, в выполнении и защите магистерской диссертации. Пособие содержит рекомендации по выполнению практической подготовки, оформлению отчетов и подготовке к государственной итоговой аттестации, отражает ноу-хау цикла подготовки, обеспечивающие успешное достижение «нулевой готовности» к производственной деятельности в должностях, соответствующих магистерской квалификации.

ИТМО

ИТМО (Санкт-Петербург) — национальный исследовательский университет, научно-образовательная корпорация. Приоритетные направления: IT и искусственный интеллект, фотоника, робототехника, квантовые коммуникации, трансляционная медицина, Life Sciences, Art&Science, Science Communication. Лидер федеральной программы «Приоритет-2030», в рамках которой реализуется программа «Университет открытого кода». С 2022 ИТМО работает в рамках новой модели развития — научно-образовательной корпорации. В ее основе академическая свобода, поддержка начинаний студентов и сотрудников, распределенная система управления, приверженность открытому коду, бизнес-подходы к организации работы, возможность выбора индивидуальной образовательной траектории.

ИТМО пять лет подряд — в сотне лучших в области Automation & Control (кибернетика) Шанхайского рейтинга. По версии SuperJob занимает первое место в Петербурге и второе в России по уровню зарплат выпускников в сфере IT. Университет в топе международных рейтингов среди российских вузов. Входит в топ-5 российских университетов по качеству приема на бюджетные места. Рекордсмен по поступлению олимпиадников в Петербурге. С 2019 года ИТМО самостоятельно присуждает ученые степени кандидата и доктора наук.

© Университет ИТМО, 2025

© Зубок Д.А., Ивановская Т.Ю., Маятин А.В., Станкевич А.С., Штумпф С.А., 2025

Содержание

Введение.....	4
1. Общая идея цикла практической подготовки.....	5
2. Научно-исследовательская работа (НИР).....	7
2.1. Общие принципы выбора и планирования НИР.....	7
2.2. Второй этап НИР: Теоретическое исследование и моделирование.....	26
2.3. Третий этап НИР: Обоснование методов решения задач магистерской диссертации.....	40
3. Учебная практика.....	42
4. Проектная работа.....	43
4.1. Организация и управление проектами.....	43
4.2. Подготовка отчетов по проектной работе.....	54
5. Производственно-технологическая практика.....	59
5.1. Внедрение информационных систем.....	59
5.2. Подготовка отчетов по производственной практике.....	63
6. Преддипломная практика.....	67
6.1. Подготовка к защите магистерской диссертации.....	67
6.2. Оформление результатов преддипломной практики.....	70
7. Заключение.....	71

Введение

Настоящее учебно-методическое пособие разработано для студентов магистратуры факультета информационных технологий и программирования, обучающихся по направлениям подготовки **09.04.02 "Информатика и программирование"** и **09.04.03 "Прикладная информатика"** на образовательных программах **«Программирование и интернет-технологии»** и **«Информационные системы бизнеса»**. Пособие предназначено для организации и проведения практической подготовки студентов, включающей научно-исследовательскую работу, учебную практику, проектную работу, производственно-технологическую практику и преддипломную практику.

Основная цель пособия — помочь студентам магистратуры освоить практические навыки и компетенции, необходимые для успешной реализации в профессиональной деятельности, а также подготовить их к выполнению и защите магистерской диссертации. Пособие содержит методические рекомендации по выполнению различных этапов практической подготовки, оформлению отчетов и подготовке к государственной итоговой аттестации. Выполнение рекомендаций позволяет получить знания о правилах и особенностях практической работы в производственной среде, развить умение планировать задачи разработки программного обеспечения и оценивать качество их выполнения, сформировать навыки разработки алгоритмов и реализации программного обеспечения систем, практической работы в проектной команде, разработки методик тестирования и тестов ПО

Пособие отражает основные ноу-хау цикла подготовки, обеспечивающие успешное достижение «нулевой готовности» к производственной деятельности в должностях, соответствующих полученной выпускником квалификации магистра в области информационных технологий.

1. Общая идея цикла практической подготовки

В крупных компаниях реального сектора, использующих интернет-решения для автоматизации производственных процессов, коммерческой деятельности, организации массовых удаленных рабочих мест, востребованы **комплексные** знания и навыки внедрения компьютерных систем. Примерами являются внедрение географически распределенных систем, облачных технологий и “умных” систем обработки информации (анализ “больших данных”, машинное обучение, системы поддержки принятия решений), интеграция с предыдущими решениями по автоматизации (“легаси”), учет особенностей эксплуатации таких систем. Образовательные программы нацелены на подготовку **старших / ведущих** ИТ-специалистов для компаний подобного профиля.

Текущее состояние ИТ-ландшафта в таких компаниях требует от старшего / ведущего специалиста знаний и опыта в нескольких **взаимосвязанных** областях, например, понимание архитектур, сетевых и облачных технологий, особенностей внедрения и финансовых аспектов эксплуатации. При этом профильные специалисты из зрелых ИТ-компаний, глубокие профессионалы в конкретной технологии, не всегда могут эффективно решить комплексную задачу с учетом потребностей и особенностей бизнеса.

Для экологичного **сбалансированного** освоения образовательной программы предусмотрены учебная, проектная, проектно-технологическая и преддипломная практики. Обучающимся создаются условия для того, чтобы эти практики были синхронизированы с теоретической образовательной программой и увязаны в логичную последовательность формирования практических навыков будущей деятельности выпускника. Поэтому практический цикл построен по модульному принципу, обеспечивающему поэтапное выполнение большого проекта, с погружением во все аспекты деятельности старшего ИТ-специалиста.

Подготовка начинается в форме **научно-исследовательской работы** (НИР) на предприятии, имеющем статус партнера по практической подготовке, либо в Университете ИТМО. В рамках этого этапа студенты получают навыки поиска и первичного анализа информации, оценки релевантности и значимости информационных источников, формирования обзоров и самостоятельного пополнения знаний на выбранную тему. НИР студентов носит преимущественно практический характер, сосредоточена на исследованиях прикладных методов решения поставленной задачи. Источниками задач являются партнеры-работодатели, лаборатории ИТМО (например, лаборатория теории информации и кодирования), инициатива студентов (при наличии одобрения руководителя ОП и экспертов Факультета). Предполагаемый результат НИР должен помочь принять обоснованное решение о вы-

боре средств и технологий реализации прикладной задачи и спланировать проект разработки.

Проектная работа предполагает участие обучающегося в проектной деятельности предприятия-партнера в форме разработки плана проекта, архитектуры информационной системы, графика проекта реализации или внедрения ПО. Партнер принимает обучающегося в качестве стажера, практиканта, либо по трудовому договору. Здесь формируется навык работы в команде, навык принятия ответственности и понимания основ координации производственного коллектива.

Проектно-технологическая практика включает в себя практическую работу в команде предприятия-партнера для разработки и внедрения информационной системы, получение навыков работы с освоенными технологиями для достижения практического результата. Основной организационной схемой взаимодействия является работа по трудовому договору.

Преддипломная практика посвящена внедрению результатов разработки информационной системы для ее практического применения пользователями и доказательства практической значимости результатов. Результат внедрения является подтверждением практической значимости выпускной квалификационной работы (ВКР).

Как было ранее указано, рекомендуется **сквозная** организация практик, когда несколько видов практики объединены общим проектом, соответствующим по объёму суммарным трудозатратам, включенным в учебный план. Так обучающийся получает возможность освоить разные виды деятельности старшего специалиста по разработке ПО в единой тематике. Для желающих также поддерживается и отдельный подход, когда обучающийся заинтересован в знакомстве с разными областями и предприятиями. Тогда виды практики организуются отдельно, студент может поступить для прохождения практики к разным Партнерам.

2. Научно-исследовательская работа (НИР)

2.1. Общие принципы выбора и планирования НИР

Научно-исследовательская работа (НИР) направлена на развитие навыков самостоятельной исследовательской деятельности, формирование умения анализировать научные публикации, формулировать задачи исследования и выбирать методы их решения.

Задачи этапа практической подготовки в форме НИР:

1. Выбрать тематику исследования, аргументировать актуальность исследования и его перспективность (возможность получения востребованного результата).
2. Сформировать терминологический базис по выбранной тематике.
3. Выявить авторские коллективы и отдельных авторов, занимающихся научными и прикладными исследованиями в рамках выбранной или смежной тематики.
4. Оценить основные результаты, полученные этими авторами, используемые ими методы и средства.
5. Конкретизировать ожидаемый результат своего исследования и перечислить задачи по его достижению (план действий).

Советы.

Как:

... выбрать тематику исследования:

- О чем будет ваше исследование?
- Какую проблему вы хотите решить?
- Какой вопрос вы хотите исследовать?

... аргументировать актуальность исследования:

- Почему эта тема важна?
- Какие проблемы она решает?
- Кому будут интересны результаты исследования?

... аргументировать перспективность исследования:

- Какие новые знания и результаты оно может дать?

- Как оно может улучшить нашу жизнь?
- Какие возможности оно открывает для будущих исследований?

Выбор темы:

- Найди тему, которая тебе интересна.
- Убедись, что по этой теме есть достаточно информации.
- Сформулируй четкий вопрос, на который ты хочешь ответить своим исследованием.

Актуальность:

- Объясни, почему твоя тема важна и кому она может быть полезна.
- Покажи, что твое исследование сможет решить проблему или ответить на важный вопрос.

Перспективность:

- Расскажи, какие новые знания и открытия может принести твое исследование.
- Объясни, как оно может улучшить нашу жизнь или открыть новые возможности для будущих исследований.

Рекомендуется анализировать **публикации** (именно публикации, не случайные страницы из интернета) с целью:

- формирования стека технологий для решения поставленной задачи,
- выбора и обоснования методов разработки, шаблонов проектирования, каркасов и библиотек, средств тестирования и т.д.

Предметом изучения может также являться оценка существующего опыта разработки, тестирования и внедрения информационных систем и их компонентов, имеющих назначение, близкое к выбранной тематике. Предметом изучения могут быть описания и модели прикладных и информационных процессов, для автоматизации которых предназначены разрабатываемые компоненты или системы.

Для обеспечения соответствия инженерной сложности работы профи-

лю образовательной программы рекомендуется понимать под «информационной системой» в целях проводимого исследования аппаратно-программный комплекс, содержательно удовлетворяющий следующим свойствам:

- автоматизирует реальный бизнес-процесс;
- содержит несколько (не менее трех) отдельных нетривиальных компонентов с точки зрения программной архитектуры (например: клиент, сервер, хранилище данных, платформа интеграции, интеллектуальный фильтр, совокупность процедур серверной обработки данных);
- существенно интегрирован со сторонними системами для передачи данных или управляющих воздействий;
- реализует нетривиальные алгоритмы обработки данных;
- имеет дело с промышленными объемами данных.

Тематики, соответствующие направлениям подготовки образовательных программ:

- разработка и реализация алгоритмов / подходов к обработке информации;
- проектирование и создание хранилищ данных;
- разработка и внедрение систем автоматизации управления;
- сбор, обработка и исследование данных, аналитические системы;
- адаптеры протоколов, системы преобразования потоков данных, интеграции;
- автоматизация производственного процесса;
- автоматизация программирования как такового, системы анализа кода, генерации, проверки, тестирования и развертывания;
- практическая реализация программных комплексов для конкретных бизнес-задач, включая подбор, интеграцию, развертывание и эксплуатацию компонентов;
- реализация встраиваемых систем для умных устройств;
- математическое моделирование физических процессов, компьютерная поддержка научных исследований;
- иные тематики, для изучения которых потребуется применить знания

в областях теории информации и теории алгоритмов, проектировании программного обеспечения, программировании, администрировании и развертывании информационных систем, их эксплуатации.

В образовательных программах, ориентированных на бизнес-анализ, акцент надлежит делать на исследованиях и разработках с прикладным эффектом, например, автоматизация бизнес-процесса для повышения его эффективности; внедрение нового бизнес-процесса; построение интегрированной информационной системы предприятия.

Материалами для анализа должны быть публикации в отечественных и зарубежных рецензируемых журналах, входящих в международные индексы цитирования или в перечень ВАК. Также рекомендуется использовать в анализе авторефераты диссертаций и рецензируемые монографии и учебники. В анализ могут быть дополнительно включены публикации, не отвечающие перечисленным выше требованиям, но, по мнению автора анализа, содержащие значимые результаты, в том числе электронные ресурсы.

2.2. Первый этап НИР: Анализ публикаций и формирование терминологического базиса

На первом этапе НИР студент должен провести анализ научных публикаций по выбранной тематике магистерской диссертации. Основные задачи этапа:

1. Подтверждение актуальности тематики. Студент должен обосновать важность выбранной темы, указав на ее практическую значимость и перспективность. Для этого необходимо проанализировать современные тенденции в области информационных технологий и программирования, а также выявить проблемы, которые могут быть решены в рамках исследования.
2. Формирование терминологического базиса. На основе анализа публикаций студент должен выделить ключевые термины, используемые в выбранной области. Важно избегать избыточного использования общеупотребительных терминов и сосредоточиться на специфических понятиях, которые будут использоваться в дальнейшем исследовании.
3. Выявление авторских коллективов и их результатов. Студент должен определить ведущие научные коллективы и отдельных авторов, занимающихся исследованиями в выбранной или смежной тематике. Необходимо оценить их основные результаты, методы и средства, ис-

пользуемые для решения задач.

4. Конкретизация задач магистерской диссертации. На основе проведенного анализа студент должен уточнить объект исследования, определить ключевые проблемы и сформулировать задачи, которые будут решаться в рамках магистерской диссертации.

