

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1 ХАРАКТЕРИСТИКИ ПРЕДПРИЯТИЯ	6
1.1 История развития, общие сведения о предприятии	6
1.2 Организационная структура организации.....	7
2 АНАЛИЗ ФУНКЦИОНИРОВАНИЯ СТРУКТУРНЫХ ПОДРАЗДЕЛЕНИЙ.....	8
2.1 Анализ функций предприятия.....	8
2.2 Анализ функций, целей и задач IT- отдела	8
3 НОРМАТИВНАЯ ДОКУМЕНТАЦИЯ	10
3.1 Содержание должностных инструкций.....	10
3.2 Модели жизненного цикла программного обеспечения.....	13
4 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, ИСПОЛЬЗУЕМОЕ В ОРГАНИЗАЦИИ.....	16
4.1 Анализ степени автоматизации в организации	16
4.2 Анализ используемых программных продуктов	16
4.3 Среды разработки и языки программирования, используемые в организации.....	17
4.4 Операционные системы, используемые в организации.....	18
4.5 Требования, предъявляемые к разрабатываемому программному продукту	19
5 ИЗУЧЕНИЕ АРХИТЕКТУРЫ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ ПРЕДПРИЯТИЯ.....	20
6. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ	22
6.1 Постановка задачи	22
6.2 Реализация проекта	22
ЗАКЛЮЧЕНИЕ.....	25

ВВЕДЕНИЕ

Данный отчёт является результатом прохождения практики на предприятии Общество с Ограниченной Ответственностью «Фабрика инноваций и решений». Основной целью прохождения практики является приобретение профессиональных навыков по профилю специальности, получение опыта работы в команде, закрепление, расширение и систематизация знаний, полученных при изучении специальных дисциплин.

Restful Web Service — это легкий, обслуживаемый и масштабируемый сервис, построенный на архитектуре REST. Веб-служба Restful, предоставляя вызывающему клиенту API из вашего приложения безопасным, единообразным способом без сохранения состояния. Вызывающий клиент может выполнять predetermined операции, используя сервис Restful. На технологическую практику были поставлены следующие задачи:

- ознакомление с функциями и задачами отдела Разработки ООО «Фабрика инноваций и решений».
- изучение типов и характеристик используемых аппаратных средств и техники;
- изучение используемого программного обеспечения;
- написание Restful веб-сервиса на фреймворке «Spring Boot»;

					ЯНА.1710412.ПЗ	5
Изм.	Лист	№	Подп.	Дата		

1 ХАРАКТЕРИСТИКИ ПРЕДПРИЯТИЯ

1.1 История развития, общие сведения о предприятии

Innowise Group – молодая и перспективная компания, работающая в сфере IT-технологий. Компания основана в 2007 году и уже более 13 лет занимается полным циклом разработки программного обеспечения. За это время были реализованы более 400 проектов в разных отраслях для клиентов из США, Канады, Великобритании, Германии, Франции, Италии, Японии и других стран. Стремление Innowise Group к использованию современного стека технологий и поиску инноваций привели компанию к собственным решениям в области медицины, которые уже получили международное признание и высокие награды (проект VOKA.IO, основанный на применении MR технологий в хирургии, победил в категории “Здоровье и благополучие” всемирного конкурса инновационных проектов World Summit Awards 2019).

Сотрудники Innowise Group работают над масштабными проектами для крупных международных компаний и собственными инновационными продуктами, постоянно внедряя новейшие технологии и передовые решения. Многие проекты предполагают возможность деловых поездок за границу.

Ядро Innowise Group составляют настоящие профессионалы, благодаря чему продукция компании отвечает всем потребностям своей целевой аудитории, независимо от жанровых предпочтений и возраста клиентов.

					ЯНА.1710412.ПЗ	6
Изм.	Лист	№	Подп.	Дата		

1.2 Организационная структура организации

Проанализировав организационную структуру, я сделал вывод, что на данном предприятии преобладает комбинированная структура управления. Такая организационная структура устанавливается исходя из целей деятельности и необходимых для достижения этих целей подразделений, выполняющих функции, составляющие бизнес-процессы организации.

Структурная схема предприятия представлена на рисунке 1.

