**Аннотация**

на дипломный проект

“АСОИ “Автоматизация процесса взаимодействия команды разработчиков программного обеспечения”. Разработка серверной части”

Структура и объем проекта

Дипломный проект состоит из 6 листов графической части и пояснительной записки на 70 страницу. Пояснительная записка состоит из задания, аннотации, введения, шести глав, заключения, списка литературы.

Проект содержит 22 иллюстраций и 14 таблиц. Список литературы включает 14 наименований.

Содержание проекта

В введении определена цель проекта и его актуальность.

В первой главе анализ объекта автоматизации.

Во второй главе описывается программная реализация разработанной информационной системы.

В третей главе рассмотрена политика информационной безопасности.

В четвертой главе рассмотрена организационно-экономическая часть.

В пятой главе рассмотрены вопросы охраны труда.

В шестой главе рассмотрены вопросы энергосбережения.

В заключении производится анализ степени выполнения, приводятся итоги проделанного исследования.

Содержание

Введение………………………………………………………………………………7

1 Анализ объекта автоматизации……………………………………………………8

1.1 Анализ процесса разработки программного обеспечения…………………...8

1.2 Описание проблем и их решение……………………………………………..12

1.3 Программные средства автоматизации процесса взаимодействия команд..13

1.4 Математическая модель……………………………………………………….14

2 Программная реализация информационной системы…………………………..17

2.1 Выбор программных средств реализации ИС……………………………….17

2.2 Разработка программного кода ИС…………………………………………..19

2.3 Руководство пользователя и контрольные примеры………………………..22

3 Политика информационной безопасности………………………………………27

3.1 Цель, принципы и задачи защиты информации в информационной системе….....27

3.2 Методы и средства обеспечения защиты информационных ресурсов…….28

3.3 Средства защиты информации и информационных ресурсов……………...28

4 Организационно-экономическая часть…………………………………………..29

4.1 Общая постановка к технико-экономическому обоснованию. Анализ процесса разработки программного обеспечения..……………………………29

4.2 Расчёт трудоёмкости (производительности)………………………………...32

4.3 Расчёт единовременных затрат……………………………………………….34

4.4 Расчёт годовых текущих издержек…………………………………………...42

4.5 Расчёт показателей экономической эффективности………………………...48

4.6 Организация внедрения системы……………………………………………..50

4.7 Заключение по разделу………………………………………………………..50

5 Охрана труда……………………………………………………………………….52

5.1 Система управления охраной труда на предприятии……………………….52

5.1.1 Политика в области охраны труда………………………………………..52

5.1.2 Ответственность…………………………………………………………...55

5.2 Наблюдение за состоянием условий труда по системе Элмери…………....56

6 Энергосбережение и ресурсосбережение при внедрении новых технологий...60

Заключение…………………………………………………………………………..64

Список использованных источников………………………………………………64

Приложение А (справочное) Характеристики рабочих мест сотрудников предприятия ..66

Ведомость документов к дипломному проекту…………………………………...70

**Введение**

Без информационных технологий сложно представить современную жизнь. Они окружают нас всюду: на работе, дома, на улице. Информационные технологии не стоят на месте и, как и все в нашем мире, движутся вперед, развиваясь и совершенствуясь. Так, на смену технологии пакетной обработки программ на большой ЭВМ в вычислительном центре пришла технология работы на персональном компьютере на рабочем месте пользователя. Начинают широко использоваться в различных областях глобальные и локальные компьютерные сети. Для информационных технологий является вполне естественным то, что они устаревают и заменяются новыми. Но эти технологии не появляются сами по себе. Существует множество специалистов, которые занимаются разработкой и внедрением всё новых и новых единиц программного обеспечения. Со временем, разработчикам необходимо охватывать всё большее количество информации и данных, чтобы выпустить новый и актуальный продукт. Для этого им необходимо прибегать к помощи со стороны и автоматизировать процесс разработки и взаимодействия внутри своей команды для достижения максимального качества своего продукта.

**1 Анализ объекта автоматизации**

* 1. **Анализ процесса разработки программного обеспечения**

Зачастую при разработке программного обеспечения члены команды сталкиваются со многими проблемами, которые необходимо решать для развития проекта. Например, во время создания очередной задачи нужно продумать: какие данные вносить для достаточно подробного описания задачи, в каком формате и по какому шаблону это делать. Ведь очень важно придерживаться некоторого общего шаблона, чтобы после создания задачи одним разработчиком, второму не пришлось собирать воедино всё, что сделал первый. Так же членам команды нужно придумать, как делиться новыми задачами, как обсуждать эти задачи, чтобы о результатах обсуждения могли узнать остальные разработчики. Само собой разумеется, что всем членам команды необходимо иметь возможность контролировать состояние каждой задачи в каждый момент времени, чтобы не начать работать над одной и той же задачей одновременно. При каком бы то ни было взаимодействии, важно понимать, какая роль у вашего собеседника. Часто приходится тратить время на то, чтобы найти человека с конкретной ролью на проекте: тестировщик, разработчик, менеджер и т. д. Очевидно, что всем участникам проекта необходимо обезопасить свою среду взаимодействия и результаты работы от внешних угроз. Процесс выполнения одной технической задачи во время разработки программного обеспечения можно представить функциональной моделью рисунок 1.1.

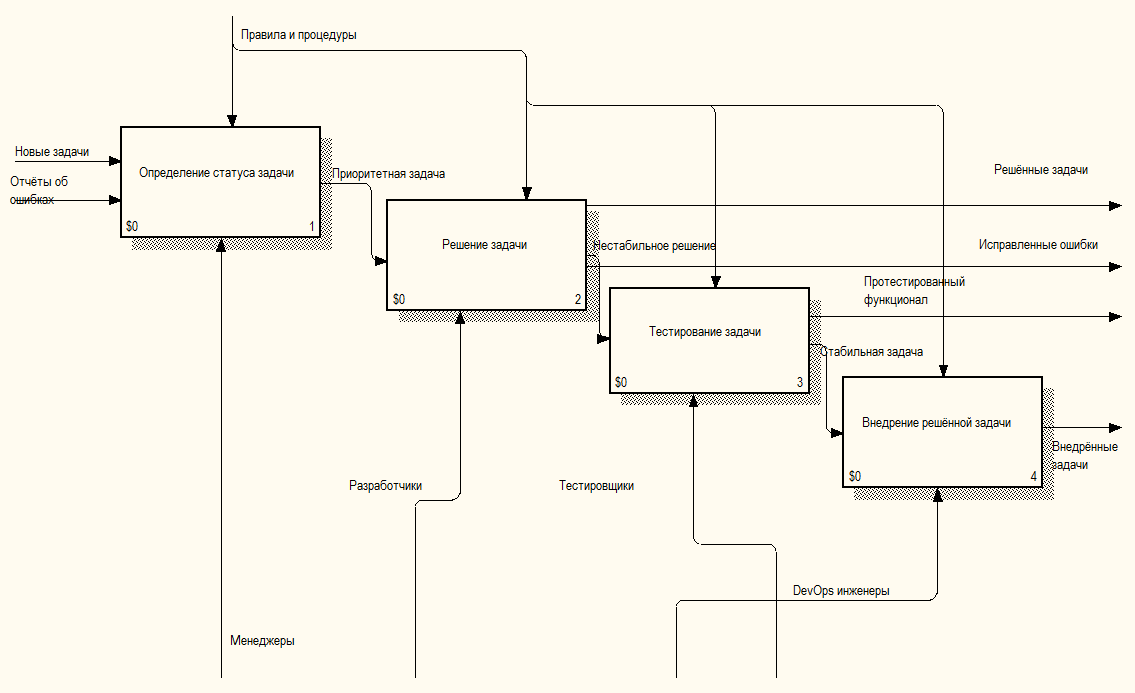


Рисунок 1.1 – Декомпозиция процесса «Выполнение технической  
задачи»

Как видно на данном рисунке, выполнение задачи включает в себя определение статуса и непосредственное создание задачи, решение задачи посредством написания кода, тестирование задачи и внедрение законченного решения в продукт. Результатом этих процессов является законченная и работоспособная часть новой или изменённой логики конечного приложения.