2.1.1. Подтверждение актуальности тематики

Подтверждение актуальности тематики позволяет обосновать необходимость и значимость выбранного направления исследования, в теоретическом и прикладном аспекте. Иначе говоря, подтверждение актуальности дает ответы на два вопроса: почему необходимо заниматься этим проектом или этой областью и почему этим вопросом необходимо заниматься именно сейчас. Для этого студент должен не только продемонстрировать понимание современных тенденций в области информационных технологий и программирования, но и выявить конкретные проблемы, которые могут быть решены в рамках исследования. Ниже приведены методики, которые помогут студенту эффективно подобрать литературу, выбрать релевантные и достоверные источники, а также выявить и систематизировать проблемы.

Использование научных баз данных и репозиториев:

- **Отечественные ресурсы:** Для поиска публикаций на русском языке рекомендуется использовать портал eLibrary.ru, который предоставляет доступ к научным статьям, монографиям и диссертациям. Особое внимание следует уделить журналам, входящим в перечень ВАК.
- **Зарубежные ресурсы:** Для поиска международных публикаций рекомендуется использовать базы данных, такие как **Scopus**, **Web of Science**, **IEEE Xplore**, **ACM Digital Library**, **SpringerLink** и **Elsevier ScienceDirect**. Эти ресурсы содержат статьи из рецензируемых журналов, конференций и книг, что обеспечивает высокое качество источников.
- **Открытые репозитории:** Для поиска актуальных исследований можно использовать открытые репозитории, такие как **arXiv.org** и **ResearchGate**, где авторы публикуют препринты своих работ.

Ключевые слова и поисковые запросы:

- Для эффективного поиска литературы необходимо сформулировать ключевые слова, связанные с тематикой исследования. Например, если тема исследования связана с машинным обучением, ключевыми словами могут быть: "machine learning", "deep learning", "neural

networks", "data analysis".

- Использование операторов поиска (AND, OR, NOT) позволяет уточнить запросы и найти наиболее релевантные статьи. Например, запрос "machine learning AND cybersecurity" позволит найти статьи, посвященные применению машинного обучения в области кибербезопасности. Для эффективного использования операторов поиска следует провести предварительную высокоуровневую систематизацию проекта, определив ключевые области, технологии, понятия.

Анализ библиографических ссылок

После нахождения ключевых статей по теме исследования рекомендуется изучить их библиографические ссылки, списки источников (литературы). Это позволит выявить дополнительные источники, которые могут быть полезны для исследования, сформировав сеть (граф) исследовательских групп, занимающихся данной тематикой.

Выбор релевантных и достоверных источников:

- **Рецензируемые журналы:** Статьи, опубликованные в рецензируемых журналах, проходят строгую проверку качества, что делает их более достоверными. Особое внимание следует уделить журналам, входящим в международные индексы цитирования (Scopus, Web of Science) или перечень ВАК.
- **Конференции:** Материалы международных конференций также могут быть полезны, особенно если конференция имеет высокий рейтинг в своей области (например, IEEE, ACM). Конференции также важны для прикладных тематик в области ИТ, так как зачастую предмет исследования может не иметь глубокого академического значения (и не создавать публикационную активность в рецензируемых журналах), но является практически востребованным, продвигаемым в обществе, соответственно — обсуждаемым на профильных конференциях.
- **Монографии и учебники:** Книги, написанные признанными экспертами в области, являются надежными источниками информации. Критерий «признанного эксперта» во многом основан на публикационной активности автора, цитируемости статей, ширине сетевых коммуникаций.

Оценка актуальности публикации. При выборе источников важно учитывать дату публикации. Для исследований в области информационных технологий и программирования рекомендуется использовать публикации, выпущенные за последние 5–7 лет, так как эта область быстро развивается.

Анализ цитирований. Количество цитирований статьи может служить индикатором ее значимости. Высокоцитируемые статьи, как правило, являются ключевыми работами в своей области. Также следует уделить внимание тому, кем цитируются статьи, то есть — в какой области исследований анализируемые результаты представляют ценность. Соотнеся это с предметом собственного проекта, можно сделать вывод о применимости технологии к своей ситуации. Например, если статьи о технологии цитируются только в исследованиях в области биологии и не цитируются в работах по экономике и статистике — то, возможно, технология имеет существенные ограничения, которые определяют такую специфику применения. При выявлении подобной аномалии цитирования требуется провести дополнительное исследование — насколько ограничения технологии значимы для своего проекта.

Систематический обзор литературы. Систематический обзор предполагает анализ большого числа публикаций с целью выявления общих тенденций, закономерностей и пробелов в исследованиях. Для этого студент должен:

- Сформулировать критерии отбора статей (например, временной период, ключевые слова, рейтинг журналов).
- Провести анализ публикаций, выделяя основные направления исследований, используемые методы и технологии.
- Сгруппировать статьи по тематикам и выявить наиболее популярные и перспективные направления.

Анализ ключевых показателей. Для выявления тенденций можно использовать количественные методы, такие как анализ частоты упоминания определенных технологий, методов или проблем в научных публикациях. Например, можно отследить, как часто упоминаются термины "blockchain", "quantum computing" или "artificial intelligence" в статьях за последние годы.

Использование инструментов визуализации. Для анализа тенденций можно использовать инструменты визуализации данных, такие как **VOSviewer** или **CiteSpace**. Эти инструменты позволяют построить карты научных публикаций, на которых видно, какие темы и технологии наиболее популярны в текущий момент.

Методы выявления и систематизации проблем

Анализ проблемных областей. Для выявления проблем, которые могут быть решены в рамках исследования, студент должен:

- Изучить публикации, посвященные критическому анализу существующих

ющих технологий и методов. В таких статьях часто указываются недостатки и ограничения текущих подходов.

- Обратить внимание на разделы «Дальнейшие исследования», «Ограничения и допущения», «Future work» или «Limitations» в научных статьях, где авторы указывают направления для дальнейших исследований. Если взять за основу информацию из этих разделов и на ее основе построить собственное исследование, это и будет обоснованием актуальности собственной работы и объяснением ее научной или прикладной ценности.

Опрос экспертов. Проведение интервью или анкетирования экспертов в выбранной области может помочь выявить актуальные проблемы и задачи. Эксперты могут указать на пробелы в исследованиях или предложить новые направления для изучения. Для опроса экспертов следует провести поиск среди специалистов Университета, занимающихся сходной тематикой (например, проведя поиск среди сотрудников релевантных подразделений на портале Университета, проанализировать тематику научных работ и графы совместных публикаций).

Анализ кейсов из практики. Изучение реальных кейсов внедрения технологий в промышленности или бизнесе позволяет выявить практические проблемы, с которыми сталкиваются разработчики и пользователи. Например, анализ кейсов внедрения систем машинного обучения может выявить проблемы, связанные с качеством данных или интерпретируемостью моделей. Для этого при поиске компании для прохождения практической подготовки можно опросить экспертов, реально работающих в интересующей области; ознакомиться с материалами профессиональных (практических) конференций, обсуждающих актуальные для отрасли проблемы внедрения технологий или бизнес-процессов на менее академическом, но значительно более прикладном уровне.

SWOT-анализ. SWOT-анализ (Strengths, Weaknesses, Opportunities, Threats) используется для систематизации проблем и возможностей в выбранной области исследования (технологии, применения, бизнес-процесса). Анализ основан на систематизации характерных свойств технологии по четырем категориям (позитивные и негативные внутренние факторы и позитивные и негативные факторы окружения):

- **S (Strengths)** — сильные стороны (внутренние факторы);
- **W (Weaknesses)** — слабые стороны (внутренние факторы);
- **O (Opportunities)** — возможности (внешние факторы);
- **T (Threats)** — угрозы (внешние факторы).

Каждый фактор должен быть четко сформулирован и конкретен. Например, вместо "хорошая команда" лучше написать "высококвалифицированные разработчики с опытом работы более 5 лет". Анализ должен быть основан на фактах, а не на предположениях. Используйте данные, статистику, отзывы, исследования для подтверждения делаемых предположений или предлагаемых выводов.

Соблюдайте правило релевантности, учитывайте только те факторы, которые действительно влияют на объект анализа. Не нужно перечислять всё подряд. Анализ должен быть направлен на достижение конкретных целей. Например, если цель — выход на новый рынок, то факторы должны быть связаны с этой задачей и рассматриваться через её призму.

Пример систематизации проблем

Предположим, тема исследования связана с применением искусственного интеллекта в медицине. В результате анализа литературы и экспертных мнений могут быть выявлены следующие проблемы:

1. **Недостаток качественных данных:** Медицинские данные часто неполны, зашумлены или содержат ошибки, что затрудняет обучение моделей.
2. **Интерпретируемость моделей:** Врачи не доверяют моделям, которые не могут объяснить свои решения.
3. **Этические и правовые вопросы:** Использование ИИ в медицине глубоко затрагивает вопросы конфиденциальности данных и ответственности за ошибки.

Эти проблемы могут стать основой для формулировки задач исследования и выбора методов их решения.

Таким образом, подтверждение актуальности тематики требует комплексного подхода, включающего анализ литературы, выявление тенденций и систематизацию проблем. Использование описанных методик позволит студенту обосновать важность выбранной темы и сформулировать задачи, которые будут решены в рамках исследования, а также выводы, подтверждающие актуальность и практическую значимость тематики исследования.

2.1.2. Формирование терминологического базиса

Формирование терминологического базиса — это важный этап научно-исследовательской работы, который позволяет студенту структурировать знания в выбранной области и обеспечить четкость изложения материала. Терминологический базис включает ключевые понятия, которые будут

использоваться в исследовании, и их определения. Важно, чтобы термины были релевантными, точными и соответствовали современным стандартам в выбранной области. Ниже приведены методики выделения релевантных терминов, а также рекомендации по работе с иноязычными терминами и введению собственных терминов.

Методики выделения релевантных терминов

Анализ ключевых публикаций. Для выделения терминов проанализируйте ключевые публикации по выбранной тематике. Особое внимание следует уделить статьям, монографиям и учебникам, которые являются основополагающими в данной области — их можно определить по индексам цитирования, а также по ссылкам в учебно-методических материалах по курсам нужной тематики. Также могут оказаться полезными подборки библиографических ссылок «Классика ...», «Лучшие книги по ...», «Обязательные к ознакомлению книги по ...» и т. п., однако следует проверить переиздание выбранных книг и ориентироваться на те из них, которые переиздавались в течение последних 5-7 лет.

В текстах публикаций необходимо выделить термины, которые часто повторяются и используются для описания ключевых концепций, методов и технологий. Например, в области машинного обучения такими терминами могут быть "нейронная сеть", "обучение с учителем", "кластеризация". Следует ознакомиться с терминологическими указателями или глоссариями, как указано далее.

Использование глоссариев и словарей. Многие научные статьи, книги и стандарты содержат глоссарии, в которых приведены определения ключевых терминов. Использование таких глоссариев позволяет уточнить значение терминов и избежать их неправильного толкования. Для поиска определений терминов можно использовать специализированные словари, такие как **IEEE Standard Glossary of Software Engineering Terminology** или **Oxford Dictionary of Computer Science**.

Анализ контекста использования терминов. Термины могут иметь разные значения в зависимости от контекста. Например, термин "облако" в области информационных технологий относится к "облачным вычислениям", а не к метеорологическому явлению. Поэтому важно анализировать контекст, в котором используется термин, чтобы понять его точное значение.

Использование инструментов анализа текста. Для автоматизации процесса выделения терминов можно использовать инструменты анализа текста, такие как **AntConc** или **Voyant Tools**. Эти инструменты позволяют анализировать частоту употребления слов и выделять ключевые термины.

Работа с иноязычными терминами

В области информационных технологий и программирования многие термины заимствованы из английского языка, и в русском языке может отсутствовать их общеупотребительный аналог. В таких случаях студент должен следовать следующим рекомендациям.

Использование транслитерации. Если термин широко используется в профессиональной среде и не имеет устоявшегося перевода, его можно оставить в оригинальном написании. Например, термины "blockchain", "framework", "backend" часто используются в русскоязычных текстах без перевода. При первом упоминании термина рекомендуется дать его определение или пояснение. Например: "Блокчейн (англ. blockchain) — это распределенная база данных, которая используется для хранения информации в виде цепочки блоков."

Использование устоявшихся переводов. Если термин имеет устоявшийся перевод, рекомендуется использовать его. Например, термин "machine learning" переводится как "машинное обучение", а "data mining" — как "анализ данных". Для поиска устоявшихся переводов можно использовать специализированные словари или глоссарии, такие как **Microsoft Language Portal** или **Глоссарий терминов по искусственному интеллекту**.

Создание собственного перевода. Если термин не имеет устоявшегося перевода, студент может предложить собственный вариант перевода. В этом случае необходимо обосновать выбор перевода и привести его определение. Например: "Термин 'feature engineering' можно перевести как 'инженерия признаков', подразумевая процесс создания новых признаков для улучшения качества модели машинного обучения."

Порядок действий при введении собственного термина

В некоторых случаях студент может столкнуться с необходимостью ввести собственный термин, особенно если исследование затрагивает новые или малоизученные области. В таком случае рекомендуется следовать следующему порядку действий:

1. **Обоснование необходимости введения термина:** студент должен объяснить, почему существующие термины не подходят для описания нового понятия. Например, если в исследовании разрабатывается новый метод обработки данных, который не имеет аналогов, может потребоваться введение нового термина.
2. **Формулировка определения:** Новый термин должен быть четко определен. Определение должно быть лаконичным и точным, чтобы

избежать двусмысленности. Например: "Термин 'когнитивная кластеризация' обозначает метод группировки данных, основанный на моделировании когнитивных процессов человека."