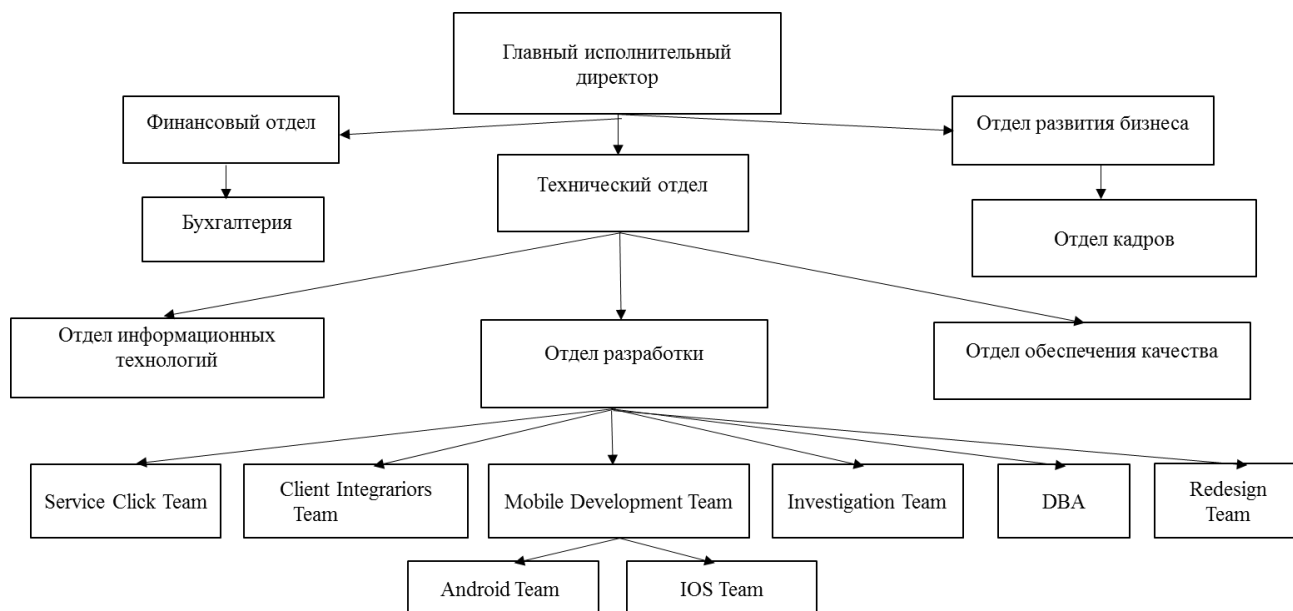


Рисунок 1.1 – Структурная схема предприятия «Фабрика инноваций и решений»

2 АНАЛИЗ ФУНКЦИОНИРОВАНИЯ СТРУКТУРНЫХ ПОДРАЗДЕЛЕНИЙ

2.1 Анализ функций предприятия

Среди клиентов Innowise Group как частные компании, так и влиятельные заказчики госсектора. В режиме аутсорсинга выполняются как краткосрочные задачи по отдельным услугам, так и долгосрочные проекты, в том числе по полному циклу.

Лидирующей технологией на данный момент является фреймворк для создания веб-приложений Spring Framework. Языком написания скриптов является Java.

2.2 Анализ функций, целей и задач IT- отдела

В активе Innowise Group есть проекты различных типов: разработка приложений для предприятий госсектора, ряд приложений для VR/AR устройств и многие другие.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НА ЗАКАЗ

Одно из основных направлений деятельности компании – разработка программного обеспечения на заказ (аутсорсинг). Типичный заказчик Innowise Group обращается в компанию, когда для его задач не существует достаточно удобного и функционально соответствующего готового универсального решения.

Команда Innowise Group полностью берет процесс под свой контроль. Создание решения начинается с четкого формулирования требований и анализа потребностей клиента. Далее обязательно следует создание визуальных моделей и предоставление их заказчику на согласование. Специалисты Innowise Group всегда активно участвуют в процессе согласования и дают клиенту экспертные рекомендации.

Разработка проекта ведется с использованием безопасной и гибкой методики Scrum. При планировании разработки используется декомпозиция с оптимальным размером фрагментов. Методология непрерывной интеграции позволяет управлять рисками, что существенно ускоряет выход продукта.