Каждый из вышеперечисленных блоков так же может быть декомпозирован на слои рисунки 1.2 – 1.5.

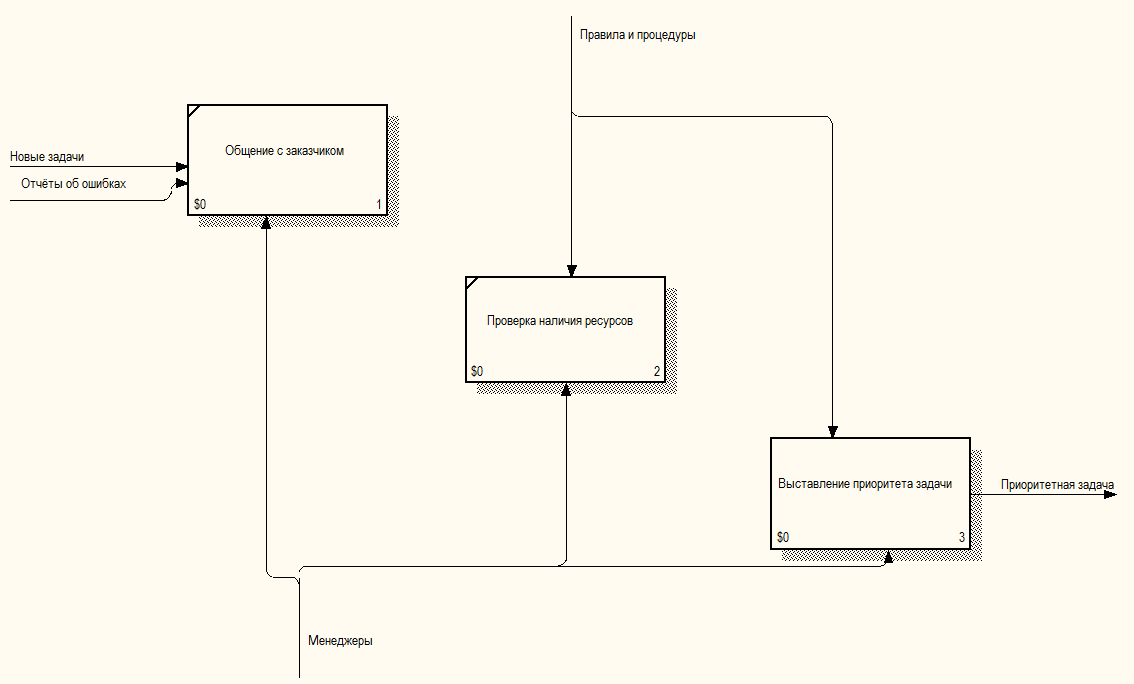


Рисунок 1.2 – Декомпозиция процесса «Определение статуса задачи»

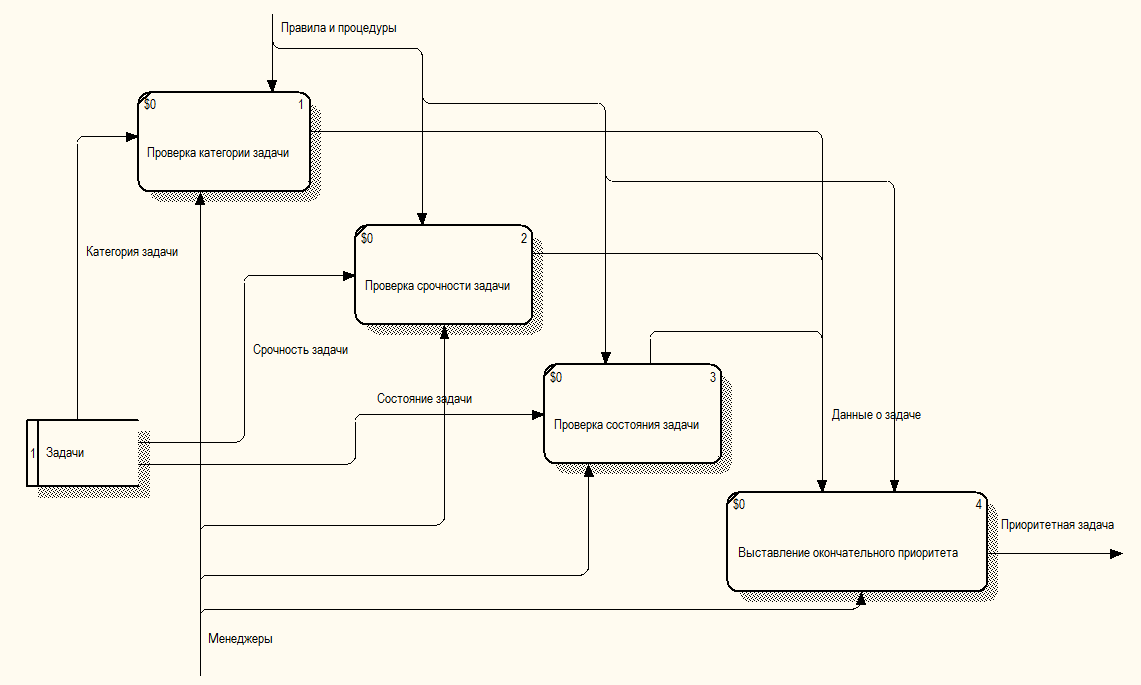


Рисунок 1.3 – Декомпозиция процесса «Решение задачи»

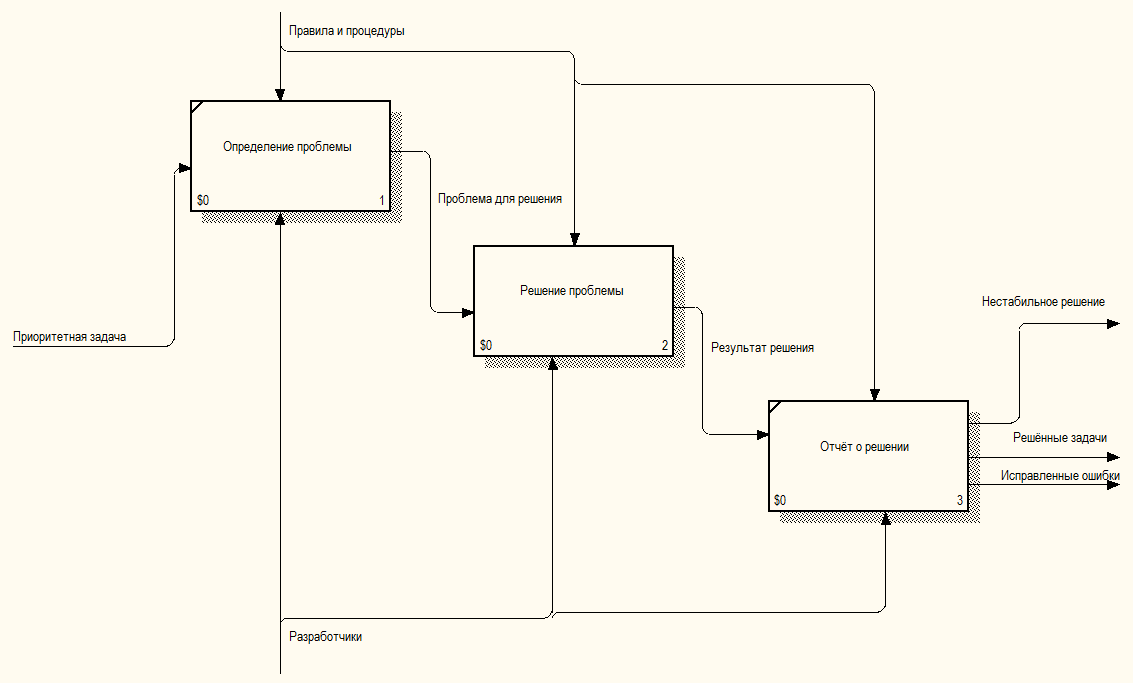


Рисунок 1.4 – Декомпозиция процесса «Тестирование задачи»