3. **Использование термина в контексте:** После введения термина студент должен использовать его в тексте исследования, чтобы закрепить его значение. Например: "В данной работе предложен метод когнитивной кластеризации, который позволяет улучшить точность группировки данных на 15% по сравнению с традиционными методами."
4. **Согласование с научным руководителем:** Перед введением нового термина рекомендуется обсудить его с научным руководителем, чтобы убедиться в его корректности и уместности.
5. **Добавление термина в глоссарий:** Новый термин и его определение должны быть добавлены в глоссарий, который обычно размещается в конце работы. Это поможет читателю быстро найти и понять значение термина.

Пример формирования терминологического базиса

Предположим, тема исследования связана с разработкой новых методов обработки больших данных. В результате анализа литературы студент выделяет следующие ключевые термины:

1. **Большие данные (Big Data)** — объемы данных, которые слишком велики для обработки традиционными методами.
2. **Машинное обучение (Machine Learning)** — метод анализа данных, который позволяет компьютерам обучаться на основе опыта.
3. **Нейронная сеть (Neural Network)** — математическая модель, имитирующая работу человеческого мозга.
4. **Когнитивная кластеризация (Cognitive Clustering)** — предложенный автором метод группировки данных, основанный на моделировании когнитивных процессов человека.

Каждый термин сопровождается определением и примерами использования в контексте исследования.

Заключение

Таким образом, формирование терминологического базиса требует внимательного анализа литературы, работы с иноязычными терминами и, при необходимости, введения новых понятий. Использование описанных методик позволит студенту создать четкий и структурированный термино-

логический аппарат, который будет использоваться в дальнейшем исследовании.

2.1.3. Выявление авторских коллективов и их результатов

Выявление ведущих научных коллективов и отдельных авторов, занимающихся исследованиями в выбранной или смежной тематике, является важным этапом научно-исследовательской работы. Это позволяет студенту понять, какие направления исследований являются наиболее перспективными, какие методы и технологии используются в данной области, а также выявить пробелы в существующих исследованиях. Ниже приведено практическое руководство по определению ведущих авторских коллективов и оценке их результатов.

Методики определения ведущих авторских коллективов

Анализ цитирований и индексов научного влияния:

- **Индекс Хирша (h-index):** Этот показатель отражает количество статей автора, которые были процитированы определенное количество раз. Авторы с высоким индексом Хирша считаются ведущими в своей области. Например, если у автора $h\text{-index} = 30$, это означает, что у него есть 30 статей, каждая из которых была процитирована не менее 30 раз.
- **Количество цитирований:** Авторы, чьи работы часто цитируются, как правило, являются признанными экспертами в своей области. Для анализа цитирований можно использовать базы данных, такие как **Google Scholar**, **Scopus** или **Web of Science**.
- **Индекс цитирования журналов (Impact Factor):** Журналы с высоким импакт-фактором публикуют статьи, которые часто цитируются. Авторы, публикующиеся в таких журналах, обычно являются ведущими исследователями. Можно ориентироваться на рейтинги, определяющие место журнала среди аналогичных по тематике на языке квартилей: журнал, относящийся к Q1 (первому квартилю), находится в числе 25% лучших по данному индексу журналов.

Анализ публикационной активности:

- **Количество публикаций:** Авторы, которые регулярно публикуют статьи в рецензируемых журналах и на конференциях, как правило, являются активными исследователями в своей области. Однако следует принимать во внимание рейтинги изданий или конференций, а также размер научной группы — большие научные группы издают боль-

ше статей, так как каждый из авторов углубляет исследование в своей области.

- **Качество публикаций:** Важно учитывать не только количество, но и качество публикаций. Статьи в журналах с высоким импакт-фактором и на престижных конференциях (например, IEEE, ACM) имеют большее значение.

Участие в научных сообществах и организациях:

- **Членство в научных обществах:** Авторы, которые являются членами престижных научных обществ (например, IEEE, ACM, AAAS), как правило, имеют высокий авторитет в своей области.
- **Руководство научными проектами:** Авторы, которые руководят крупными научными проектами или являются главными исследователями (Principal Investigators), обычно являются ведущими экспертами.

Анализ научных наград и премий:

- **Научные награды:** Авторы, получившие престижные научные награды (например, Turing Award, IEEE Medal of Honor), являются признанными лидерами в своей области.
- **Гранты и финансирование:** Авторы, которые получают значительное финансирование для своих исследований (например, от NSF, EU Horizon 2020), обычно являются ведущими исследователями.

Анализ сотрудничества и соавторства:

- **Сети соавторства:** Авторы, которые активно сотрудничают с другими ведущими исследователями, как правило, являются частью крупных научных коллективов. Для анализа сетей соавторства можно использовать инструменты, такие как **VOSviewer** или **Gephi**. В рамках портала нашего Университета также реализован инструмент анализа сетей соавторства среди сотрудников вуза.
- **Международное сотрудничество:** Авторы, которые сотрудничают с исследователями из разных стран, обычно имеют более широкий кругозор и доступ к новым идеям и методам.

Оценка основных результатов, методов и средств

Анализ ключевых публикаций. Для оценки результатов авторского коллектива необходимо проанализировать их ключевые публикации. Особое внимание следует уделить статьям, которые имеют высокий уровень цитирований и опубликованы в престижных журналах.

В публикациях следует выделить основные результаты, методы и тех-

нологии, которые использовались для решения задач. Например, если авторский коллектив занимается разработкой алгоритмов машинного обучения, важно понять, какие именно алгоритмы они разработали и какие задачи решали.

Использование инструментов анализа научной деятельности:

- **Google Scholar:** Позволяет найти профили авторов, их публикации и цитирования.
- **Scopus и Web of Science:** Предоставляют подробную информацию о публикациях, цитированиях и индексах научного влияния.
- **ResearchGate:** Социальная сеть для ученых, где можно найти профили авторов, их публикации и проекты.

Анализ патентов и коммерциализации. Если авторский коллектив занимается прикладными исследованиями, важно проанализировать их патенты и коммерческие продукты. Например, такие компании, как Google, IBM и Microsoft, активно патентуют свои разработки в области искусственного интеллекта и машинного обучения.

Наличие патентов и коммерческих продуктов свидетельствует о практической значимости исследований.

Участие в конференциях и семинарах. Авторы, которые регулярно выступают на престижных конференциях и семинарах, как правило, являются ведущими экспертами в своей области. Например, участие в конференциях, таких как NeurIPS, CVPR или ICML, свидетельствует о высоком уровне исследований.

Пример выявления ведущих авторских коллективов

Предположим, тема исследования связана с разработкой алгоритмов глубокого обучения для обработки изображений. В результате анализа студент может выделить следующие ведущие авторские коллективы:

Группа исследователей из Google Brain:

- **Основные результаты:** Разработка архитектуры Transformer, которая используется в моделях обработки естественного языка и изображений.
- **Методы и технологии:** Использование методов глубокого обучения, таких как сверточные нейронные сети (CNN) и рекуррентные нейронные сети (RNN).
- **Публикации:** Статьи в журналах с высоким импакт-фактором, такие как Nature и IEEE Transactions on Pattern Analysis and Machine

Intelligence.

Лаборатория компьютерных наук и искусственного интеллекта MIT (CSAIL):

- **Основные результаты:** Разработка алгоритмов для автоматического анализа медицинских изображений.
- **Методы и технологии:** Использование методов машинного обучения и компьютерного зрения.
- **Публикации:** Статьи на конференциях, таких как CVPR и ICCV.

Исследовательская группа DeepMind:

- **Основные результаты:** Разработка алгоритмов для игры в Go и StarCraft, а также для анализа медицинских данных.
- **Методы и технологии:** Использование методов глубокого обучения с подкреплением (Reinforcement Learning).
- **Публикации:** Статьи в журналах, таких как Nature и Science.

Таким образом, выявление ведущих авторских коллективов требует комплексного подхода, включающего анализ цитирований, публикационной активности, участия в научных сообществах и оценки результатов. Использование описанных методик позволит студенту определить ключевых исследователей в выбранной области и оценить их вклад в развитие науки и технологий.

2.1.4. Конкретизация задач квалификационной работы (ВКР, диссертации)

Конкретизация задач квалификационной работы — это ключевой этап научно-исследовательской работы, который позволяет студенту четко определить, что именно будет исследоваться, какие результаты ожидаются и как они будут достигнуты. Этот этап требует тщательного анализа литературы, формулировки исследовательских вопросов и разработки плана исследования. Ниже приведены четкие инструкции по постановке задач, формулировке ожидаемых результатов, разработке плана исследования и определению критериев достижения результата.

Постановка задачи исследования

Анализ существующих исследований. Прежде чем формулировать задачи, необходимо провести анализ существующих исследований в выбранной области. Это поможет выявить пробелы в знаниях и определить,

какие задачи еще не решены или решены недостаточно хорошо.

Например, если тема исследования связана с машинным обучением, можно проанализировать, какие алгоритмы уже разработаны, какие задачи они решают и какие проблемы остаются нерешенными.

Формулировка исследовательских вопросов. На основе анализа литературы студент должен сформулировать конкретные исследовательские вопросы. Эти вопросы должны быть четкими, измеримыми и достижимыми.

Пример исследовательского вопроса: "По каким параметрам и каким способом можно улучшить точность классификации изображений с использованием методов глубокого обучения?"

Определение объекта и предмета исследования

Объект исследования — это область или явление, которое изучается. Например, объектом исследования может быть "обработка изображений с использованием нейронных сетей".

Предмет исследования — это конкретный аспект объекта, который будет изучаться. Например, предметом исследования может быть "разработка нового алгоритма для классификации изображений".

Формулировка задач исследования

Задачи исследования должны быть конкретными и направленными на достижение цели исследования. Каждая задача должна быть сформулирована как действие, которое необходимо выполнить. Например:

- Провести анализ существующих методов классификации изображений.
- Разработать новый алгоритм для повышения точности классификации.
- Провести эксперименты для оценки эффективности предложенного алгоритма.

Формулировка ожидаемого результата

Определение ожидаемых результатов. Ожидаемые результаты должны быть конкретными и измеримыми. Например, если задача заключается в разработке нового алгоритма, ожидаемым результатом может быть "повышение точности классификации изображений на 5% по сравнению с существующими методами".

Результаты могут быть как теоретическими (например, новая модель

или алгоритм), так и практическими (например, программная реализация алгоритма).

Описание практической значимости. Необходимо указать, как результаты исследования могут быть применены на практике. Например, разработанный алгоритм классификации изображений может быть использован в медицинской диагностике для автоматического анализа рентгеновских снимков.

Разработка плана исследования

Определение этапов исследования. План исследования должен включать четкие этапы, которые необходимо выполнить для достижения цели. Каждый этап должен быть связан с конкретной задачей. Названия этапов должны быть конкретно связаны с темой работы, следует избегать типовых названий и слишком общих формулировок.

Примеры наименований этапов:

1. Анализ литературы по методам оптимизации производительности систем распознавания образов и постановка задачи поиска алгоритма предварительной фильтрации видеопотока (1 месяц).
2. Разработка алгоритма кодирования видеоизображения на основе вэйвлетов для ускорения обработки опорных кадров (2 месяца).
3. Проведение экспериментов по распознаванию изображений людей в промышленной среде и анализ фактического прироста скорости обработки. (2 месяца).
4. Оформление результатов исследования в виде научной статьи объемом до 12 страниц по требованиям издательства Springer (1 месяц).

Определение методов и инструментов. Для каждого этапа исследования необходимо указать методы и инструменты, которые будут использоваться. Например, для анализа литературы можно использовать методы систематического обзора, а для разработки алгоритма — языки программирования Python и библиотеки TensorFlow или PyTorch.

Составление графика выполнения работ. План исследования должен включать график выполнения работ с указанием сроков выполнения каждого этапа. Это поможет контролировать процесс исследования и своевременно вносить корректировки.

Формулировка критериев достижения результата

Определение критериев успеха. Критерии достижения результата

должны быть четко определены и измеримы. Например, если задача заключается в повышении точности классификации изображений, критерием успеха может быть "достижение точности классификации не менее 95% на тестовом наборе данных".

Оценка качества результатов. Необходимо определить, как будет оцениваться качество результатов. Например, для оценки эффективности алгоритма можно использовать метрики, такие как точность, полнота и F1-мера.

Определение рамок исследования. Рамки исследования определяют границы, в которых будет проводиться исследование. Например, если исследование связано с обработкой изображений, рамки могут быть ограничены определенным типом изображений (например, медицинские изображения) или определенным набором данных (например, ImageNet).

Пример конкретизации задач выпускной квалификационной работы

Тема исследования: "Разработка алгоритма для автоматической классификации медицинских изображений с использованием методов глубокого обучения".

Задачи исследования:

1. Провести анализ существующих методов классификации медицинских изображений.
2. Разработать новый алгоритм на основе сверточных нейронных сетей для повышения точности классификации.
3. Провести эксперименты для оценки эффективности предложенного алгоритма на наборе данных медицинских изображений, полученных от организации-партнера, объемом 43000 уникальных изображений.
4. Сравнить результаты предложенного алгоритма с существующими методами классификации, выявленными в процессе обзора.

Ожидаемые результаты:

1. Разработанный алгоритм, который повышает точность классификации медицинских изображений на 5% по сравнению с существующими методами.
2. Программная реализация алгоритма, которая может быть использована в медицинской диагностике.

План исследования:

1. Анализ литературы и постановка задачи (1 месяц).

2. Разработка алгоритма (2 месяца).
3. Проведение экспериментов и анализ результатов (2 месяца).
4. Оформление результатов исследования (1 месяц).