Эффективные коммуникации и отслеживание ошибок обеспечиваются использованием системы JIRA, а хранение всех промежуточных версий кода – применением Git. Для различных уровней (разработка, продакшн, поддержка, тестирование) применяются отдельные среды, что исключает возникновение проблем и обеспечивает поступление заказчику исключительно рабочих сборок ПО.

Для тестирования разрабатываемого программного обеспечения в Innowise Group выделен специальный отдел, что позволяет обеспечить не только полное соответствие разрабатываемого решения требованиям технического задания, но и снизить риски, связанные с его поведением в нештатных ситуациях.

					ЯНА.1710412.ПЗ	8
Изм.	Лист	№	Подп.	Дата		

Применение современных методик и контроль процесса разработки на всех этапах позволяет минимизировать влияние человеческого фактора. Это обеспечивает неизменно высокое качество результата, а также соблюдение сроков и бюджета проекта.

3 НОРМАТИВНАЯ ДОКУМЕНТАЦИЯ

3.1 Содержание должностных инструкций

1. ОБЩИЕ ПОЛОЖЕНИЯ

1. Ведущий инженер-программист относится к категории специалистов, принимается на работу и увольняется с работы приказом руководителя.

2. На должность ведущего инженера-программиста назначается лицо, имеющее высшее образование соответствующей квалификации.

3. В своей деятельности ведущий инженер-программист руководствуется:

- нормативными документами по вопросам выполняемой работы;
- уставом организации;
- правилами трудового распорядка;
- приказами и распоряжениями руководителя организации (непосредственного руководителя);
- настоящей должностной инструкцией.

4. Ведущий инженер-программист должен знать:

- руководящие и нормативные материалы, регламентирующие методы разработки алгоритмов, программ и использования вычислительной техники при обработке информации;
- основные принципы структурного и объектно-ориентированного программирования;
- виды программного обеспечения;
- стандарты программной документации;
- основные методы, средства и методологии разработки программного обеспечения;
- стандартные алгоритмы;
- языки программирования;
- технологию автоматизированной обработки информации;
- виды и порядок оформления технической документации;
- стандарты качества программного обеспечения;
- основные общепринятые технологии в области разработки программных продуктов;
- технико-эксплуатационные характеристики, конструктивные особенности, назначение и режимы работы оборудования, правила его технической эксплуатации;

					ЯНА.1710412.ПЗ	10
Изм.	Лист	№	Подп.	Дата		

- основы организации труда и управления;
- основы законодательства о труде;
- правила и нормы охраны труда и пожарной безопасности.

5. Во время отсутствия ведущего инженера-программиста его обязанности выполняет в установленном порядке назначаемый заместитель, несущий полную ответственность за надлежащее исполнение возложенных на него обязанностей.

2. ДОЛЖНОСТНЫЕ ОБЯЗАННОСТИ

Для выполнения возложенных на него функций ведущий инженер-программист обязан:

1. На основе анализа математических моделей и алгоритмов решения научных, прикладных экономических и других задач разрабатывать программы, обеспечивающие возможность выполнения средствами вычислительной техники алгоритма и поставленной задачи.
2. Участвовать в выборе языка программирования для описания алгоритмов и структур данных.
3. Разрабатывать технологию решения задачи на всех этапах.
4. Определять информацию, подлежащую обработке средствами вычислительной техники, ее объемы, структуру, макеты и схемы ввода, обработки, хранения и вывода, методы ее контроля.
5. Определять объем и содержание данных контрольных примеров, обеспечивающих наиболее полную проверку соответствия программ их функциональному назначению.
6. Осуществлять запуск отлаженных программ и ввод исходных данных, определяемых условиями поставленных задач.
7. Определять возможность использования готовых программных продуктов.
8. Разрабатывать, отлаживать, анализировать и оптимизировать программный код на основе готовых спецификаций.
9. Интегрировать программные компоненты.
10. Проводить проверку программ на основе логического анализа.
11. Проводить отладку разработанных программ, корректировать их в процессе стабилизации и сопровождения.
12. Выполнять работу по унификации процессов разработки программ.
13. Разрабатывать инструкции по работе с программами, оформлять необходимую техническую документацию.
14. Разрабатывать и внедрять методы автоматизации программирования.