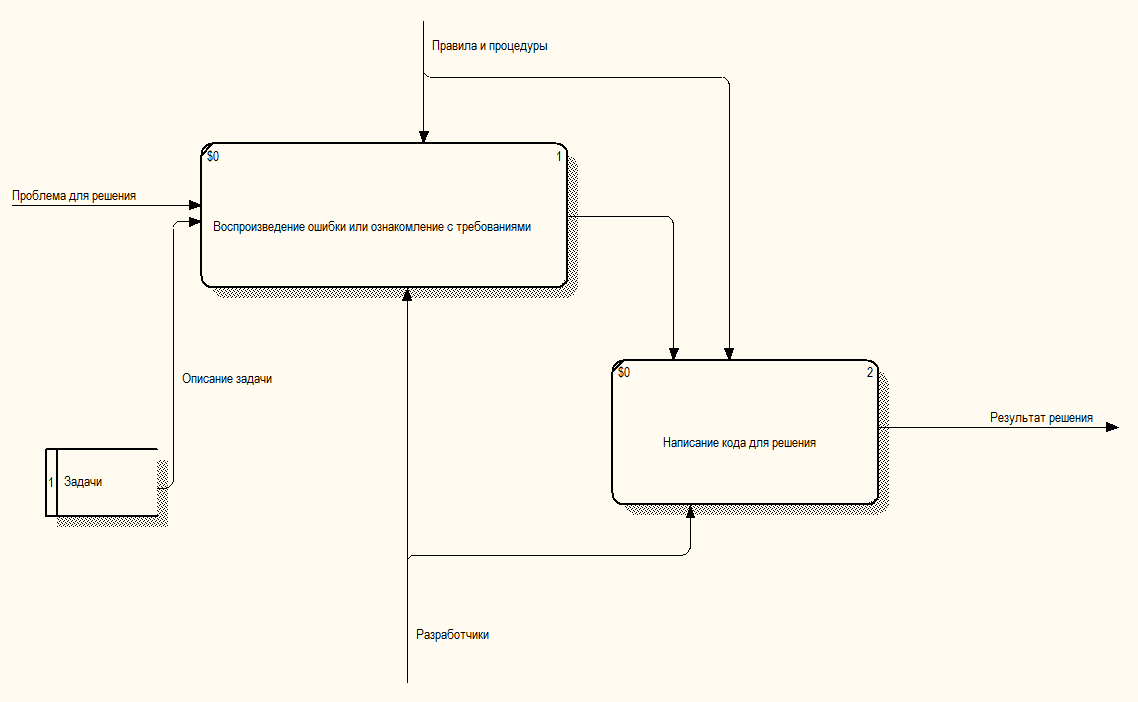


Рисунок 1.5 – Декомпозиция процесса «Внедрение решённой задачи»

Как можно заметить, на каждом из уровней декомпозиции всем членам команды необходимо проделывать множество разнообразных действий. В том числе и по кооперированию друг с другом. Если не прибегать к некоторому дополнительному программному обеспечению, на выполнение всех этих действий уйдёт огромное количество времени. А одна из целей практически любой разработки – это завершение создания продукта за минимальное время. Более того, экономя время на любом виде взаимодействия с командой, каждый отдельно взятый член этой команды может потратить больше времени на повышения качества разрабатываемого продукта. Именно поэтому необходим инструмент, который и позволит эффективнее использовать время команды.

**1.2 Описание проблем и их решение**

Таким образом можно вынести список основных возникающих проблем при разработке ПО, на которые тратиться дополнительное время:

– Обсуждение плана разработки или отдельно взятой задачи. Необходим механизм оценки трудоёмкости задач:

– Предоставление результатов обсуждений всей команде;

– Описание и объяснение задач;

– Контроль задач и сотрудников на проекте;

– Защита всей информации на проекте;

– Контроль доступа к данным.

Данные проблемы можно решить следующим образом:

– Необходим механизм, позволяющий общаться между собой в реальном времени. Результаты общения должны быть видны всем членам команды;

– Необходим механизм оценки трудоёмкости задач;

– Необходим механизм визуализации связей между задачами;

– Необходим механизм полного контроля состояния задач;

– Необходим механизм, для стандартизации всех входных данных;

– Необходим механизм, позволяющий ознакомится с каждым членом команды. – Необходима возможность для связи со всеми участниками проекта;

– Необходим механизм авторизации и аутентификации.

Для решение всех вышеперечисленных проблем при разработке серверной части приложения необходимо создать следующие инструменты. Будет внедрён механизм авторизации и аутентификации посредством JWT-токенов и стандартных средств языков программирования. Будет разработана база данных для хранения всех программных сущностей. Будет разработан алгоритм создания графа, описывающего связи между задачами, а также алгоритм определения критического пути на графе с учётом оценок трудоёмкости задач. Будет разработана API для предоставления клиентской части всех необходимых инструментов, таких как доступ к базе данных, взаимодействия с сущностями в ней. Будет разработан механизм разграничения на роли пользователей. Будет внедрён механизм кэширования часто используемых данных.

**1.3** **Программные средства автоматизации процесса взаимодействия команд**

Уже созданные подобные инструменты, такие как Jira и Azure DevOps, зачастую слишком сложны и не подходят для небольших команд разработчиков, не имеющих опыта в координации и разделении задач во время разработки. Разрабатываемый инструмент, в отличие от вышеперечисленных, проще в освоении и использовании, так как не настолько нагружен функциональными особенностями, которые обычно используются в крупных компаниях или опытными разработчиками при разработке больших приложений.

В Azure, например, при загрузке некоторых компонентов экран просто темнеет. Это сбивает с толку пользователя, заставляя думать, что произошла ошибка. В нашем варианте пользователю показывается специальный загрузочный компонент. Так что пользователь понимает, что идет процесс загрузки данных. Так же в Azure, чтобы добраться до некоторых компонентов приложения, нужно сделать очень много кликов мыши, т.к. эти компоненты имеют глубокую вложенность. В нашем решении все инструменты доступны за один-два клика. У Jira большим недостатком является долгая загрузка всех страниц. Особенно критично это ощущается, когда на каждое новое действие пользователя Jira начинает загружать новую страницу. Так же у Jira по умолчанию настроены ограничения на состояния задач. То есть нельзя свободно перетягивать задачу из одного состояния в другое. Такая возможность появляется только если соблюдены определённые условия, которые могут быть неочевидны пользователю. Таким образом необходим механизм, который учитывал бы все неудобства, которые существуют в похожих реализованных систем.

**1.4 Математическая модель**

При одновременной работе нескольких программистов задачи могут выполняться параллельно. В таком случае разные разработчики выполняют разные задачи. Однако они в силу различных причин тратят на выполнение задач неодинаковое количество времени.

Например, рассмотрим следующий граф (рисунок 1.6). Состояния 1 и 2 являются соответственно начальным и конечным состояниями некоторого разрабатываемого приложения в некотором отрезке времени. Для перехода из состояния 1 в состояние 2 необходимо выполнить все задачи, обозначенные буквами. В процессе параллельной разработки задачи A1, B1-2, C1-2 могут быть выполнены существенно быстрее задач A2, C3. В таком случае разработчики, закончившие раньше, начинают простаивать, что является неэффективным использованием времени.

Разрабатываемое приложение устраняет причины лишних затрат времени.