Критерии достижения результата:

1. Точность классификации не менее 95% на тестовом наборе данных.
2. Улучшение точности классификации на 5% по сравнению с существующими методами.

Рамки исследования:

1. Исследование ограничено обработкой медицинских изображений, таких как рентгеновские снимки и МРТ.
2. Используемый набор данных — публично доступный набор медицинских изображений (например, CheXpert).

Таким образом, конкретизация задач магистерской диссертации требует четкого формулирования исследовательских вопросов, определения ожидаемых результатов, разработки плана исследования и постановки критериев достижения результата. Использование описанных методик позволит студенту структурировать свою работу и достичь поставленных целей.

2.2. Второй этап НИР: Теоретическое исследование и моделирование

На втором этапе НИР студент переходит к теоретическому исследованию, которое включает моделирование объекта исследования, анализ существующих технологий и проектирование программных решений. Основные задачи этапа:

1. **Моделирование объекта исследования.** Студент должен разработать модели прикладных процессов, архитектуры информационной системы или других объектов, связанных с тематикой исследования. Модели должны быть адекватны поставленным задачам и обоснованы с точки зрения выбранных методов моделирования.
2. **Анализ существующих технологий.** На основе публикаций и других источников студент должен провести сравнительный анализ технологий, методов проектирования и разработки программного обеспечения. Результаты анализа должны быть использованы для обоснования

архитектуры предлагаемого решения.

3. **Проектирование программного компонента.** Студент должен разработать проектные решения на уровне архитектуры данных и программной архитектуры. Результаты проектирования должны быть представлены в виде элементов проектной документации.

2.2.1. Моделирование объекта исследования

Моделирование объекта исследования — это важный этап научно-исследовательской работы, который позволяет студенту формализовать и визуализировать процессы, системы или объекты, связанные с тематикой исследования. Моделирование помогает лучше понять структуру и поведение объекта, а также проверить гипотезы и предложить решения для поставленных задач. Однако для того чтобы модель была полезной, она должна быть адекватной поставленным задачам. Ниже приведено четкое руководство по алгоритму оценки адекватности модели.

Алгоритм оценки адекватности модели

Определение целей моделирования. Прежде чем приступить к созданию модели, необходимо четко определить, для чего она создается. Цели моделирования должны быть связаны с задачами исследования. Например, если задача заключается в оптимизации бизнес-процесса, целью моделирования может быть "создание модели, которая позволит выявить узкие места в процессе и предложить пути их устранения".

Выбор типа модели. В зависимости от целей исследования выбирается тип модели. Модели могут быть:

- **Математическими:** Используются для описания процессов с помощью уравнений и формул. Например, модели оптимизации или прогнозирования.
- **Графическими:** Используются для визуализации процессов или систем. Например, диаграммы UML, блок-схемы, сетевые графики.
- **Имитационными:** Используются для моделирования сложных систем, которые трудно описать аналитически. Например, моделирование работы производственной линии или транспортной системы.
- **Статистическими:** Используются для анализа данных и выявления закономерностей. Например, регрессионные модели или модели машинного обучения.

Определение параметров модели. Модель должна включать ключевые параметры, которые влияют на поведение объекта исследования.

Например, если модель описывает процесс обработки заказов в интернет-магазине, параметрами могут быть время обработки заказа, количество сотрудников, загруженность системы.

Сбор и анализ данных. Для создания модели необходимо собрать данные, которые будут использоваться для ее построения и валидации. Данные могут быть получены из различных источников: экспериментов, наблюдений, баз данных, публикаций.

Важно убедиться, что данные релевантны и достаточны для построения модели. Например, если модель описывает поведение пользователей на сайте, необходимо собрать данные о времени, проведенном на сайте, количестве посещений, частоте покупок.

Построение модели. На основе собранных данных и выбранного типа модели студент приступает к ее построению. Например, если модель математическая, необходимо разработать уравнения, описывающие процесс. Если модель графическая, необходимо создать диаграммы или схемы.

Валидация модели. Валидация модели — это процесс проверки ее адекватности реальному объекту или процессу. Для этого используются следующие методы:

- **Сравнение с реальными данными:** Модель должна воспроизводить реальные данные с достаточной точностью. Например, если модель прогнозирует продажи, ее прогнозы должны соответствовать фактическим данным.
- **Экспертная оценка:** Модель может быть проверена экспертами в данной области, которые оценят ее соответствие реальным процессам.
- **Тестирование на различных сценариях:** Модель должна быть устойчивой к изменениям входных данных и параметров. Например, если модель описывает работу системы, она должна корректно работать при изменении нагрузки.

Оценка адекватности модели. Адекватность модели означает возможность интерпретировать результаты, полученные в процессе моделирования, для практического использования в реальном мире. Адекватность оценивается по следующим критериям:

- **Точность:** Модель должна точно воспроизводить поведение объекта исследования. Например, если модель прогнозирует спрос на товары, ее прогнозы должны быть близки к реальным значениям.
- **Устойчивость:** Модель должна быть устойчивой к изменениям вход-

ных данных и параметров. Например, если модель описывает работу системы, она должна корректно работать при изменении нагрузки.

- **Простота:** Модель должна быть достаточно простой для понимания и использования. Сложные модели могут быть трудны для интерпретации и применения.
- **Релевантность:** Модель должна быть релевантна поставленным задачам. Например, если задача заключается в оптимизации процесса, модель должна позволять выявлять узкие места и предлагать пути их устранения.

Корректировка модели. Если модель не соответствует критериям адекватности, необходимо внести корректировки. Это может включать:

- Уточнение параметров модели.
- Изменение структуры модели.
- Использование других методов моделирования.

Документирование модели. В процессе разработки модели ее необходимо документировать, последовательно создавая необходимые разделы и внося в них корректировки по мере уточнения. В любой момент времени документация должна отражать текущие знания о модели, достаточные для ее представления или первоначального ознакомления. Документация в целом должна включать на финальном этапе:

- Описание целей модели.
- Описание параметров и структуры модели.
- Результаты валидации и оценки адекватности.
- Рекомендации по использованию модели.

Пример оценки адекватности модели

Тема исследования: "Оптимизация процесса обработки заказов в интернет-магазине".

Цель моделирования: Создать модель, которая позволит выявить узкие места в процессе обработки заказов и предложить пути их устранения.

Тип модели: Имитационная модель, построенная с использованием программного обеспечения AnyLogic.

Параметры модели:

- Время обработки заказа.

- Количество сотрудников.
- Загруженность системы.

Сбор данных: Данные о времени обработки заказов, количестве сотрудников и загруженности системы за последний год.

Построение модели: На основе собранных данных создана имитационная модель, которая описывает процесс обработки заказов.

Валидация модели:

- Сравнение с реальными данными: Прогнозы модели соответствуют фактическим данным с точностью 95%.
- Экспертная оценка: Эксперты подтвердили, что модель адекватно описывает процесс обработки заказов.
- Тестирование на различных сценариях: Модель корректно работает при изменении количества сотрудников и загруженности системы.

Оценка адекватности модели:

- Точность: Модель точно воспроизводит процесс обработки заказов.
- Устойчивость: Модель устойчива к изменениям входных данных.
- Простота: Модель достаточно проста для понимания и использования.
- Релевантность: Модель позволяет выявить узкие места и предложить пути их устранения.

Корректировка модели: Внесены незначительные изменения в параметры модели для повышения точности.

Документирование модели: Модель документирована, включая описание целей, параметров, результатов валидации и рекомендации по использованию.

Таким образом, оценка адекватности модели требует четкого алгоритма, включающего определение целей, выбор типа модели, сбор данных, валидацию и оценку. Использование описанных методик позволит студенту создать модель, которая будет адекватна поставленным задачам и полезна для дальнейшего исследования.

2.2.2. Анализ существующих технологий

Анализ существующих технологий — это важный этап научно-иссле-

довательской работы, который позволяет студенту выбрать наиболее подходящие методы, инструменты и технологии для решения поставленных задач. Этот этап включает сравнительный анализ существующих решений, оценку их преимуществ и недостатков, а также обоснование выбора архитектуры предлагаемого решения. Ниже приведен подробный алгоритм и подходы для сравнения технологий и выбора оптимальной архитектуры.

Алгоритм анализа и выбора технологий

Определение критериев выбора технологий. Прежде чем приступить к анализу, необходимо определить критерии, по которым будут сравниваться технологии. Эти критерии должны быть связаны с целями исследования и требованиями к системе. Примеры критериев:

- **Производительность:** Способность технологии обрабатывать большие объемы данных или выполнять сложные вычисления.
- **Масштабируемость:** Возможность масштабирования системы при увеличении нагрузки.
- **Надежность:** Устойчивость технологии к сбоям и ошибкам.
- **Совместимость:** Возможность интеграции с другими системами и технологиями.
- **Стоимость:** Затраты на внедрение и поддержку технологии.
- **Поддержка сообщества:** Наличие документации, форумов и активного сообщества разработчиков.

Сбор информации о технологиях. Для анализа необходимо собрать информацию о существующих технологиях, которые могут быть использованы для решения поставленных задач. Источники информации:

- **Научные публикации:** Статьи в рецензируемых журналах и материалы конференций.
- **Техническая документация:** Руководства, white papers и case studies от разработчиков технологий.
- **Опыт других проектов:** Анализ успешных кейсов внедрения технологий в аналогичных проектах.
- **Отзывы и рейтинги:** Оценки технологий на специализированных платформах, таких как **Gartner**, **Stack Overflow**, **GitHub**.

Сравнительный анализ технологий. На основе собранной информации проводится сравнительный анализ технологий по выбранным критериям. Для этого можно использовать следующие методы:

- **Таблицы сравнения:** Создание таблицы, в которой технологии сравниваются по каждому критерию. Если критерий не предполагает численного значения параметра, необходимо предусмотреть распределение характеристик по нескольким «корзинам», каждой из которых сопоставлена условная числовая оценка (1-2-3..., 1-2-4-8..., 1-10-100...). При назначении числовой оценки настоятельно рекомендуется придерживаться единого принципа «лучше-хуже» при увеличении числа, не смешивая рейтинговый (1 — наилучший, 2 — хуже...) и количественный (1 — меньше, 2 — больше, 10 — еще больше...) принципы. Например:

Технология	Скорость	Масштаб	Надежность	Стоимость	Поддержка сообщества
Технология А	100	2	5	1	3
Технология В	10	3	3	10	2
Технология С	1	1	1	100	1

- **Метод взвешенных оценок:** Каждому критерию присваивается вес в зависимости от его важности для исследования, который является весовым коэффициентом для числовой оценки. Затем технологии оцениваются по каждому критерию, и вычисляется общий балл с учетом весовых коэффициентов. Например:

Критерий	Вес	Технология А	Технология В	Технология С
Производительность	30%	5	3	1
Масштабируемость	25%	3	5	2
Надежность	20%	5	3	2
Стоимость	15%	4	3	1
Поддержка сообщества	10%	5	3	2
Итого	100%	4.4	3.5	1.7

- **SWOT-анализ:** Анализ сильных и слабых сторон технологий, а также возможностей и угроз, связанных с их использованием. Например:

Технология	Сильные стороны	Слабые стороны	Возможности	Угрозы
Технология А	Высокая производительность	Ограниченная масштабируемость	Интеграция с новыми системами	Конкуренция со стороны более дешевых решений
Технология В	Хорошая масштабируемость	Средняя надежность	Расширение функциональности	Зависимость от сторонних библиотек
Технология С	Низкая стоимость	Низкая	Использование в	Ограниченная

С

производительность

небольших
проектах

поддержка
сообщества

Выбор оптимальной технологии. На основе результатов сравнительного анализа выбирается технология, которая наилучшим образом соответствует критериям и целям исследования. Например, если основным критерием является производительность, выбирается технология с наивысшей оценкой по этому критерию.

Обоснование выбора архитектуры. После выбора технологий необходимо обосновать архитектуру предлагаемого решения. Архитектура должна быть выбрана в зависимости от целей исследования и требований к системе. Примеры архитектурных подходов:

- **Монолитная архитектура:** Подходит для небольших проектов с простой структурой. Преимущества: простота разработки и развертывания. Недостатки: сложность масштабирования и поддержки.
- **Микросервисная архитектура:** Подходит для крупных проектов с высокой нагрузкой. Преимущества: масштабируемость, гибкость, независимость сервисов. Недостатки: сложность управления и интеграции.
- **Сервис-ориентированная архитектура (SOA):** Подходит для проектов, требующих интеграции с другими системами. Преимущества: возможность повторного использования сервисов, гибкость. Недостатки: сложность управления и высокая стоимость.

Документирование выбора технологий и архитектуры. Результаты анализа и выбора технологий должны быть документированы. Документация должна включать:

- Описание выбранных технологий и их преимуществ.
- Обоснование выбора архитектуры.
- План внедрения технологий и архитектуры в проект.

Методологии рейтингования и выбора технологий

1. Gartner Magic Quadrant:

Методология Gartner Magic Quadrant используется для оценки технологий и поставщиков на основе их способности к выполнению задач и полноты видения. Технологии оцениваются по

двум осям: "Ability to Execute" (способность к выполнению) и "Completeness of Vision" (полнота видения). Результаты представляются в виде квадрантов: лидеры, претенденты, визионеры и нишевые игроки.

<https://www.gartner.com/en/research/methodologies/magic-quadrants-research>

2. Technology Readiness Level (TRL):

Методология TRL используется для оценки уровня готовности технологии к внедрению. Уровни готовности варьируются от TRL 1 (базовые исследования) до TRL 9 (успешное внедрение в производство).