					ЯНА.1710412.ПЗ	11
Изм.	Лист	№	Подп.	Дата		

15. Участвовать в сопровождении программного обеспечения.
16. Оказывать помощь специалистам на различных стадиях разработки программного обеспечения при сборе и документировании требований пользователя, в разработке спецификаций.
17. Вести и представлять установленную отчетность.
18. Своевременно и точно исполнять производственные приказы, задания, указания и распоряжения руководителя.
19. Соблюдать установленные на предприятии Правила внутреннего трудового распорядка.
20. Оказывать содействие и сотрудничать с нанимателем в деле обеспечения здоровых и безопасных условий труда, немедленно сообщать непосредственному руководителю о каждом случае производственного травматизма и профессионального заболевания, а также о чрезвычайных ситуациях, которые создают угрозу здоровью и жизни для него и окружающих, обнаруженных недостатках и нарушениях охраны труда.
21. Принимать необходимые меры по ограничению развития аварийной ситуации и ее ликвидации.

3. ПРАВА

Ведущий инженер-программист имеет право:

1. Знакомиться с проектами решений руководства организации, касающимися его деятельности.
2. Вносить на рассмотрение руководства по совершенствованию информационной системы организации в целом.
3. Получать от руководителей структурных подразделений, специалистов информацию и документы, необходимые для выполнения своих должностных обязанностей.
4. Привлекать специалистов всех структурных подразделений организации для решения возложенных на него обязанностей (если это предусмотрено положениями о структурных подразделениях, если нет - с разрешения руководителя организации).
5. Требовать от руководства организации оказания содействия в исполнении своих должностных обязанностей и прав.
6. Принимать участие в обсуждении вопросов охраны труда, выносимых на рассмотрение собраний (конференций) трудового коллектива (профсоюзной организации).

4. ОЦЕНКА РАБОТЫ И ОТВЕТСТВЕННОСТЬ

Работу ведущего инженера-программиста оценивает непосредственный руководитель (иное должностное лицо).

Ведущий инженер-программист несет ответственность:

					ЯНА.1710412.ПЗ	12
Изм.	Лист	№	Подп.	Дата		

1. За неисполнение (ненадлежащее исполнение) своих должностных обязанностей, предусмотренных настоящей должностной инструкцией - в пределах, определенных действующим трудовым законодательством Республики Беларусь.

2. За совершенные в процессе осуществления своей деятельности правонарушения – в пределах, определенных действующим административным, уголовным и гражданским законодательством Республики Беларусь.

3. За причинение материального ущерба – в пределах, определенных действующим трудовым, уголовным и гражданским законодательством Республики Беларусь.

4. За несоблюдение правил и норм охраны труда, техники безопасности, производственной санитарии и противопожарной защиты – в соответствии с требованиями нормативных правовых актов Республики Беларусь и локальных актов.

3.2 Модели жизненного цикла программного обеспечения

Стандарт ISO/IEC 12207 не предлагает конкретную модель ЖЦ и методы разработки ПО. Стандарт ISO/IEC 12207 не предлагает конкретную модель ЖЦ и методы разработки ПО (под моделью ЖЦ понимается структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач, выполняемых на протяжении ЖЦ. Модель ЖЦ зависит от специфики ИС и специфики условий, в которых последняя создается и функционирует). Его регламенты являются общими для любых моделей ЖЦ, методологий и технологий разработки. Стандарт ISO/IEC 12207 описывает структуру процессов ЖЦ ПО, но не конкретизирует в деталях, как реализовать или выполнить действия и задачи, включенные в эти процессы.

К настоящему времени наибольшее распространение получили следующие две основные модели ЖЦ:

- каскадная модель (70-85 гг.);
- спиральная модель (86-90 гг.).

Стандарт ISO 12207 – Процессы жизненного цикла программного обеспечения – наиболее полно на уровне международных стандартов отражает жизненный цикл, технологию разработки и обеспечения качества сложных программных средств. Жизненный цикл ПО представлен набором этапов, частных работ и операций в последовательности их выполнения и взаимосвязи, регламентирующих ведения разработки на всех стадиях от подготовки технического задания до завершения испытаний ряда версий и окончания эксплуатации ПО. В ЖЦ включаются описания исходной информации, способов выполнения

операций и работ, устанавливаются требования к результатам и правилам их контроля, а также к содержанию технологических и эксплуатационных документов. Определяется организационная структура коллективов, распределение и планирование работ, а также контроль за реализацией ЖЦ ПО.