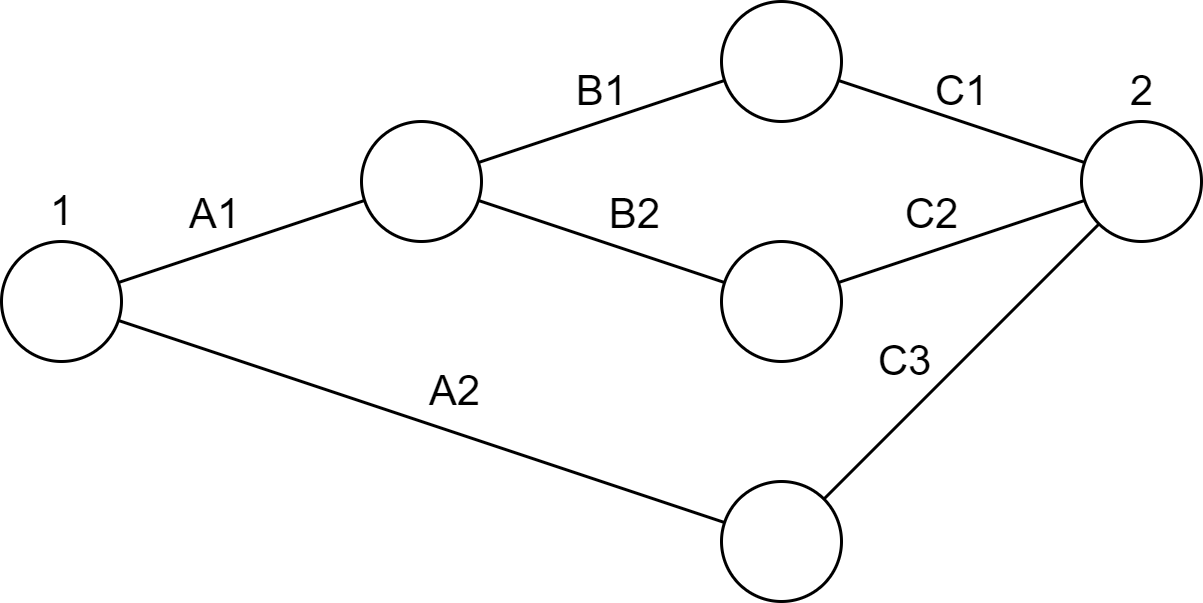


Рисунок 1.6 — Граф состояний проекта

Для начала введём понятие критического пути. Критический путь состоит из самой длительной последовательности задач от начала проекта до его завершения. Тогда время, которое необходимо затратить на разработку всего проекта (*T*), равно времени, затрачиваемом на каждую задачу, лежащую на критическом пути. Тогда:

, (1.1)

где *t* — время, которое необходимо затратить на одну задачу,

*i* — номер задачи на критическом пути.

Каждая задача состоит из одинаковых этапов. Тогда *t* можно записать как:

где *tdes* — время на описание задачи,

*tdev* — время на непосредственно написание кода,

*tcoop* — время, затрачиваемое на взаимодействие членов команды.

Тогда *T* можно представить так:

Из формулы видно, что, уменьшая время на выполнение задач, лежащих на критическом пути, можно достичь сокращения времени на выполнение всего проекта.

**2 Программная реализация информационной системы**

**2.1 Выбор программных средств реализации ИС**

При разработке данной информационной системы в качестве основного средства разработки был выбран язык программирования C#, т.к. он занимает лидирующие позиции среди языков программирования, используемых для написания веб-приложений. C# был использован с платформой .Net 6, т.к. эта платформа обладает всеми необходимыми пакетами для создания гибких и производительный клиент-серверных приложений. Для создание графического интерфейса использовался Java Script в связке с React, который позволяет создавать интерактивные веб-приложения. В качестве базы данных была выбрана MS SQL Server благодаря своим возможностям интеграции с языком C# и многими продуктами компании Microsoft. Приложение уже находится в открытом доступе по ссылке https://justinmind.azurewebsites.net. Так же приложение может использоваться локально, для этого необходимо произвести предварительную настройку базы данных и окружения. После этого приложение можно свободно использовать на локальной машине. При некоторых дополнительных доработках есть возможность внедрить приложение на кафедры вузов для контроля учебного процесса. Сфера использования приложения не ограничена конкретными рамками, оно может быть использовано в различных целях. Во время разработки приложения было использовано большинство современных и эффективных подходов и практик. Далее дано их описание.

API — Application Programming Interface (интерфейс для программирования приложений). То есть это некоторый интерфейс, позволяющий клиентскому приложению обращаться к серверному приложению. Таких клиентских приложений может быть несколько, и они могут обращаться к одному серверному приложению через общий интерфейс.

REST — Representational State Transfer (передача состояния представления). Это актуальный архитектурный подход для создания API.

Redux — популярный менеджер состояний в веб-приложениях. Обычно его используют в связке с React, но поддержка не ограничена только этой популярной JS-библиотекой

Saga — это библиотека, которая призвана упростить и улучшить побочные эффекты (т.е. такие действия, как асинхронные операции, например, загрузки данных, и "грязные" действия, такие, как доступ к браузерному кешу), сделать лёгкими в тестировании и лучше справляться с ошибками.

SOLID — это аббревиатура пяти основных принципов проектирования в объектно-ориентированном программировании — Single responsibility, Open-closed, Liskov substitution, Interface segregation и Dependency inversion. В переводе на русский: принципы единственной ответственности, открытости / закрытости, подстановки Барбары Лисков, разделения интерфейса и инверсии зависимостей

KISS — это принцип проектирования и программирования, при котором простота системы декларируется в качестве основной цели или ценности. Есть два варианта расшифровки аббревиатуры: keep it short and simple.

Паттерн Repository — Репозиторий позволяет абстрагироваться от отдельных подключений к источникам данных, с гарантией работы программы, и является промежуточным звеном между классами, взаимодействующими с данными, и структурами.

Паттерн Singleton — порождающий паттерн, который гарантирует, что для определенного класса будет создан только один объект, а также предоставит к этому объекту точку доступа.

Dependency Injection — стиль настройки объекта, при котором поля объекта задаются внешней сущностью. Другими словами, объекты настраиваются внешними объектами. DI — это альтернатива самонастройке объектов.

ORM Dapper — технология программирования, которая связывает базы данных с концепциями объектно-ориентированных языков программирования, создавая «виртуальную объектную базу данных»

Механизм CI/CD — Непрерывная интеграция (Continuous Integration, CI) и непрерывная поставка (Continuous Delivery, CD) представляют собой культуру, набор принципов и практик, которые позволяют разработчикам чаще и надежнее развертывать изменения программного обеспечения.

FIRST — 5 простых принципов которым должны соответствовать юнит тесты. Fast, Independent, Repeatable, Self-Validating, Timely.

**2.2 Разработка программного кода ИС**

ASP.Net приложения включают в себя несколько основных частей. При проектировании приложения было принято решение придерживаться стандартной трёхуровневой архитектуры. Это значит, что приложение делится на серверную часть, часть доступа к данным и часть бизнес логики. Пример того, как выглядит работа стандартной реализации трёхуровневого приложения представлен на рисунке 2.1.

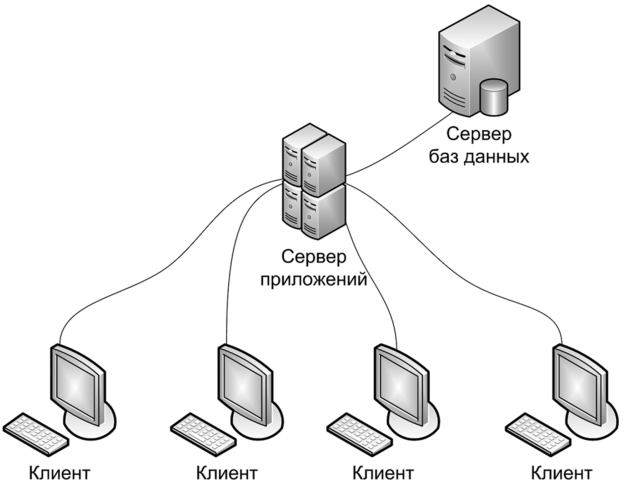


Рисунок 2.1 — Трехуровневая архитектура клиент-сервер

Серверная часть в основном состоит из контроллеров. Это класс задачей которого является ожидание и обработка веб-запросов от некоторого клиента. Контроллеры создаются, чтобы предоставить возможность клиенту выполнять базовые(CRUD) операции над некоторой сущностью: чтение, запись, изменение, удаление. Пример контроллера по работе с задачами представлен на рисунке 2.2.