<https://www.nasa.gov/directorates/somd/space-communications-navigation-program/technology-readiness-levels/>

3. SWOT-анализ:

SWOT-анализ (Strengths, Weaknesses, Opportunities, Threats) используется для оценки сильных и слабых сторон технологий, а также возможностей и угроз, связанных с их использованием.

<https://www.mindtools.com/ambj63/swot-analysis>

4. Weighted Scoring Model (WSM):

Метод взвешенных оценок используется для сравнения технологий по нескольким критериям с учетом их важности. Каждому критерию присваивается вес, и технологии оцениваются по каждому критерию. Общий балл вычисляется как сумма произведений оценок на веса.

<https://theproductmanager.com/topics/weighted-scoring-model/>

Пример выбора технологий и архитектуры

Тема исследования: "Разработка системы анализа больших данных для интернет-магазина".

Критерии выбора технологий:

- Производительность (метрика).
- Масштабируемость (метрика).
- Надежность (метрика).
- Стоимость (метрика).

- Поддержка сообщества (конкретика по сути и способам поддержки).

Сравнительный анализ технологий (в реальной ситуации заполняется конкретными метриками и фактами):

Технология	Скорость	Масштаб	Надежность	Стоимость	Поддержка сообществ	Итого
Apache Hadoop	Высокая	Высокая	Высокая	Средняя	Высокая	4.5
Apache Spark	Очень высокая	Высокая	Высокая	Высокая	Высокая	4.7
Google BigQuery	Высокая	Очень высокая	Высокая	Высокая	Средняя	4.3

Выбор технологии: Apache Spark, как наиболее сбалансированное решение по всем критериям согласно весовой таблице.

Обоснование архитектуры: Выбрана микросервисная архитектура, так как она обеспечивает масштабируемость не менее (метрика) и надежность (метрика), необходимые для обработки объемов данных, требуемых в задаче (метрика объемов).

Таким образом, анализ существующих технологий и выбор оптимальной архитектуры требуют четкого алгоритма, включающего определение критериев, сбор информации, сравнительный анализ и обоснование выбора. Использование описанных методик и подходов позволит студенту выбрать наиболее подходящие технологии и архитектуру для решения поставленных задач.

2.2.3. Проектирование программного компонента

Проектирование программного компонента — это ключевой этап научно-исследовательской работы, который позволяет студенту определить структуру и поведение компонента, а также разработать документацию, необходимую для его реализации. Этот этап включает разработку архитектуры данных и программной архитектуры, а также создание проектной документации. Ниже приведен подробный процесс разработки проектных решений, описание архитектуры данных и программной архитектуры, а также инструкции по выбору типов диаграмм и способов представления проектной документации.

Определение требований к компоненту

Прежде чем приступить к проектированию, необходимо четко определить требования к компоненту. Эти требования могут включать функциональные (что должен делать компонент) и нефункциональные (как он должен это делать) аспекты. Например, если компонент предназначен для обработки данных, требования могут включать:

- **Функциональные:** Возможность загрузки данных, их обработки и сохранения результатов.
- **Нефункциональные:** Высокая производительность, масштабируемость, надежность.

Разработка архитектуры данных

Архитектура данных определяет, как данные будут храниться, обрабатываться и передаваться в системе. Основные шаги:

1. **Определение структуры данных:** Необходимо определить, какие данные будут использоваться, их типы и форматы. Например, если компонент обрабатывает заказы, структура данных может включать поля: ID заказа, дата, сумма, статус.
2. **Выбор способа хранения данных:** В зависимости от требований к данным (объем, скорость доступа, надежность) выбирается способ хранения: реляционные базы данных (например, MySQL, PostgreSQL), NoSQL (например, MongoDB, Cassandra) или файловые системы.
3. **Определение потоков данных:** Необходимо описать, как данные будут перемещаться между компонентами системы. Например, данные могут передаваться через API, очереди сообщений (например, Kafka, RabbitMQ) или файлы.

Разработанная архитектура данных включает следующие элементы:

1. **Модель данных.** Модель данных описывает структуру данных и их взаимосвязи. Для представления модели данных используются диаграммы:
 - **ER-диаграммы (Entity-Relationship):** Используются для описания сущностей и их связей в реляционных базах данных.
 - **Диаграммы классов (Class Diagrams):** Используются для описания структуры данных в объектно-ориентированных системах.
2. **Схемы хранения данных.** Схемы хранения данных описывают, как данные будут храниться в базе данных или файловой системе. Для

представления схем используются:

- **Схемы баз данных.** Описывают таблицы, индексы, связи между таблицами.
 - **Диаграммы NoSQL.** Описывают коллекции, документы, ключи и значения в NoSQL базах данных.
3. **Потоки данных.** Потоки данных описывают, как данные перемещаются между компонентами системы. Для представления потоков данных используются:
- **Диаграммы потоков данных (Data Flow Diagrams, DFD):** Описывают потоки данных между процессами, хранилищами данных и внешними сущностями.
 - **Диаграммы последовательности (Sequence Diagrams):** Описывают последовательность передачи данных между компонентами.

Разработка программной архитектуры

Программная архитектура определяет структуру и взаимодействие компонентов системы. Основные шаги:

1. **Выбор архитектурного стиля.** В зависимости от требований к системе выбирается архитектурный стиль: монолитная, микросервисная, сервис-ориентированная (SOA) или событийно-ориентированная архитектура.
2. **Определение компонентов системы.** Необходимо выделить основные компоненты системы и описать их функции. Например, в системе обработки заказов могут быть компоненты: загрузка данных, обработка заказов, хранение данных, отчетность.
3. **Описание взаимодействия компонентов.** Необходимо описать, как компоненты будут взаимодействовать друг с другом. Например, компонент загрузки данных может передавать данные в компонент обработки через API.
4. **Создание проектной документации.** Проектная документация должна включать описание архитектуры данных, программной архитектуры и взаимодействия компонентов. Для визуализации архитектуры используются диаграммы, которые помогают лучше понять структуру и поведение системы.

Разработанная программная архитектура включает следующие элементы:

1. **Компоненты системы.** Компоненты системы описывают основные

модули или сервисы, из которых состоит система. Для представления компонентов используются:

- **Диаграммы компонентов (Component Diagrams):** Описывают компоненты системы и их взаимодействие.
 - **Диаграммы развертывания (Deployment Diagrams):** Описывают, как компоненты будут развернуты на физических или виртуальных серверах.
2. **Взаимодействие компонентов.** Взаимодействие компонентов описывает, как компоненты обмениваются данными и вызывают друг друга. Для представления взаимодействия используются:
- **Диаграммы последовательности (Sequence Diagrams):** Описывают последовательность вызовов методов и передачи данных между компонентами.
 - **Диаграммы взаимодействия (Communication Diagrams):** Описывают взаимодействие компонентов в виде сообщений.
3. **Архитектурные стили.** Архитектурные стили определяют общие принципы организации системы. Для представления архитектурных стилей используются:
- **Диаграммы слоев (Layered Diagrams):** Описывают систему как набор слоев, каждый из которых выполняет определенные функции.
 - **Диаграммы микросервисов (Microservices Diagrams):** Описывают систему как набор независимых сервисов, взаимодействующих через API.

Инструкция по выбору типов диаграмм и способов представления проектной документации

Типы диаграмм выбираются в зависимости от целей проектирования и аудиторки, для которой создается документация. Например:

- Для описания структуры данных используются ER-диаграммы или диаграммы классов.
- Для описания потоков данных используются диаграммы потоков данных (DFD) или диаграммы последовательности.
- Для описания компонентов системы используются диаграммы компонентов или диаграммы развертывания.

Способы представления проектной документации:

1. **Текстовый документ:** Описание архитектуры, требований и решений в виде текста с диаграммами. Подходит для детального описания архитектуры и решений. Используется для технической документации, которая будет использоваться разработчиками.
2. **Презентация:** Краткое описание архитектуры и ключевых решений в виде слайдов. Подходит для краткого представления архитектуры и ключевых решений. Используется для презентации проекта заинтересованным сторонам.
3. **Интерактивные диаграммы:** Использование инструментов, таких как **Lucidchart**, **Draw.io** или **Enterprise Architect**, для создания интерактивных диаграмм, которые можно изменять и комментировать. Подходят для совместной работы над проектом и обсуждения архитектурных решений. Используются для визуализации сложных систем и взаимодействия между компонентами.

Пример проектирования программного компонента

Тема исследования: "Разработка компонента для анализа данных в реальном времени".

Архитектура данных:

- **Модель данных:** Используется ER-диаграмма для описания сущностей (например, данные сенсоров, результаты анализа) и их связей.
- **Схема хранения данных:** Данные хранятся в NoSQL базе данных (MongoDB). Используется диаграмма коллекций и документов.
- **Потоки данных:** Данные передаются через очередь сообщений (Kafka). Используется диаграмма потоков данных (DFD).

Программная архитектура:

- **Компоненты системы:** Используется микросервисная архитектура. Основные компоненты: загрузка данных, обработка данных, хранение данных, визуализация результатов. Используется диаграмма компонентов.
- **Взаимодействие компонентов:** Компоненты взаимодействуют через API. Используется диаграмма последовательности для описания вызовов методов.
- **Архитектурный стиль:** Используется событийно-ориентированная архитектура. Используется диаграмма микросервисов.

Проектная документация:

- **Текстовый документ:** Детальное описание архитектуры данных и программной архитектуры.
- **Презентация:** Краткое описание ключевых решений и диаграмм.
- **Интерактивные диаграммы:** С помощью Lucidchart создаются интерактивные диаграммы, используемые в рабочем режиме на разных этапах проекта. Участники проекта вносят комментарии или изменения, финальные диаграммы служат основой для статической документации и для презентации.

Таким образом, проектирование программного компонента требует четкого процесса разработки, включающего определение требований, разработку архитектуры данных и программной архитектуры, а также создание проектной документации. Использование описанных методик и инструментов позволит студенту разработать качественные проектные решения и представить их в виде понятной и структурированной документации.

Отчет по второму этапу НИР должен содержать описание объекта исследования, постановку задачи, результаты теоретического исследования и выводы о возможности использования полученных результатов. Объем отчета не должен превышать 15 страниц.

2.3. Третий этап НИР: Обоснование методов решения задач магистерской диссертации

На третьем этапе НИР студент должен обосновать методы решения задач магистерской диссертации. Основные задачи этапа:

1. **Финальная формулировка темы и задач ВКР.** Студент должен уточнить тему магистерской диссертации и сформулировать задачи, которые будут решаться в рамках исследования.
2. **Обоснование методов решения задач.** Для каждой задачи студент должен провести анализ существующих методов, описать критерии выбора и обосновать выбор конкретного метода. В качестве обоснования могут быть использованы научные публикации, результаты прототипирования или сравнительного тестирования.
3. **Планирование решения задач.** Студент должен разработать план-график решения задач с детализацией по неделям.

Отчет по третьему этапу НИР должен содержать финальную формулировку темы и задач ВКР, обоснование методов решения задач и план-график их выполнения. Инструменты и артефакты, приводимые в отчете, совпадают по сути с описанными на предыдущем этапе, отличия лишь в практической направленности — на данном этапе требуется конкретная проработка технических решений (выбор сделан и обоснован), конкретика в архитектуре (архитектура выбрана, частично реализована), конкретный план проекта (даны оценки сложности задач, сделано планирование). Объем отчета не должен превышать 10 страниц.

3. Учебная практика

Учебная практика направлена на получение первичных навыков научно-исследовательской работы. В рамках учебной практики студент должен выбрать тематику исследования, сформировать терминологический базис и провести анализ публикаций по выбранной тематике, соответствующей некоторому учебному проекту (возможно, не имеющему практического воплощения). Все применяемые инструменты и подходы соответствуют описанным в предыдущей главе.

Основные задачи учебной практики:

1. **Выбор тематики исследования.** Студент должен выбрать тему, которая соответствует направлениям подготовки и представляет интерес для дальнейшего исследования.
2. **Анализ публикаций.** Студент должен провести анализ научных публикаций, выявить ключевые результаты и методы, используемые в выбранной области.
3. **Формирование терминологического базиса.** На основе анализа публикаций студент должен выделить ключевые термины, которые будут использоваться в дальнейшем исследовании.

Отчет по учебной практике должен содержать описание выбранной тематики, результаты анализа публикаций и выводы по проведенному исследованию. Отчет оформляется в соответствии с ГОСТ 7.32-2001, должен содержать ссылки на анализируемые публикации. Объем отчета по учебной практике составляет 10 страниц, включая основные диаграммы.

4. Проектная работа

Проектная работа направлена на получение практических навыков управления проектами и работы в команде. В рамках практики студент должен участвовать в постановке целей проекта, планировании задач и контроле их выполнения.

4.1. Организация и управление проектами

Основные задачи проектной работы:

1. **Постановка целей проекта.** Студент должен сформулировать цели проекта в виде конкретных метрик и результатов.
2. **Планирование задач.** Студент должен разработать план проекта, определить исполнителей задач и составить сетевой график.
3. **Контроль выполнения проекта.** Студент должен отслеживать выполнение задач, оценивать прогресс проекта и управлять рисками.

4.1.1. Постановка целей проекта

Постановка целей проекта — это фундаментальный этап, который определяет направление и успех всего исследования. Цели проекта должны быть четко сформулированы, измеримы и достижимы. Они служат основой для принятия архитектурных решений, выбора технологий и оценки результатов. В этом разделе мы подробно рассмотрим, как формулировать цели проекта, какие метрики использовать для их измерения, и как связать цели с архитектурными решениями и критериями приемки конечного результата.