Стандарт определяет архитектуру, процессы, разделы и подразделы ЖЦ ПО, а также перечень базовых работ и детализирует содержание каждой из них. Архитектура ЖЦ ПО в стандарте базируется на трех крупных компонентах:

- основные процессы жизненного цикла ПО и определяющие работы;
- вспомогательные процессы и работы, поддерживающие жизненный цикл ПО;
- организационные процессы и управление жизненным циклом ПО.

Процессы приобретения и/или подготовки к созданию ПО должны начинаться с инициализации проекта, анализа концепции, анализа рынка продуктов, выработки требований и состава поддерживающих документов, создания предварительного плана проекта. Основные работы по созданию сложного комплекса программ рекомендуется начинать с определения состава сопровождающих документов, выбора средств конфигурационного управления и обеспечения качества, а также выбора методов и средств технологического обеспечения разработки всей информационной системы. Кодирование и тестирование каждого компонента ПО должно быть оформлено совокупностью документов, удостоверяющих соответствие компонента первичной спецификации, содержащих тесты и результаты тестирования.

Рекомендуется разрабатывать план работ, включающий комплексирование компонентов, тестирование по всем разделам требований и показателям качества, а также документирование плана, результатов интеграции, использованных тестов, критериев оценки и полученных результатов. Далее программное обеспечение следует подвергать квалификационному (аттестационному) тестированию по всем разделам требований контракта, при широком варьировании тестов, изменениях значений критериев, а также тестировать полноту и адекватность технологической и пользовательской документации реальному программному продукту. Проверенный таким образом комплекс программ интегрируется в вычислительные средства информационной системы, средства визуализации и телекоммуникации.

Эти работы взаимодействуют с работами, обеспечивающими сопровождение программного обеспечения. Специалисты анализируют сообщения об ошибках и предложения на модификацию программного обеспечения, отбирают их на соответствие требованиям контракта и оценивают целесообразность проведения изменений.

					ЯНА.1710412.ПЗ	14
Изм.	Лист	№	Подп.	Дата		

Подготовленные изменения тестируются и проверяются по критериям, определенным в документации.

Процессы документирования программного обеспечения должны охватывать планирование и обеспечение документирования, рекомендации по стандартизации, проектированию и разработке, а также по производству, конфигурационному управлению и сопровождению комплекта документации на ПО. Для обеспечения гарантий качества следует использовать планирование, методологию, процедуры и стандарты поддержки качества программного обеспечения в соответствии с контрактом с учетом доступных ресурсов. Верификация программного обеспечения должна включать ее организацию, планирование и техническое обеспечение. Удостоверение правильности (аттестация) должна гарантировать полное соответствие программного продукта спецификациям, требованиям и документации на программное обеспечение и возможность его надежного функционирования и безопасного применения пользователем.

Организация жизненного цикла программного обеспечения включает основные работы по управлению проектом, производством и средствами для обеспечения процессов по разработке, эксплуатации и сопровождению. Процессы формирования инфраструктуры должны состоять из выбора и установления аппаратных и программных средств, технологии, стандартов и обслуживания, используемых для разработки, сопровождения и обеспечения эксплуатации ПС. Процессы совершенствования жизненного цикла ПС состоят в установлении, оценивании, измерении, контроле и корректировке процессов жизненного цикла конкретного программного обеспечения. Процессы обучения определяются требованиями к проекту, должны учитывать необходимые ресурсы, управление и технические средства.

					ЯНА.1710412.ПЗ	15
Изм.	Лист	№	Подп.	Дата		

4 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, ИСПОЛЬЗУЕМОЕ В ОРГАНИЗАЦИИ

4.1 Анализ степени автоматизации в организации

Программное обеспечение, используемое в подразделении, подразделяется на две группы:

1. Прикладное ПО – совокупность программ, предназначенных для выполнения определенных пользовательских задач и рассчитанных на непосредственное взаимодействие с пользователями.