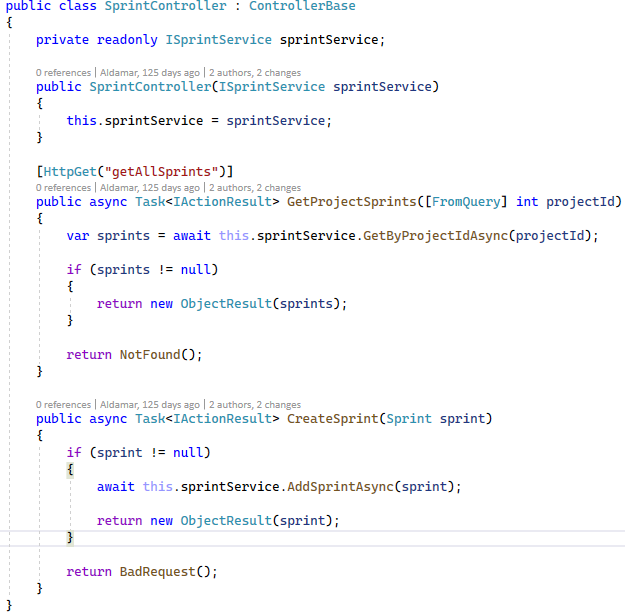


Рисунок 2.2 – Фрагмент кода контроллера

Часть бизнес логики представляет собой сервиса, которые выступают прослойкой между серверной частью и частью доступа к данным. Основная роль сервисов, это предварительная обработка данных, полученных из хранилища данных. Обработка происходит перед тем, как отправить данные клиенту. Пример сервиса предоставлен на рисунке…

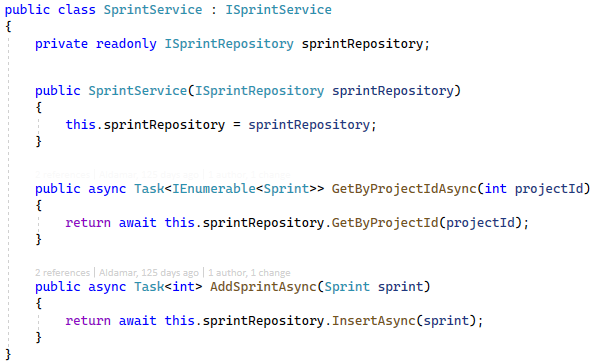


Рисунок 2.3 — Фрагмент кода сервисного класса

Наконец, на рисунке 2.4 представлен пример класса репозитория.

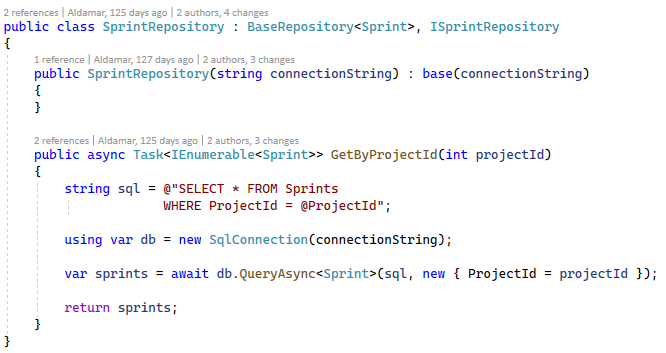


Рисунок 2.4 — Фрагмент кода класса репозитория

Репозитории осуществляют доступ к хранилищу данных и передают полученные данные на уровень сервисов. В этом слое находится только логика по работе с базой данных и ничего больше. Репозитории должны быть свободны от любой посторонней бизнес-логики. В примере выше описан репозиторий, который получает из базы данных все спринты, относящиеся к определённому проекту.

Отдельного внимания заслуживает и контроллер, отвечающий за авторизацию и аутентификацию пользователя. На рисунке 2.5 представлен пример кода, который авторизует пользователя.

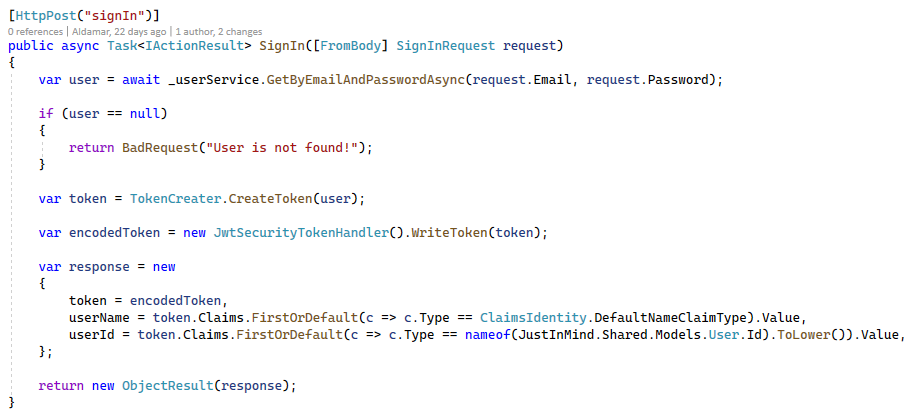


Рисунок 2.5 — Блок авторизации пользователя

**2.3 Руководство пользователя и контрольные примеры**

Если приложение запускается локально, то адрес, по которому к нему можно получить доступ может быть установлен системным администратором. Начало работы с приложением происходит через окно входа рисунок 2.6.

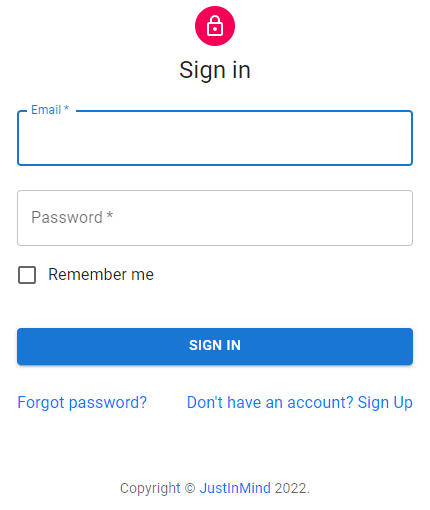


Рисунок 2.6 —Окно входа в приложение

Если у пользователя ещё нет учётное записи, то ему предоставляется возмодность пройти регистрацию, нажав на соответствующую ссылку Sign Up рисунок 2.7. При нажатии на эту ссылку откроется окно регистрации рисунок.

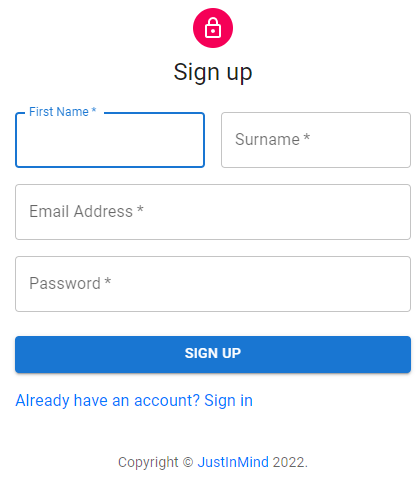


Рисунок 2.7 — Окно регистрации в приложение

После прохождения процедуры аутентификации и авторизации пользователю представляется основное рабочее окно. Это окно работы с проектом рисунок 2.8.