Цели проекта — это конкретные, измеримые и достижимые результаты, которые должны быть достигнуты в ходе исследования. Цели должны быть сформулированы с учетом следующих критериев:

1. **Конкретность:** Цели должны быть четко определены и не допускать двусмысленности. Например, вместо общей цели "улучшить производительность системы" следует указать конкретную цель: "увеличить скорость обработки данных на 20%".
2. **Измеримость:** Цели должны быть выражены в количественных показателях (метриках), которые можно измерить и оценить. Например, "снизить время отклика системы до 100 мс".
3. **Достижимость:** Цели должны быть реалистичными и достижимыми с учетом имеющихся ресурсов и ограничений. Например, если проект

ограничен по времени и бюджету, цель "разработать полностью автономную систему искусственного интеллекта" может быть недостижимой.

4. **Релевантность:** Цели должны быть связаны с задачами исследования и иметь практическую значимость. Например, если проект связан с анализом данных, цель "разработать алгоритм для прогнозирования спроса" будет релевантной.
5. **Ограниченность во времени:** Цели должны быть привязаны к конкретным срокам. Например, "достичь снижения времени отклика системы до 100 мс в течение 3 месяцев".

Метрики — это количественные показатели, которые используются для измерения достижения целей. Они должны быть четко определены и измеримы. Примеры метрик:

- **Точность (Accuracy):** Процент правильных предсказаний модели. Например, точность классификации изображений.
- **Время отклика (Response Time):** Время, необходимое для выполнения запроса или обработки данных. Например, время обработки запроса в базе данных.
- **Пропускная способность (Throughput):** Количество операций, выполняемых системой за единицу времени. Например, количество запросов в секунду.
- **Надежность (Reliability):** Вероятность безотказной работы системы в течение определенного времени. Например, время наработки на отказ (MTBF).
- **Масштабируемость (Scalability):** Способность системы обрабатывать увеличивающуюся нагрузку. Например, количество пользователей, которые могут одновременно работать с системой.

Примеры целей проекта и метрик

Цель: Увеличить точность классификации изображений на 5% по сравнению с существующими методами.

- **Метрика:** Точность классификации (Accuracy).
- **Результат:** Разработанный алгоритм, который обеспечивает точность классификации 95%.

Цель: Снизить время обработки запросов в базе данных до 50 мс.

- **Метрика:** Время обработки запросов (Query Response Time).

- **Результат:** Оптимизированная база данных, которая обрабатывает запросы за 50 мс.

Цель: Увеличить пропускную способность системы до 1000 запросов в секунду.

- **Метрика:** Пропускная способность (Throughput).
- **Результат:** Система, которая обрабатывает 1000 запросов в секунду.

Связь целей проекта с архитектурными решениями

Цели проекта напрямую влияют на выбор архитектурных решений. Например:

1. **Цель:** Увеличить пропускную способность системы до 1000 запросов в секунду.

Архитектурное решение: Использование микросервисной архитектуры, которая позволяет масштабировать отдельные компоненты системы независимо друг от друга.

2. **Цель:** Снизить время обработки запросов в базе данных до 50 мс.

Архитектурное решение: Оптимизация структуры базы данных, использование индексов и кэширования.

3. **Цель:** Увеличить точность классификации изображений на 5%.

Архитектурное решение: Использование глубоких нейронных сетей (CNN) и методов аугментации данных для улучшения качества модели.

Критерии приемки конечного результата

Критерии приемки — это условия, которые должны быть выполнены для того, чтобы результат проекта считался успешным. Они должны быть четко определены и связаны с целями проекта. Примеры критериев приемки:

- **Точность классификации:** Разработанный алгоритм должен обеспечивать точность классификации не менее 95%.
- **Время обработки запросов:** Система должна обрабатывать запросы за время, не превышающее 50 мс.
- **Пропускная способность:** Система должна обрабатывать не менее 1000 запросов в секунду.
- **Надежность:** Система должна работать без сбоев в течение 99.9%

времени.

Пример постановки целей проекта

Тема исследования: "Разработка системы анализа больших данных для интернет-магазина".

Цели проекта:

1. Увеличить скорость обработки данных на 20%.
 - **Метрика:** Время обработки данных (Data Processing Time).
 - **Результат:** Система, которая обрабатывает данные за 80% от текущего времени.
 - **Архитектурное решение:** Использование распределенной обработки данных (Apache Spark).
 - **Критерий приемки:** Время обработки данных не должно превышать 80% от текущего значения.
2. Снизить время отклика системы до 100 мс.
 - **Метрика:** Время отклика системы (Response Time).
 - **Результат:** Система, которая обрабатывает запросы за 100 мс.
 - **Архитектурное решение:** Оптимизация базы данных и использование кэширования.
 - **Критерий приемки:** Время отклика системы не должно превышать 100 мс.
3. Увеличить точность прогнозирования спроса на 5%.
 - **Метрика:** Точность прогнозирования (Forecasting Accuracy).
 - **Результат:** Модель, которая обеспечивает точность прогнозирования 95%.
 - **Архитектурное решение:** Использование методов машинного обучения (например, LSTM).
 - **Критерий приемки:** Точность прогнозирования должна быть не менее 95%.

Таким образом, постановка целей проекта требует четкого формулирования, выбора метрик и определения критериев приемки. Цели проекта должны быть связаны с архитектурными решениями и конечными результа-

тами, что обеспечивает целостность и успешность исследования. Использование описанных методик позволит студенту разработать качественные проектные решения и достичь поставленных целей.

4.1.2. Планирование задач

Планирование задач — это ключевой этап управления проектом, который позволяет определить последовательность выполнения работ, распределить ресурсы и установить сроки. Этот этап включает разработку плана проекта, определение исполнителей задач, составление сетевого графика и оценку продолжительности и трудоемкости работ. Ниже приведены общие подходы к планированию, описание основных инструментов и методик, таких как диаграмма Гантта, сетевой график, критический путь, а также процедуры оценки продолжительности и трудоемкости.

Общие подходы к планированию

1. Прямое планирование:

- Прямое планирование предполагает последовательное определение задач, начиная с начальной точки проекта и заканчивая его завершением. Этот подход используется, когда цели и сроки проекта четко определены.
- Пример: Если проект начинается с анализа требований, то следующие задачи будут связаны с проектированием, разработкой, тестированием и внедрением.

2. Обратное планирование:

- Обратное планирование начинается с конечной цели проекта и движется назад к начальной точке. Этот подход полезен, когда сроки завершения проекта жестко фиксированы, и необходимо определить, какие задачи должны быть выполнены к определенным датам.
- Пример: Если проект должен быть завершен к 1 декабря, то обратное планирование поможет определить, когда должны быть завершены тестирование, разработка, проектирование и анализ требований.

Основные инструменты планирования

1. Диаграмма Гантта:

- **Определение:** Диаграмма Гантта — это инструмент визуализации плана проекта, который отображает задачи в виде горизонтальных полос на временной шкале. Каждая полоса представляет задачу, ее продолжительность и сроки выполнения.
- **Преимущества:** Диаграмма Гантта позволяет легко визуализировать последовательность задач, их продолжительность и зависимости между ними.
- **Пример:** На диаграмме Гантта можно отобразить задачи "Анализ требований", "Проектирование", "Разработка", "Тестирование" и "Внедрение" с указанием их продолжительности и сроков выполнения.

2. Сетевой график:

- **Определение:** Сетевой график — это графическое представление задач проекта и их взаимосвязей. Он состоит из узлов (задач) и ребер (зависимостей между задачами).
- **Преимущества:** Сетевой график позволяет определить критический путь и оптимизировать последовательность выполнения задач.
- **Пример:** На сетевом графике можно отобразить, что задача "Проектирование" зависит от задачи "Анализ требований", а задача "Разработка" зависит от задачи "Проектирование".

3. Критический путь:

- **Определение:** Критический путь — это последовательность задач, которая определяет минимальное время, необходимое для завершения проекта. Задачи на критическом пути не имеют резерва времени, и любая задержка в их выполнении приведет к задержке всего проекта.
- **Преимущества:** Определение критического пути позволяет сосредоточить внимание на наиболее важных задачах и управлять рисками.
- **Пример:** Если задачи "Анализ требований", "Проектирование", "Разработка" и "Тестирование" образуют критический путь, то задержка в любой из этих задач приведет к задержке проекта.

Процедуры оценки продолжительности и трудоемкости

- 1. Оценка продолжительности задач.** Продолжительность задачи — это время, необходимое для ее выполнения. Для оценки продолжительности можно использовать следующие методы:
 - **Экспертная оценка:** Оценка продолжительности на основе опыта экспертов.
 - **Аналогии:** Использование данных о продолжительности аналогичных задач в предыдущих проектах.
 - **Параметрическая оценка:** Использование математических моделей для оценки продолжительности на основе параметров задачи (например, объема работы, сложности).
- 2. Оценка трудоемкости задач.** Трудоемкость задачи — это количество человеко-часов, необходимых для ее выполнения. Для оценки трудоемкости можно использовать следующие методы:
 - **Разделение задачи на подзадачи:** Оценка трудоемкости каждой подзадачи и суммирование результатов.
 - **Использование нормативов:** Использование стандартных нормативов трудоемкости для определенных типов задач.
 - **Метод Дельфи:** Оценка трудоемкости группой экспертов с последующим усреднением результатов.

Пример планирования задач

Тема исследования: "Разработка системы анализа больших данных для интернет-магазина".

Задачи проекта:

1. Анализ требований.
2. Проектирование системы.
3. Разработка компонентов системы.
4. Тестирование системы.
5. Внедрение системы.

Диаграмма Ганта:

Задача	Начало	Окончание	Продолжительность
Анализ требований	01.10.2023	15.10.2023	2 недели
Проектирование системы	16.10.2023	05.11.2023	3 недели

Разработка компонентов	06.11.2023	26.11.2023	3 недели
Тестирование системы	27.11.2023	10.12.2023	2 недели
Внедрение системы	11.12.2023	15.12.2023	1 неделя

▪ **Оценка продолжительности и трудоемкости:**

Задача	Продолжительность	Трудоемкость (человеко-часы)
Анализ требований	2 недели	80
Проектирование системы	3 недели	120
Разработка компонентов	3 недели	180
Тестирование системы	2 недели	80
Внедрение системы	1 неделя	40

Таким образом, планирование задач требует использования различных инструментов и методик, таких как диаграмма Гантта, сетевой график и критический путь. Оценка продолжительности и трудоемкости задач позволяет более точно спланировать проект и распределить ресурсы. Использование описанных подходов поможет студенту эффективно управлять проектом и достичь поставленных целей.

4.1.3. Контроль выполнения проекта

Контроль выполнения проекта — это процесс отслеживания прогресса, оценки выполнения задач и управления рисками для обеспечения успешного завершения проекта в установленные сроки и в рамках бюджета. Этот этап включает регулярный мониторинг выполнения задач, анализ отклонений от плана и принятие корректирующих действий. Ниже приведены типовые практики управления проектом в ходе его выполнения, примеры процедур регулярного контроля и ревизии статуса.

Типовые практики управления проектом

1. Регулярный мониторинг прогресса:

- **Еженедельные встречи команды:** Проведение регулярных встреч для обсуждения текущего статуса задач, выявления проблем и планирования следующих шагов.
- **Отчеты о статусе:** Подготовка еженедельных или ежемесячных отчетов, которые включают информацию о выполнении задач,

отклонениях от плана и принятых мерах.

2. **Управление изменениями:**

- **Реестр изменений:** Ведение реестра изменений, в котором фиксируются все запросы на изменения, их статус и влияние на проект.
- **Процедура утверждения изменений:** Установление процедуры для рассмотрения и утверждения изменений, включая оценку их влияния на сроки, бюджет и качество.

3. **Управление рисками:**

- **Реестр рисков:** Ведение реестра рисков, в котором фиксируются идентифицированные риски, их вероятность, влияние и меры по их минимизации.
- **Регулярный пересмотр рисков:** Проведение регулярных встреч для пересмотра и обновления реестра рисков, а также для оценки эффективности принятых мер.

4. **Управление качеством:**

- **Проведение проверок качества:** Регулярное проведение проверок качества для обеспечения соответствия результатов проекта установленным стандартам.
- **Использование метрик качества:** Применение метрик качества для оценки выполнения задач и выявления отклонений.

Процедуры регулярного контроля и ревизии статуса

1. **Еженедельные статусные встречи:**

- **Цель:** Обсуждение текущего статуса задач, выявление проблем и планирование следующих шагов.
- **Участники:** Члены проектной команды, руководитель проекта, заинтересованные стороны.
- **Повестка дня:**
 1. Обзор выполнения задач за прошедшую неделю.
 2. Обсуждение проблем и рисков.
 3. Планирование задач на следующую неделю.
 4. Обновление реестра рисков и реестра изменений.

2. **Ежемесячные отчеты о статусе:**

- **Цель:** Предоставление обновленной информации о статусе проекта заинтересованным сторонам.
 - **Содержание отчета:**
 1. Обзор выполнения задач за прошедший месяц.
 2. Анализ отклонений от плана и принятых мер.
 3. Обновленная информация о рисках и изменениях.
 4. Планирование задач на следующий месяц.
3. **Проведение контрольных точек (Milestones):**
- **Цель:** Оценка прогресса проекта на ключевых этапах.
 - **Процедура:**
 1. Определение контрольных точек в плане проекта.
 2. Проведение встречи для оценки достижения контрольной точки.
 3. Принятие решений о продолжении проекта или внесении корректировок.
4. **Регулярный пересмотр реестра рисков:**
- **Цель:** Оценка текущих рисков и эффективности принятых мер.
 - **Процедура:**
 1. Проведение встречи для обсуждения текущих рисков.
 2. Обновление реестра рисков с учетом новых данных.
 3. Планирование мер по управлению выявленными новыми рисками.