К группе прикладного ПО относятся:

- профессиональные программные средства;
- программные средства общего назначения;
- программные средства специального назначения.

2. Системное ПО – комплекс программ, которые обеспечивают эффективное управление компонентами компьютерной системы. Системное ПО не решает конкретные прикладные задачи, а обеспечивает работу других программ. В настоящем Положении термин «системное ПО» включает совокупность ПО общего назначения, предназначенного для решения часто встречающихся задач обработки информации;

К группе системного ПО относятся:

- операционные системы (Windows, Linux, Mac OS);
- коммуникационное ПО (ПО, предназначенное для работы с коммуникационным оборудованием (модемами, коммутаторами и т. п.));
- программы создания резервных копий, диагностики, управления памятью и др.

4.2 Анализ используемых программных продуктов

Сотрудники ООО «Фабрика инноваций и решений» и в частности отдела разработки используют свободное программное обеспечение, в том числе разработанное и поддерживаемое самим отделом.

Программное обеспечение, используемое в отделе, подразделяется на две группы:

1. Прикладное ПО — совокупность программ, предназначенных для выполнения определенных пользовательских задач и рассчитанных на непосредственное взаимодействие с пользователями;

К группе прикладного ПО относятся:

- профессиональные программные средства (ведение отчетности)

					ЯНА.1710412.ПЗ	16
Изм.	Лист	№	Подп.	Дата		

— программные средства общего назначения (текстовые редакторы, графические редакторы, архиваторы, файловые менеджеры, программы сканирования и распознавания текста и т. п.);

— программные средства специального назначения (мультимедиа, гипертекстовые системы, системы мониторинга сети, создания и записи образов, удаленного доступа, в т. ч. терминального, FTP-серверы и FTP клиенты, система электронной почты и т. п.).

2. Системное ПО — комплекс программ, которые обеспечивают эффективное управление компонентами компьютерной системы. Системное ПО не решает конкретные прикладные задачи, а обеспечивает работу других программ. В настоящем Положении термин «системное ПО» включает совокупность ПО общего назначения, предназначенного для решения часто встречающихся задач обработки информации;

К группе системного ПО относятся:

- операционные системы (Windows, Linux);
- коммуникационное ПО (ПО, предназначенное для работы с коммуникационным оборудованием (модемами, коммутаторами и т. п.));
- программы создания резервных копий, диагностики, управления памятью и др.

Используемое ПО для разработки:

- IntelliJ IDEA — интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java.
- Продукты компании JetBrains.

4.3 Среды разработки и языки программирования, используемые в организации

Разработка ПО — процесс, направленный на создание и поддержание работоспособности, качества и надежности ПО, включающий изменение действующего ПО, предполагающее создание принципиально нового программного модуля, влекущего изменение технологии обработки информации, перевод ПО на новую платформу (операционную систему или СУБД).

Языки программирования : Java.

Инструментальное ПО — программное обеспечение, предназначенное для использования в ходе проектирования, разработки и сопровождения программ, в отличие от прикладного и системного программного обеспечения.

К группе инструментального ПО относятся:

- средства разработки ПО — трансляторы, компиляторы (.NET, Android, Java и др.), т.е. системы, посредством которых создается новое ПО;

					ЯНА.1710412.ПЗ	17
Изм.	Лист	№	Подп.	Дата		

- системы управления базами данных (PostgreSQL, Oracle, MySQL и др.);
- CASE-средства проектирования информационных систем, предназначенные для моделирования бизнес-процессов, баз данных, компонентов ПО.

4.4 Операционные системы, используемые в организации

В рамках данного предприятия используются операционные системы Windows 7/8/8.1/10, Linux, Mac OS.

					ЯНА.1710412.ПЗ	18
Изм.	Лист	№	Подп.	Дата		

4.5 Требования, предъявляемые к разрабатываемому программному продукту

1. Правильность – функционирование в соответствии с техническим заданием.
2. Универсальность – обеспечение правильной работы при любых допустимых данных и защиты от неправильных данных.
3. Надежность – точность результатов, обеспечение погрешности результатов не выше заданной.
4. Защищенность – обеспечение конфиденциальности информации.
5. Программная совместимость – возможность совместного функционирования с другим программным обеспечением.
6. Аппаратная совместимость – возможность совместного функционирования некоторым оборудованием.
7. Эффективность – использование минимально возможного количества ресурсов технических средств, например, времени микропроцессора или объема оперативной памяти.