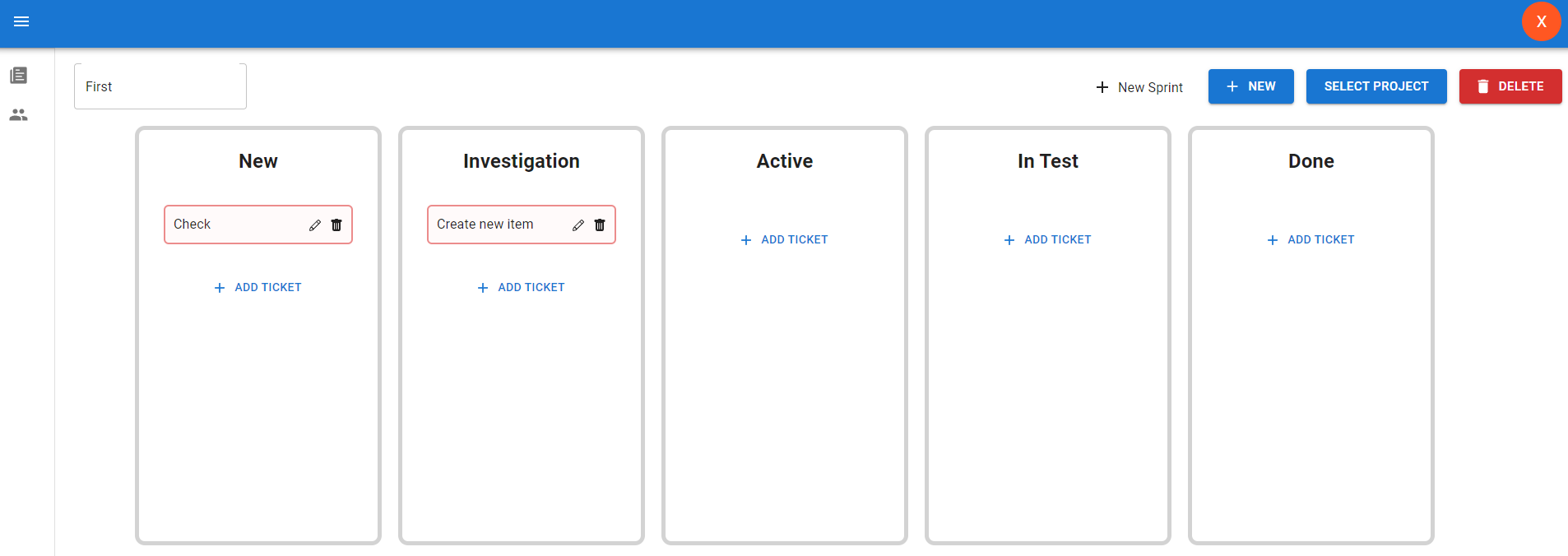


Рисунок 2.8 — Доска задач проекта

Здесь пользователь может создавать новые проекты либо выбрать уже существующие путём нажатия на соответствующие кнопки. Так же при желании есть возможность удалить проект. На доске задач пользователь может перетаскивать задачи из одного состояния в другое или же при нажатии на рисунок карандаша увидеть полное описание задачи рисунок 2.9.

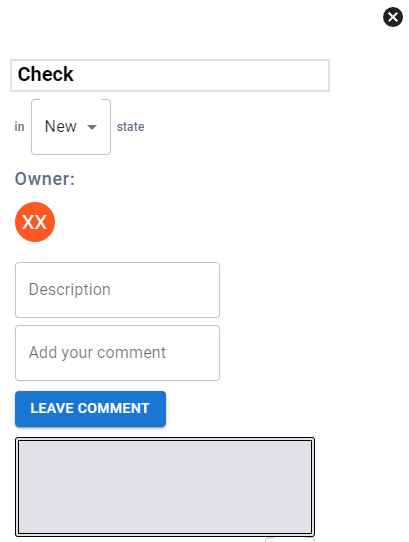


Рисунок 2.9 — Окно редактирования задачи

В этом окне пользователь имеет контроль почти над каждой частью задачи. Есть возможность поменять название, состояние, владельца, описание. Так же можно начать обсуждение, добавив комментарий к задаче.

Если команде необходимо вести разработку на основании методолгоий Agile, то у неё есть возможность создавать спринты. Пример всплывающей формы создания спринта рисунок 2.10.

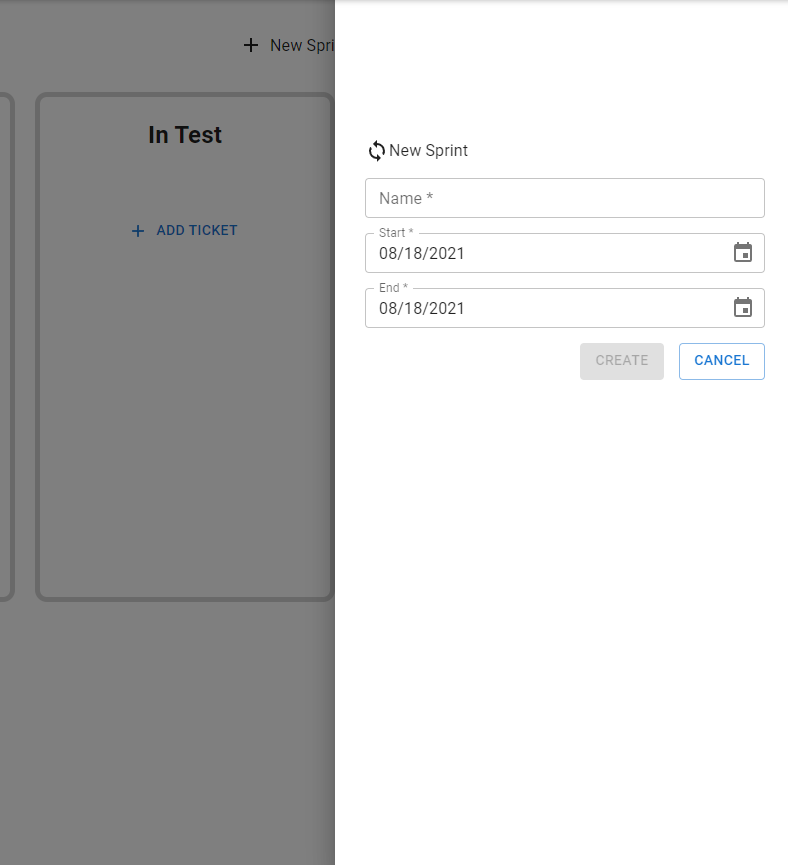


Рисунок 2.10 — Форма создания нового спринта

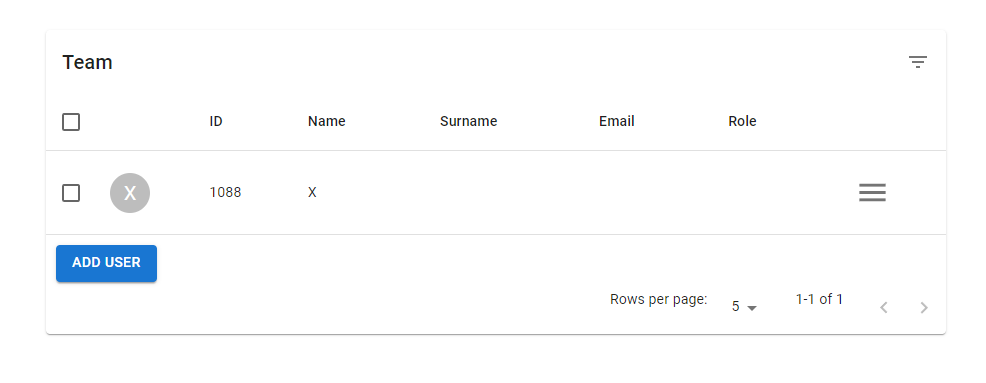
Так же в меню пользователь может увидеть всю команду в виде таблицы рисунок 2.11.  


Рисунок 2.11 — Таблица всех членов проекта

Здесь пользователь может пригласить новых участников, написать сообщение на электронную почту или удалить члена команлы из списка.

# 3 Политика информационной безопасности

**3.1 Цель, принципы и задачи защиты информации в информационной системе**

Информационная безопасность – обеспечение конфиденциальности и целостности информации, недопущение несанкционированных действий с ней, в частности, ее использования, раскрытия, искажения, изменения, исследования и уничтожения.

Главная цель, которая стоит перед системой информационной безопасности является обеспечение защиты данных от внешних и внутренних угроз. Недопущение неправомерного доступа, уничтожения, модификации (изменения), копирования, распространения и (или) предоставления информации, блокирования правомерного доступа к информации, а также иных неправомерных действий

Для обеспечения в информационной системе полной конфиденциальности применяются четыре метода, актуальных для любого формата информации:

* ограничение или полное закрытие доступа к информации;
* шифрование;
* дробление на части и разрозненное хранение;
* скрытие самого факта существования информации.

В данной ИС стоят цели обеспечение правового режима использования информации и информационных ресурсов, обрабатываемых в ИС, как объектов собственности, а также обеспечение и постоянное поддержание в соответствии с условиями, выдвинутыми собственником информационных ресурсов, свойств конфиденциальности, доступности и целостности.

В ИС решена задача снижения риска утечки информации ограниченного доступа в связи с локальным расположением системы и рядом технологий связи, с которыми данная система функционирует.

Решена задача снижения риска несанкционированного воздействия на информацию, система работает с методами защиты данных, разработанными корпорацией Microsoft, они помогают избежать такого рода проблемы в приложении.