Пример процедуры регулярного контроля

Тема исследования: "Разработка системы анализа больших данных для интернет-магазина".

Еженедельная статусная встреча:

1. **Обзор выполнения задач за прошедшую неделю:**
 - Задача "Анализ требований" выполнена на 100%.
 - Задача "Проектирование системы" выполнена на 80%.
 - Задача "Разработка компонентов" выполнена на 50%.

2. Обсуждение проблем и рисков:

- Проблема: Задержка в выполнении задачи "Проектирование системы" из-за отсутствия необходимых данных.
- Риск: Возможная задержка в выполнении задачи "Разработка компонентов" из-за сложности интеграции с существующей системой.

3. Планирование задач на следующую неделю:

- Завершение задачи "Проектирование системы".
- Начало задачи "Тестирование системы".
- Продолжение задачи "Разработка компонентов".

4. Обновление реестра рисков и реестра изменений:

- Добавлен новый риск: "Задержка в поставке оборудования для тестирования".
- Обновлен статус риска: "Сложность интеграции с существующей системой" — приняты меры по управлению, проведена декомпозиция задач, приняты архитектурные решения для снижения сложности.

Ежемесячный отчет о статусе:

1. Обзор выполнения задач за прошедший месяц:

- Задача "Анализ требований" выполнена на 100%.
- Задача "Проектирование системы" выполнена на 90%.
- Задача "Разработка компонентов" выполнена на 60%.

2. Анализ отклонений от плана и принятых мер:

- Задержка в выполнении задачи "Проектирование системы" на 1 неделю из-за отсутствия необходимых данных.
- Приняты меры по ускорению выполнения задачи "Разработка компонентов" — добавлены ресурсы на задачу.

3. Обновленная информация о рисках и изменениях:

- Риск: "Задержка в поставке оборудования для тестирования" — вероятность 30%, влияние — высокая.
- Изменение: Утверждено изменение в требованиях к системе.

4. Планирование задач на следующий месяц:

- Завершение задачи "Проектирование системы".
- Завершение задачи "Разработка компонентов".
- Начало задачи "Тестирование системы".

Таким образом, контроль выполнения проекта требует регулярного мониторинга прогресса, управления изменениями и рисками, а также проведения процедур регулярного контроля и ревизии статуса. Использование описанных практик и процедур поможет студенту эффективно управлять проектом и обеспечивать его успешное завершение.

4.2. Подготовка отчетов по проектной работе

Подготовка отчетов по проектной работе — это важный этап, который позволяет студенту структурировать и представить результаты своей работы, продемонстрировать компетентность в выбранной области и обосновать достигнутые результаты. Отчет должен быть кратким, но содержательным, чтобы читатель мог быстро понять, какие задачи были выполнены, какие методы использовались и какие результаты были достигнуты. Ниже приведены рекомендации по подготовке отчетов, а также пример структуры отчета.

Рекомендации по подготовке отчетов:

- **Краткость и ясность:**

Отчет должен быть написан четко и лаконично. Избегайте избыточного использования технических терминов и сложных формулировок.

Используйте структурированный подход: разделы, подзаголовки, списки и таблицы для улучшения читаемости.

- **Полнота информации:**

Отчет должен содержать всю необходимую информацию, чтобы читатель мог понять, какие задачи были выполнены, какие методы использовались и какие результаты были достигнуты.

Включите описание проблем, с которыми вы столкнулись, и как вы их решили.

- **Обоснование выбора методов и технологий:**

Объясните, почему вы выбрали определенные методы и технологии для решения задач. Это покажет вашу способность анализировать и выбирать оптимальные решения.

- **Демонстрация результатов:**

Представьте результаты в виде графиков, таблиц, диаграмм или других визуальных элементов. Это поможет читателю быстро понять, что было достигнуто.

Укажите, как результаты соответствуют поставленным целям и критериям приемки.

- **Рефлексия и выводы:**

Включите раздел с выводами, где вы обобщите свои достижения, укажите, что было сделано хорошо, а что можно улучшить в будущем.

Покажите, как ваш проект вносит вклад в выбранную область и какие перспективы он открывает.

Пример структуры отчета по проектной работе

1. **Введение:**

- Краткое описание проекта: цели, задачи, актуальность.
- Описание предметной области и проблем, которые решает проект.
- Пример: "Целью проекта является разработка системы анализа больших данных для интернет-магазина, которая позволит увеличить точность прогнозирования спроса на 5%."

2. **Описание выполненных задач:**

- Перечень задач, которые были выполнены в рамках проекта.
- Краткое описание каждой задачи, включая методы и технологии, которые использовались.
- Пример:
 1. "Задача 1: Анализ требований. Проведен анализ существующих систем и сформулированы требования к новой системе."
 2. "Задача 2: Проектирование системы. Разработана архитектура системы с использованием микросервисного подхода."

3. **Методы и технологии:**

- Описание методов и технологий, которые использовались для решения задач.
- Обоснование выбора методов и технологий.
- Пример: "Для обработки больших объемов данных был выбран Apache Spark, так как он обеспечивает производительность на уровне 1000 запросов в секунду на узел и масштабируемость до 10 узлов со средним коэффициентом масштабируемости 1.7."

4. **Результаты:**

- Описание достигнутых результатов, включая количественные показатели (метрики).
- Пример: "В результате разработки системы удалось увеличить точность прогнозирования спроса на 5%, а время обработки данных сократить на 20%."

5. **Визуализация результатов:**

- Графики, таблицы, диаграммы, которые наглядно демонстрируют достигнутые результаты.
- Пример: График, показывающий увеличение точности прогнозирования спроса после внедрения системы.

6. **Обсуждение:**

- Анализ результатов, их соответствие поставленным целям.
- Описание проблем, с которыми вы столкнулись, и как вы их решили.
- Пример: "Основной проблемой была интеграция системы с существующей инфраструктурой. Для ее решения был разработан адаптер, который обеспечил совместимость систем."

7. **Выводы:**

- Обобщение достигнутых результатов.
- Указание на вклад проекта в выбранную область.
- Пример: "Разработанная система позволяет улучшить точность прогнозирования спроса на 10% по сравнению с рассмотренными решениями и может быть использована в других интернет-магазинах."

8. **Рекомендации и перспективы:**

- Рекомендации по дальнейшему развитию проекта.
- Пример: "Для дальнейшего улучшения системы рекомендуется внедрить методы машинного обучения, что позволит повысить точность прогнозирования спроса еще на 20-25%."

Пример краткого отчета

1. Введение:

- Целью проекта является разработка системы анализа больших данных для интернет-магазина, которая позволит увеличить точность прогнозирования спроса на 5%.

2. Выполненные задачи:

- Анализ требований: Проведен анализ существующих систем и сформулированы требования к новой системе.
- Проектирование системы: Разработана архитектура системы с использованием микросервисного подхода.
- Разработка компонентов: Реализованы основные компоненты системы, включая модуль обработки данных и модуль прогнозирования спроса.
- Тестирование системы: Проведено тестирование системы на реальных данных, выявлены и устранены ошибки.

3. Методы и технологии:

- Для обработки больших объемов данных был выбран Apache Spark, так как он обеспечивает высокую производительность и масштабируемость. Для прогнозирования спроса использовались методы машинного обучения, такие как линейная регрессия и случайные леса.

4. Результаты:

- В результате разработки системы удалось увеличить точность прогнозирования спроса на 5%, а время обработки данных сократить на 20%.

5. Визуализация результатов:

- График, показывающий увеличение точности прогнозирования спроса после внедрения системы.

6. Обсуждение:

- Основной проблемой была интеграция системы с существующей инфраструктурой (декларация конкретных сложностей). Для ее решения был разработан адаптер, который обеспечил совместимость систем (архитектурная схема интеграции и обоснование решения декларированных сложностей в предложенном решении).

7. Выводы:

- Разработанная система позволяет улучшить точность прогнозирования спроса на 10% и может быть использована в других интернет-магазинах (подтверждение таблицами, графиками и статистическим анализом).

8. Рекомендации и перспективы:

- Для дальнейшего улучшения системы рекомендуется внедрить методы машинного обучения, что позволит повысить точность прогнозирования спроса еще на 25% (подтверждение с помощью обзора технологий и перспективных оценок).

Таким образом, подготовка отчетов по проектной работе требует четкой структуры, краткости и полноты информации. Использование описанных рекомендаций и примеров поможет студенту подготовить качественный отчет, который продемонстрирует его компетентность в выбранной области и достигнутые результаты.

Отчет по проектной работе должен содержать описание проекта, цели и результаты, а также описание конкретных задач, выполненных студентом. Отчет оформляется в соответствии с ГОСТ 7.32-2001 и должен содержать ссылки на использованные источники. Объем отчета не превышает 15 страниц.

5. Производственно-технологическая практика

Производственно-технологическая практика направлена на получение опыта работы в реальных условиях предприятия. В рамках практики студент должен участвовать в разработке и внедрении информационных систем.

5.1. Внедрение информационных систем

Основные задачи производственной практики:

1. **Разработка информационной системы.** Студент должен участвовать в разработке информационной системы, включая проектирование, тестирование и внедрение.
2. **Оценка результатов внедрения.** Студент должен оценить результаты внедрения информационной системы и сформулировать выводы по проведенной работе.

Разработка информационной системы в рамках реального предприятия

Разработка информационной системы (ИС) в рамках реального предприятия — это сложный и многогранный процесс, который требует не только технических знаний, но и умения взаимодействовать с заинтересованными сторонами, включая куратора со стороны компании. Этот процесс включает этапы проектирования, тестирования и внедрения системы, а также тесное взаимодействие с куратором для обеспечения успешной реализации проекта. Ниже приведены рекомендации по организации работы, роли куратора, правилам взаимодействия с ним и порядку предоставления отзыва.

Этапы разработки информационной системы

1. **Проектирование системы:**
 - **Анализ требований:** На этом этапе студент совместно с куратором и другими заинтересованными сторонами анализирует требования к системе. Это включает сбор информации о бизнес-процессах, определение функциональных и нефункциональных требований.
 - **Разработка архитектуры:** На основе анализа требований разрабатывается архитектура системы, включая выбор технологий, определение компонентов системы и их взаимодействия.
 - **Создание проектной документации:** Разрабатываются диаграммы (например, UML), схемы баз данных, спецификации

интерфейсов и другие документы, необходимые для реализации системы.

2. Разработка системы:

- **Реализация компонентов:** Студент участвует в разработке компонентов системы, включая написание кода, настройку баз данных, интеграцию с другими системами.
- **Соблюдение стандартов:** Разработка должна соответствовать стандартам компании, включая coding standards, guidelines по безопасности и качеству кода.

3. Тестирование системы:

- **Разработка тестовых сценариев:** Студент разрабатывает тестовые сценарии для проверки функциональности системы, включая unit-тесты, интеграционные тесты и тесты производительности.
- **Проведение тестирования:** Тестирование проводится на различных этапах разработки, включая модульное тестирование, системное тестирование и приемочное тестирование.
- **Устранение ошибок:** Выявленные в ходе тестирования ошибки фиксируются и устраняются.

4. Внедрение системы:

- **Подготовка к внедрению:** Студент участвует в подготовке к внедрению системы, включая миграцию данных, настройку окружения и обучение пользователей.
- **Постепенное внедрение:** Внедрение системы может происходить поэтапно, начиная с пилотного проекта и заканчивая полным развертыванием.
- **Мониторинг и поддержка:** После внедрения системы студент может участвовать в мониторинге ее работы и устранении возникающих проблем.

Роль куратора со стороны компании

Куратор со стороны компании играет ключевую роль в успешной реализации проекта. Его основные функции:

1. Координация и поддержка:

- Куратор помогает студенту понять бизнес-процессы компании и требования к системе.

- Он обеспечивает доступ к необходимым ресурсам, включая данные, документацию и техническую поддержку.
- 2. Контроль качества:**
 - Куратор контролирует выполнение задач и качество работы, предоставляет обратную связь и рекомендации.
 - Он участвует в приемочном тестировании и утверждении результатов.
 - 3. Связь с заинтересованными сторонами:**
 - Куратор выступает в качестве связующего звена между студентом и другими заинтересованными сторонами компании.
 - Он организует встречи и обсуждения, чтобы обеспечить согласованность действий.

Правила и порядок взаимодействия с куратором

- 1. Регулярные встречи:**
 - Проведение регулярных встреч с куратором для обсуждения текущего статуса проекта, выявления проблем и планирования следующих шагов.
 - Пример: Еженедельные статусные встречи, на которых студент представляет отчет о проделанной работе и обсуждает планы на следующую неделю.
- 2. Прозрачность и отчетность:**
 - Студент должен предоставлять куратору регулярные отчеты о ходе выполнения работ, включая описание выполненных задач, достигнутых результатов и возникших проблем.
 - Пример: Еженедельный отчет, который включает список выполненных задач, график выполнения и обновленный план работ.
- 3. Соблюдение сроков и обязательств:**
 - Студент должен соблюдать установленные сроки и выполнять обязательства перед компанией.
 - Пример: Если студент берет на себя обязательство завершить определенную задачу к определенной дате, он должен сделать все возможное, чтобы выполнить это обязательство.
- 4. Обратная связь и корректировка:**

- Студент должен быть открыт для обратной связи от куратора и готов вносить корректировки в свою работу на основе полученных рекомендаций.
- Пример: Если куратор указывает на недостатки в проектировании системы, студент должен внести необходимые изменения.

Порядок предоставления отзыва куратора от компании

1. Форма отзыва:

- Отзыв куратора может быть предоставлен в письменной форме, включая оценку работы студента, его вклад в проект и рекомендации для дальнейшего развития.
- Пример: Отзыв может включать оценку по следующим критериям: качество работы, соблюдение сроков, инициативность, взаимодействие с командой.