5 ИЗУЧЕНИЕ АРХИТЕКТУРЫ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ ПРЕДПРИЯТИЯ

Объединение персональных компьютеров в виде локальной вычислительной сети дает ряд преимуществ:

- разделение ресурсов, которое позволяет экономно использовать дорогостоящее оборудование, например, лазерные принтеры, со всех присоединенных рабочих станций;
- разделение данных, которое предоставляет возможность доступа и управления базами данных и элементами файловой системы с периферийных рабочих мест, нуждающихся в информации. При этом обеспечивается возможность администрирования доступа пользователей соответственно уровню их компетенции;
- разделение программного обеспечения, которое предоставляет возможность одновременного использования централизованных, ранее установленных программных средств;
- разделение ресурсов процессора, при котором возможно использование вычислительных мощностей для обработки данных другими системами, входящими в сеть.

Топология сети тип «ЗВЕЗДА»

Преимущества топологии «ЗВЕЗДА»:

- легко подключить новый ПК;
- имеется возможность централизованного управления;
- легкое объединение рабочих групп;
- сеть устойчива к неисправностям отдельных ПК и к разрывам соединения отдельных ПК;
- недорогой кабель и быстрая установка.

Оборудование

Витая пара – это один из видов кабеля связи. Кабель витая пара представляет собой одну или несколько пар свитых между собой с определенным шагом проводов, покрытых сверху пластиковой оболочкой. Наибольшее распространение на сегодняшний день получила витая пара 5Е категории. Для прокладки сетей используется как витая пара UTP неэкранированная, так и витая пара FTP экранированная. Кабель витая пара UTP является наиболее часто используемым кабелем. Скручивание проводов уменьшает влияние внешних электромагнитных полей на передаваемые сигналы, скорость передачи информации может достигать 10 - 40 Гбит/сек.

По сравнению с волоконно-оптическими и коаксиальными кабелями, использование витой пары обладает рядом существенных преимуществ. Такой кабель более тонкий, более гибкий и его проще устанавливать. Он также недорог. И вследствие этого, витая пара является идеальным средством передачи данных для офисов или рабочих групп, где нет электромагнитных помех.

					ЯНА.1710412.ПЗ	20
Изм.	Лист	№	Подп.	Дата		

Коннектор RJ-45

Для подключения кабеля витой пары используем коннектор RJ-45 5Е. Коннекторы имеют 8 контактов.

Коммутатор

Сетевой коммутатор или свитч (switch – переключатель) – устройство, предназначенное для соединения нескольких узлов компьютерной сети в пределах одного сегмента. В отличие от концентратора, который распространяет трафик от одного подключенного устройства ко всем остальным, коммутатор передаёт данные только непосредственно получателю. Коммутатор с 24 портами 10/100Base-TX + 2 портами 10/100/1000Base-T.

Источник бесперебойного питания

ИБП – источник вторичного электропитания, автоматическое устройство, назначение которого обеспечить подключенное к нему электрооборудование бесперебойным снабжением электрической энергией в пределах нормы.

					ЯНА.1710412.ПЗ	21
Изм.	Лист	№	Подп.	Дата		

6. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

6.1 Постановка задачи

На данной практике была поставлена задача написать RESTful веб-сервис «Тур сервис» с использованием фреймворка Spring Boot.

Заказчиком были выставлены требования для будущего приложения.

Общие требования. На разработку этого проекта было задействовано шесть человек. Все приложение состояло из шести взаимосвязанных между собой микросервисов:

- Аутентификация
- Сервис администратора
- Сервис «Отели
- Сервис «Рейсы»
- Сервис «Тур»
- Сервис подбора туров

Роли.

- Администратор
- Пользователь

6.2 Реализация проекта

Все обновления проект в ходе разработке нужно было оформлять на ресурсе GitHub.

Ссылка на проект: <https://github.com/NikitaYakovlev17/TourSelectionService>.