**3.2 Методы и средства обеспечения защиты информационных ресурсов**

При разработке данного дипломного проекта не обеспечивалась конфиденциальность используемых данных, т.к. в этом отсутствует острая необходимость, так как используемые данные числу публикаций и мест в рейтинге брались из открытого доступа, также обязанности по обеспечению целостности данных делегируются на системное программное обеспечение, установленное на компьютере пользователя. Примером такой защиты могут выступать учётные записи пользователей, которые требуют ввода пароля при попытке получения доступа, или предустановленная программа BitLocker, которая позволяет уберечь данные в случае утери компьютера.

**3.3 Средства защиты информации и информационных ресурсов**

Для решения защиты информации в разрабатываемой ИС были использованы следующие методы:

* идентификация и аутентификация пользователя;
* шифрование с помощью BitLocker;
* антивирусная защита информационных ресурсов.

**4 Организационно-экономическая часть**

**4.1 Общая постановка к технико-экономическому обоснованию. Анализ процесса разработки программного обеспечения**

Дипломный проект на тему АСОИ “Автоматизация процесса взаимодействия команды разработчиков программного обеспечения”. Разработка серверной части выполнен с целью ускорения и упрощения процесса разработки. Автоматизированный модуль разработан с целью повышения эффективности взаимодействия между участниками команды разработчиков, сокращение времени разработки. Внедрение автоматизированной информационной среды в системе управления рабочим процессом в предприятии, создание клиентской базы по параметрам и критериям управления обеспечат эффективный мониторинг, контроль, редакцию задач и проектов и соответствующее управление участниками процесса разработки.

Целью дипломного проекта является автоматизация операций, выполняемых разработчиком или менеджером проекта при формировании, задач и мониторинге состояний этих задач, обработке и обобщении информации, корректировке направления разработки программного обеспечения. Задачи автоматизации управления задачами, с точки зрения разработки или адаптации ИТ-решения состоят в следующем: создание единого хранилища задач на проекте, самих проектов и всех участниках проекта; обеспечение консистентности хранимых данных; автоматизация процессов создания, утверждения, регистрации и контроля исполнения, рассылки, поиска всевозможной информации касательно проекта; обеспечение оперативного доступа к данным о проектах и сотрудниках; обеспечение интеграции между разными проектами одной команды; обеспечение информационной безопасности (исключение или существенное затруднение возможности получения злоумышленниками защищаемой информации, а также исключение или существенное затруднение возможности несанкционированного и непреднамеренного воздействия на защищаемую информацию и ее носители); разработка серверной части приложения; разработка специализированных программных модулей, обеспечивающих выполнение всех необходимых служебных функций.

Все основные параметры разработанной системы представлены в таблице 4.1.

На данный момент не существует стандартизированной и единой системы взаимодействия для команд разработчиков. Многие аспекты разработки и взаимодействия внутри команды основаны на ручном способе внесения и форматирования данных.

Функционирующая на данный момент система взаимодействия внутри команды разработчиков основана на ручном внесении и обработке данных. Задачи на проекте являются основной единицей данных, с которыми приходится работать всем членам команды.

Таблица 4.1 - Характеристики проектируемой информационной системы

|  |  |
| --- | --- |
| Показатель | Параметр |
| 1 | 2 |
| Область прикладной деятельности | Автоматизированное рабочее место разработчика или менеджера |
| Цель автоматизации | Повышение оперативности обработки информации и взаимодействия, снижение вероятности ошибок |
| Функция программных средств | Обработка данных; поддержка принятия оперативных управленческих решений |
| Уровень автоматизации | Автоматизированный сбор и предоставление информации |
| Порядок внедрения и использования | Документация и обеспечение ее качества; проведение контрольных расчетов |
| Модель данных | Реляционная (табличная) |
| ЭТО КАК-ТО УБРАТЬ | Убрать линию. Над продолжением таблицы пишется “Продолжение таблицы 4.1” |
| Прямая эффективность | Существенное сокращение времени разработки ПО |
| Косвенная эффективность | Актуальность и целостность данных |
|  |  |

Продолжение таблицы 4.1

|  |  |
| --- | --- |
| 1 | 2 |
| Режим эксплуатации обработки данных | Система развёрнута как Web-приложения в режиме реального времени с поддержкой одновременной работы нескольких пользователей. Может использоваться локально. |
| Масштаб программных средств | Свыше 1500 рукописных строк кода |
| Исходный язык | Объектно-ориентированный (C#) |
| Класс пользователя | Любой член команды |
| Требуемые рабочие характеристики | Малая емкость памяти, высокое время обработки, высокая производительность |
| Требование защиты | Защита от несанкционированного доступа посредством авторизации |
| Требование надёжности | Высокая надёжность |
| Требования к вычислительным ресурсам | Intel Core 3 3037U (2x1.8GHz); Socket1150 Chipset FCBGA1023; 8GB DDR3 ; HDD 750GB |

На основании поступающих задач происходит контроль и формирование направления разработки. В функциональные обязанности всех членов команды входят контроль, описание, обсуждение и разрешение всех задач.

В обязанности менеджера входит поиск, создание, структурирование, сортировка и фильтрования задач.

Разработанная информационная система позволит в автоматизированном режиме выполнять описанные выше операций, соответственно сократив сроки взаимодействия внутри команды, повысить оперативность, точность и количество полезной информации, высокое качество расчетов и объективность представляемой информации.

Поскольку базовый вариант является неудовлетворительным по причине низкой скорости обработки информации, а также преимущественного использования в базовом варианте бумажных носителей информации, предлагается программный модуль по поставленным задачам. В таблице 4.2 представлена общая характеристика сравниваемых вариантов.

Таблица 4.2 - Общая характеристика сравниваемых вариантов

|  |  |  |
| --- | --- | --- |
| Наименование показателей | Базовый | Проектный |
| Информационный процесс | Взаимодействие команд разработки | |
| Средства информационного процесса: | | |
| Структурирование данных | Свободный и нестандартизированный формат | Стандартизированный и единый для всех формат |
| Хранение данных | На бумажных носителях или ПК | В базе данных |
| Поиск данных | Ручной поиск | Автоматизированный (получение через пользовательский интерфейс) |
| Обработка и контроль актуальности | Отсутствует | Автоматизированный контроль со стороны сервера |
| Предоставление доступа к данным | Ручной контроль доступа. Рискованный | Авторизация и аутентификация |
| Исполнители процесса | Разработчик, менеджер | |

Для определения эффективности разрабатываемой информационной системы ПИ сравнивают с существующим способом решения аналогичной задачи. При этом рассматриваются следующие варианты: традиционная система обработки информации разработчиком и менеджером проекта (базовый вариант); автоматизированная система обработки информации разработчиком и менеджером проекта (проектируемый вариант).

## 4.2 Расчёт трудоёмкости (производительности)

Разработанная информационная система позволяет повысить производительность труда разработчика или менеджера проекта за счёт автоматизированных механизмов взаимодействия. Функционально норма штучно-калькуляционного времени на решение задачи складывается из следующих элементов:

, (4.1)

где tПЗ - подготовительно-заключительно время на партию решаемых задач;

nп - количество последовательно решаемых задач за один прогон;

tОП - оперативное время выполнения задачи (сумма основного и вспомогательного неперекрываемого времени);

tОБ - время обслуживания рабочего места;

tОТЛ - время на отдых и личные надобности.

Время tОБ и tОТЛ чаще определяется косвенно как доля от оперативного времени tОП в размере 0,12 - 0,16.

Результаты расчета трудоемкости произведены на основе нормативной трудоемкости, норма штучно-калькуляционного времени приведена в таблице 4.3

Таблица 4.3 - Трудоёмкость решения задачи по вариантам решения задачи

|  |  |  |
| --- | --- | --- |
| Наименование элементов нормы времени | Норма времени по вариантам формирования перечня технических заданий для ИТ разработчиков (tШК), мин | |
| базовый | проектный |
| Документ “Регламент технических задач для инженера(разработчика или менеджера)” | | |
| Подготовительно-заключительное время | 1,00 | 1,00 |
| Оперативное время | 29,31 | 14,33 |
| Время обслуживания | 4,04 | 2,57 |
| Время на отдых и личные надобности | 2,11 | 2,00 |
| Итого на задачу | 36,51 | 18,9 |

Годовая программа АГ по полному процессу описания и завершения технической задачи принята в количестве 24 задач за одну рабочую смену (АГ = 23\*255 = 5865 задач)

## 4.3 Расчёт единовременных затрат

По вариантам сравнения единовременные затраты (инвестиции) складываются из следующих основных элементов:

(4.2)

где КО - стоимость комплекта машин и оборудования с учётом офисной мебели, р.;

КОБ - стоимость запасов в оборотные средства, р.;

КЗД - стоимость потребной площади здания, р.;

КПР - затраты на проектирование, р.