2. Содержание отзыва:

- Отзыв должен содержать конкретные примеры работы студента, его достижения и вклад в проект.
- Пример: "Студент успешно разработал модуль обработки данных, который позволил сократить время обработки на 20%. Он проявил инициативу в решении сложных технических задач и эффективно взаимодействовал с командой."

3. Использование отзыва:

- Отзыв куратора может быть использован для оценки работы студента в рамках учебного процесса, а также для его дальнейшего профессионального развития.
- Пример: Отзыв может быть включен в портфолио студента или использован для рекомендаций при трудоустройстве.

Пример взаимодействия с куратором

- **Тема исследования:** "Разработка системы анализа больших данных для интернет-магазина".
- **Регулярные встречи:** Еженедельные статусные встречи с куратором, на которых студент представляет отчет о проделанной работе и обсуждает планы на следующую неделю.
- **Отчетность:** Еженедельный отчет, который включает список выполненных задач, график выполнения и обновленный план работ.

- **Обратная связь:** Куратор предоставляет обратную связь по результатам работы, указывает на недостатки и дает рекомендации по улучшению.
- **Отзыв куратора:** "Студент успешно разработал модуль обработки данных, который позволил сократить время обработки на 20%. Он проявил инициативу в решении сложных технических задач, предложив оригинальное архитектурное решение и эффективно взаимодействовал с командой из 4 человек, включая системного архитектора и тестировщика."

Таким образом, разработка информационной системы в рамках реального предприятия требует тесного взаимодействия с куратором, соблюдения правил и порядка работы, а также предоставления регулярных отчетов и отзывов. Использование описанных рекомендаций поможет студенту успешно реализовать проект и продемонстрировать свои профессиональные навыки.

5.2. Подготовка отчетов по производственной практике

Отчет по производственной практике — это документ, который отражает опыт студента, полученный в ходе работы на предприятии. Он должен быть кратким, но содержательным, чтобы продемонстрировать навыки практической работы, усовершенствованные в процессе практики, и показать вклад студента в решение задач предприятия. Отчет должен включать описание предприятия, выполненной работы и результатов внедрения информационной системы. Ниже приведены рекомендации по подготовке отчета и пример его структуры.

Структура отчета по производственной практике

1. Введение:

- Краткое описание предприятия: название, сфера деятельности, основные продукты или услуги.
- Цели и задачи производственной практики.
- Пример: "Практика проходила в компании ООО «ТехноПрофиль», которая занимается разработкой программного обеспечения для автоматизации бизнес-процессов. Целью практики было участие в разработке и внедрении системы управления заказами."

2. Описание предприятия:

- Краткая характеристика предприятия: история, миссия, структура, ключевые направления деятельности.
- Описание отдела или подразделения, в котором проходила практика.
- Пример: "Компания ООО «ТехноПрофиль» была основана в 2010 году и специализируется на разработке CRM-систем. Практика проходила в отделе разработки, который состоит из 10 сотрудников, включая разработчиков, тестировщиков и аналитиков."

3. **Описание выполненной работы:**

- Перечень задач, которые были выполнены в ходе практики.
- Краткое описание каждой задачи, включая методы и технологии, которые использовались.
- Пример:
 1. "Задача 1: Анализ требований к системе управления заказами. Проведены интервью с сотрудниками отдела продаж, сформулированы функциональные и нефункциональные требования."
 2. "Задача 2: Разработка модуля обработки заказов. Использовались технологии Java и Spring Framework для реализации бизнес-логики."
 3. "Задача 3: Тестирование системы. Разработаны тестовые сценарии, проведено модульное и интеграционное тестирование."

4. **Результаты внедрения информационной системы:**

- Описание результатов внедрения системы, включая количественные показатели (метрики).
- Пример: "В результате внедрения системы управления заказами время обработки заказов сократилось на 30%, а количество ошибок при вводе данных уменьшилось на 50%."

5. **Визуализация результатов:**

- Графики, таблицы, диаграммы, которые наглядно демонстрируют достигнутые результаты.
- Пример: График, показывающий снижение времени обработки заказов после внедрения системы.

6. **Обсуждение:**

- Анализ результатов, их соответствие поставленным целям.

- Описание проблем, с которыми вы столкнулись, и как вы их решили.
- Пример: "Основной проблемой была интеграция системы с существующей CRM. Для ее решения был разработан адаптер, который обеспечил совместимость систем."

7. Выводы:

- Обобщение достигнутых результатов.
- Указание на вклад практики в профессиональное развитие студента.
- Пример: "Практика позволила мне усовершенствовать навыки разработки и тестирования программного обеспечения, а также получить опыт работы в команде."

8. Рекомендации и перспективы:

- Рекомендации по дальнейшему развитию системы или улучшению процессов на предприятии.
- Пример: "Для дальнейшего улучшения системы рекомендуется внедрить модуль аналитики, который позволит более точно прогнозировать спрос."

Пример краткого отчета

- **Введение:** Практика проходила в компании ООО «ТехноПрофиль», которая занимается разработкой программного обеспечения для автоматизации бизнес-процессов. Целью практики было участие в разработке и внедрении системы управления заказами.
- **Описание предприятия:** Компания ООО «ТехноПрофиль» была основана в 2010 году и специализируется на разработке CRM-систем. Практика проходила в отделе разработки, который состоит из 10 сотрудников, включая разработчиков, тестировщиков и аналитиков.
- **Описание выполненной работы:**
 - Анализ требований к системе управления заказами. Проведены интервью с сотрудниками отдела продаж, сформулированы функциональные и нефункциональные требования.
 - Разработка модуля обработки заказов. Использовались технологии Java и Spring Framework для реализации бизнес-логики.
 - Тестирование системы. Разработаны тестовые сценарии, проведе-

но модульное и интеграционное тестирование.

- **Результаты внедрения информационной системы:** В результате внедрения системы управления заказами время обработки заказов сократилось на 30%, а количество ошибок при вводе данных уменьшилось на 50%.
- **Визуализация результатов:** График, показывающий снижение времени обработки заказов после внедрения системы.
- **Обсуждение:** Основной проблемой была интеграция системы с существующей CRM. Для ее решения был разработан адаптер, который обеспечил совместимость систем.
- **Выводы:** Практика позволила мне усовершенствовать навыки разработки и тестирования программного обеспечения, а также получить опыт работы в команде.
- **Рекомендации и перспективы:** Для дальнейшего улучшения системы рекомендуется внедрить модуль аналитики, который позволит более точно прогнозировать спрос.

Таким образом, отчет по производственной практике должен быть кратким, но содержательным, чтобы отразить всю специфику работы и продемонстрировать навыки практической работы. Использование описанной структуры и рекомендаций поможет студенту подготовить качественный отчет, который покажет его вклад в решение задач предприятия и профессиональное развитие. Отчет оформляется в соответствии с ГОСТ 7.32-2001 и должен содержать ссылки на использованные источники.

6. Преддипломная практика

Преддипломная практика направлена на подготовку к защите магистерской диссертации. В рамках практики студент должен завершить исследование, оформить результаты и подготовиться к защите.

6.1. Подготовка к защите магистерской диссертации

Основные задачи преддипломной практики:

- **Завершение исследования.** Студент должен завершить все этапы исследования, оформить результаты и подготовить магистерскую диссертацию.
- **Подготовка к защите.** Студент должен подготовить презентацию и тезисы для защиты магистерской диссертации.

Завершение исследования — это финальный этап подготовки магистерской диссертации, на котором студент завершает все этапы научно-исследовательской работы, оформляет результаты и готовится к защите. В рамках преддипломной практики студент обеспечивает внедрение, тестирование и демонстрацию результатов разработки, выполненной в рамках выпускной квалификационной работы (ВКР). Этот этап требует тщательной подготовки, включая финальную доработку системы, проведение тестирования, внедрение в реальные условия и подготовку демонстрационных материалов. Ниже приведены рекомендации по завершению исследования и подготовке к защите.

Этапы завершения исследования

1. **Финальная доработка системы:**

- **Устранение выявленных недостатков:** На основе результатов тестирования и обратной связи от куратора студент вносит последние корректировки в систему.
- **Оптимизация производительности:** Проводится оптимизация кода, баз данных и других компонентов системы для повышения производительности и надежности.
- **Документирование:** Завершается разработка документации, включая руководство пользователя, техническую документацию и отчеты о тестировании.

2. **Тестирование системы:**

- **Функциональное тестирование:** Проверяется соответствие си-

стемы функциональным требованиям. Тестируются все сценарии использования системы.

- **Тестирование производительности:** Проводится нагрузочное тестирование для оценки производительности системы при различных уровнях нагрузки.
- **Приемочное тестирование:** Тестирование проводится совместно с заинтересованными сторонами (куратором, представителями предприятия) для подтверждения готовности системы к внедрению.

3. Внедрение системы:

- **Подготовка к внедрению:** Студент участвует в подготовке к внедрению системы, включая миграцию данных, настройку окружения и обучение пользователей.
- **Постепенное внедрение:** Внедрение системы может происходить поэтапно, начиная с пилотного проекта и заканчивая полным развертыванием.
- **Мониторинг и поддержка:** После внедрения системы студент может участвовать в мониторинге ее работы и устранении возникающих проблем.

4. Демонстрация результатов:

- **Подготовка демонстрационных материалов:** Студент готовит презентацию, демонстрационные видеоролики и другие материалы, которые помогут наглядно показать результаты работы.
- **Проведение демонстрации:** Демонстрация проводится для заинтересованных сторон, включая куратора, представителей предприятия и научного руководителя.

Подготовка магистерской диссертации

1. Оформление результатов исследования:

- **Структура диссертации:** Диссертация должна включать введение, теоретическую часть, описание методов и технологий, результаты исследования, выводы и рекомендации.
- **Описание внедрения и тестирования:** В диссертации должно быть подробно описано, как система была внедрена и протестирована, какие результаты были достигнуты.
- **Визуализация результатов:** Используйте графики, таблицы,

диаграммы и другие визуальные элементы для наглядного представления результатов.

2. Подготовка к защите:

- **Написание автореферата:** Автореферат — это краткое изложение диссертации, которое включает основные цели, задачи, методы и результаты исследования.
- **Подготовка презентации:** Презентация должна быть краткой и содержательной, включать ключевые моменты исследования и демонстрацию результатов.
- **Репетиция защиты:** Проведите репетицию защиты перед научным руководителем и коллегами, чтобы получить обратную связь и улучшить презентацию.

Пример завершения исследования

1. **Тема исследования:** "Разработка системы анализа больших данных для интернет-магазина".
2. **Финальная доработка системы:**
 - Устранены выявленные в ходе тестирования ошибки.
 - Проведена оптимизация производительности системы.
 - Завершена разработка документации, включая руководство пользователя и техническую документацию.
3. **Тестирование системы:**
 - Проведено функциональное тестирование, подтверждено соответствие системы всем требованиям.
 - Проведено нагрузочное тестирование, система выдерживает до 1000 запросов в секунду.
 - Проведено приемочное тестирование, система готова к внедрению.
4. **Внедрение системы:**
 - Подготовка к внедрению: Миграция данных, настройка окружения, обучение пользователей.
 - Постепенное внедрение: Начато с пилотного проекта в одном из отделов компании.
 - Мониторинг и поддержка: После внедрения система работает стабильно, выявленные проблемы оперативно устраняются.

5. Демонстрация результатов:

- Подготовлены презентация и демонстрационный видеоролик.
- Проведена демонстрация для заинтересованных сторон, включая куратора и научного руководителя.

6. Подготовка магистерской диссертации:

- Оформлены результаты исследования, включая описание внедрения и тестирования системы.
- Подготовлен автореферат и презентация для защиты.
- Проведена репетиция защиты, получена обратная связь и улучшена презентация.

Таким образом, завершение исследования требует тщательной подготовки, включая финальную доработку системы, тестирование, внедрение и демонстрацию результатов. Использование описанных рекомендаций поможет студенту успешно завершить исследование, оформить результаты и подготовиться к защите магистерской диссертации.

6.2. Оформление результатов преддипломной практики

Отчет по преддипломной практике должен содержать описание завершенного исследования, результаты и выводы по проведенной работе. Отчет оформляется в соответствии с ГОСТ 7.32-2001 и должен содержать ссылки на использованные источники. В качестве отчета по преддипломной практике во многих случаях может быть использована третья глава магистерской диссертации, содержащая сведения о завершении разработки и внедрении продукта.

7. Заключение

Настоящее учебно-методическое пособие предназначено для студентов магистратуры факультета информационных технологий и программирования. На основе данных рекомендаций предлагается организовать прохождение практической подготовки в едином цикле, с оптимизацией трудозатрат на проведение исследований, обработку результатов и подготовку отчетных материалов. Следование данному руководству позволит пройти полный цикл практической подготовки на основе реальных, полезных для промышленности и саморазвития проектов (задач), завершив цикл обучения в состоянии «нулевой готовности» к реальной работе на предприятии.

Зубок Дмитрий Александрович
Маятин Александр Владимирович
Станкевич Андрей Сергеевич
Ивановская Татьяна Юрьевна
Штумпф Святослав Алексеевич

Методические указания по выполнению научно-исследовательских работ, прохождению производственной практики и подготовке к государственной итоговой аттестации

Учебно-методическое пособие

В авторской редакции

Редакционно-издательский отдел Университета ИТМО

Зав. РИО

Н.Ф. Гусарова

Подписано к печати

Заказ №

Тираж

Отпечатано на ризографе

Редакционно-издательский отдел
Университета ИТМО
197101, Санкт-Петербург, Кронверкский пр., 49, литер А