Мной был выбран сервис подбора туров. Для того чтобы хранить и доставать данные, мне необходимо создать БД, а также заполнить ее данными. У меня получилась БД из шести основных таблиц:

- Hotels
- Seasons
- Locations
- Tours
- Airplanes
- Customers

В данном проекте я использовал систему управления БД PostgreSQL, а также библиотеку для работы с БД Hibernate.

Листинг 6.1 – Скрипт создания таблиц БД «Тур сервис»

```
1: CREATE TABLE customers (
```

					ЯНА.1710412.ПЗ	
Изм.	Лист	№	Подп.	Дата		22

```

2: customer_id INT NOT NULL GENERATED ALWAYS AS IDENTITY PRIMARY
KEY,
3: customer_name VARCHAR(25) NOT NULL,
4: customer_second_name VARCHAR(25) NOT NULL,
5: passport_number VARCHAR(25) NOT NULL UNIQUE
6: );

7: CREATE TABLE hotels (
8: hotel_id INT NOT NULL GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
9: hotel_name VARCHAR(25) NOT NULL,
10: raiting INT CHECK(raiting > 0 AND raiting < 6) NOT NULL
11: );
12: CREATE TABLE seasons (
13: season_id INT NOT NULL GENERATED ALWAYS AS IDENTITY PRIMARY
KEY,
14: season_name VARCHAR(25) NOT NULL,
15: season_start DATE NOT NULL,
16: season_end DATE NOT NULL
17: );
18: CREATE TABLE airplanes (
19: airplane_id INT NOT NULL GENERATED ALWAYS AS IDENTITY PRIMARY
KEY,
20: airplane_class VARCHAR(25) NOT NULL
21: );
22: CREATE TABLE locations (
23: location_id INT NOT NULL GENERATED ALWAYS AS IDENTITY PRIMARY
KEY,
24: country_name VARCHAR(25) NOT NULL,
25: city_name VARCHAR(25) NOT NULL,
26: beach BOOLEAN NOT NULL,
27: hotel_id INT NOT NULL,
28: CONSTRAINT fk_hotel_id FOREIGN KEY (hotel_id) REFERENCES hotels
(hotel_id)
29: );
30: CREATE TABLE tours (

```

```

31: tour_id INT NOT NULL GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
32: tour_name VARCHAR(25) NOT NULL,
33: location_id INT NOT NULL,
34: season_id INT NOT NULL,
35: airplane_id INT NOT NULL,
36: CONSTRAINT fk_location_id FOREIGN KEY (location_id) REFERENCES
locations (location_id),
37:  CONSTRAINT  fk_season_id  FOREIGN  KEY  (season_id)  REFERENCES
seasons (season_id)
38: );

```

После создания таблиц БД, а также связей между ними мне нужно было реализовать передачу данных из БД на сервер. Мной был реализован промежуточный слой DTO.

DTO — это так называемый value-object на стороне сервера, который хранит данные, используемые в слое представления.

Листинг 6.2 – DTO для таблицы Tours

```

1: @Getter
2: @Setter
3: @Builder
4: @NoArgsConstructor
5: @AllArgsConstructor
6: public class TourDTO {
7:     private Long id;
8:     private String tourName;
9:     private LocationDTO location;
10:    private SeasonDTO season;
11:    private Set<AirplaneDTO> airplanes;
12: }

```

ЗАКЛЮЧЕНИЕ

По итогам прохождения производственной практики была изучена структура организации ООО «Фабрика инноваций и решений». А именно, произведен анализ характеристики предприятия, рассмотрена его организационная структура, изучены особенности структуры работы отдела разработки. Было подробно рассмотрено программное обеспечение, используемое в организации и характеристики используемых аппаратных средств. В результате выполнения индивидуального задания, были получены навыки разработки на фреймворке Spring на языке программирования Java.

Поставленные в течение прохождения практики задачи, были достигнуты.

					ЯНА.1710412.ПЗ	25
Изм.	Лист	№	Подп.	Дата		

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. <https://www.jetbrains.com/ru-ru>
2. <https://innowise-group.com/about>
3. <https://spring.io>
4. <https://ru.wikipedia.org>

					ЯНА.1710412.ПЗ	26
Изм.	Лист	№	Подп.	Дата		