Стоимость единовременных затрат в оборудование определяются по формуле

(4.3)

где Nni - принятое число единиц i-го оборудования (Nni ≥ Npi - до ближайшего целого в большую сторону или целая часть Ni, если дробная часть Npi<0,1), шт;

Poi - цена приобретения i-го оборудования по варианту, р.;

αTi, αMi - коэффициенты, учитывающие величину транспортно-заготовительных расходов (αTi = 0,05 - 0,10), величину затрат на монтаж и отладку (αMi = 0,05-0,10);

dз - доля занятости принятых рабочих мест

dЗ = NP/NП.

Расчётное количество машин (рабочих мест) вычисляется по формуле

(4.4)

где FД - годовой действительный фонд работы оборудования (рабочего времени), ч;

kЗ - коэффициент запаса, учитывающий неравномерность поступления информации (kЗ=0,9).

Величина годового действительного фонда рабочего места оператора определяется по формуле

, (4.5)

где FСМ - номинальный сменный фонд работы, ч;

КСМ - коэффициент сменности - число смен работы в течение рабочего дня;

DР - число рабочих дней в году (принято рабочих 255 дней, из них 252 с полной продолжительностью и 3 с сокращенной продолжительностью);

КПР - коэффициент, учитывающий долю времени простоев в плановых ремонтах, КПР=0,03-0,06. Для расчёта принять Kпр = 0.06.

Таким образом, подставив значения для нахождения расчётного количества машин (рабочих мест), имеем

Определим принятое количество рабочих мест путём округления их расчётной величины NР до ближайшего целого числа в большую сторону:

Соответственно, доля занятости принятых рабочих мест на решение задачи по вариантам:

В дипломном проекте для технического обеспечения программного модуля на рабочем месте разработчика предусмотрена модернизация компьютерного оборудования. Стоимость оборудования АРМ разработчика или менеджера представлена в таблице 4.4.

Таблица 4.4 - Цены на оборудование рабочего места разработчика

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование оборудования | Количество | Цена, р. | Стоимость, р. |
| Моноблок Тесла Директор / Q24B2 | Корпус Q24b2 | 1 | 2157 | 2157 |
| Windows 10 Pro | 1 | 978,63 | 978,63 |
| МФУ PANTUM M6500W | 1 | 649 | 649 |
| Итого | - | - | 3784,63 |

Определена стоимость единовременных затрат в оборудование по вариантам, используя формулу стоимости единовременных затрат в оборудование:

Стоимость оборотных средств, связанных с решением задачи по базовому и проектируемому вариантам, рассчитываются по формуле:

(4.6)

где PMj - цена приобретения j-го материала, используемого при решении задачи по варианту, р.;

ZMj - средний запас j-го материала.

Цены на расходные материалы представлены в таблице 4.5

Таблица 4.5 - Используемые материалы по вариантам

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование материалов | Базовый | | Проектный | |
| Цена за 1 ед., р. | Запас, шт. | Цена за 1 ед., р. | Запас, шт. |
| Бумага А4 “Снегурочка”, 500 л., 80 г/м2 | 18,00 | 4 | 18,00 | 1 |
| Заправочный комплект Pantum TN-420 H (3000 страниц) | 36,00 | 2 | 36,00 | 1 |

Определена стоимость оборотных средств по вариантам использования

Стоимость потребной площади здания определяются по формуле:

, (4.7)

где SЗДПР, SЗДСБ - нормативы производственной и служебно-бытовой площадей, SЗДПР=8 м2 и SЗДСБ =6 м2;

РЗДПР, РЗДСБ - цены (стоимости) 1 м2 производственного и служебно-бытового зданий, РЗДПР= 170 долл. и РЗДСБ = 230 долл. (по курсу НБ РБ 2,51 руб./долл. США на 17.05.2022).

Рассчитаем стоимость потребной площади здания для решения поставленной задачи:

Затраты на проектирования для базового варианта не включается в расчёт единовременных затрат. Произведён расчёт затрат на проектирование АСОИ в рамках решаемой задачи по предварительному анализу лёгких. Затраты на проектирование определяются как сметная стоимость работ (постановка задачи и её моделирование, программирование, создание информационного обеспечения длительного пользования, отладка и внедрение разработанной системы) по формуле:

(4.8)

где РПР - сметная ставка 1 чел.-мес. Проектирования, тыс.р.;

ТПР - трудоёмкость проектирования, чел.-мес.;

Дi и Дi+1 - величина дефектности для исходного уровня качества (по базовому варианту - iσ, по проектируемому - (i+1)σ);

КВД и КНД - коэффициенты уровня трудовых затрат на устранение выявленных и не выявленных дефектов;

КДУ - уровень выявления дефектов в программном изделии в процессе проведения тестирования.

Сметная ставка одного человеко-месяца проектирования рассчитывается в рублях по формуле:

(4.9)

где ЗТ - месячная ставка 1-го разряда, ЗТ=200 р.;

КТ - тарифный коэффициент проектирования, КТ12=2,84;

КП - коэффициент премирования, КП=1,5;

КД - коэффициент, учитывающий дополнительную заработную плату, КД=0,1;

КСС - коэффициент, учитывающий отчисления на социальные нужды, КСС=0,347;

КНР - коэффициент, учитывающий накладные расходы, КНР=0,7.

Трудоёмкость проектирование ПИ в человеко-месяцах в соответствии с конструктивной моделью стоимости рассчитывается по следующей формуле:

(4.10)

где АТ, В - коэффициенты конструктивной модели стоимости по принятому типу проекта.

Коэффициент В изменяется в диапазоне 1,01 - 1,26 и зависит от пяти масштабных факторов Wi (в таблице 4.5 факторы Wi оцениваются экспертно рангом из шести уровней: от очень низкого с оценкой 5 баллов до сверхвысокого с оценкой 0 баллов).

На основании экспертных оценок коэффициент вычисляется по формуле:

(4.11)

KLOC - количество тысяч строк в программном продукте без учёта строк, полученных в результате автоматического генерирования кодов, KLOC = 1.5 тыс. строк;

МР - поправочный множитель, который зависит от 15 факторов затрат конструктивной модели стоимости на основании принятых характеристик факторов для проекта и значений численных значений множителей Mi (таблица 4.7),

(4.12)

Тauto - затраты на автоматически генерируемый программный код,

, (4.13)

где KALOC - количество строк автоматически генерируемого кода, KALOC = 10,0 тыс.

АТ - процент автоматически генерируемого кода, АТ= 10/11.5 86,96%;

ATPROD - производительность автоматически генерируемого кода, ATPROD=0.5 тысяч строк в месяц.

Определим затраты на автоматически генерируемый программный код

Характеристика масштабных факторов приведена в таблице 4.7.

Таблица 4.7 - Характеристика масштабных факторов

|  |  |
| --- | --- |
| Масштабный фактор Wi | Уровень фактора |
| Предсказуемость PREC | 3 |
| Гибкость разработки FLEX | 2 |
| Разрешение архитектуры риска RESL | 2 |
| Связанность группы TEAM | 1 |
| Зрелость процесса PMAT | 1 |
| Итого | 9 |

Коэффициент В на основании экспертных оценок будет иметь значение

Тип модели - распространённый, соответственно коэффициенты определены в размере: АТ=2,4; В=1,1.

Факторы затрат конструктивной модели стоимости обобщены в таблице 4.8

Таблица 4.8 - Факторы затрат конструктивной модели стоимости

|  |  |  |  |
| --- | --- | --- | --- |
| Название Mi-го фактора | Уровень фактора | Описание | Численное значение |
| 1 Требуемая надежность ПО – RELY | Низкий | Низкие, легко восстанавливаемые потери | 0,88 |
| 2 Размер базы данных – DATA (D - байты БД; P – LOC программного изделия) | Номинальный | 10≤D/P<100 | 1,00 |
| 3 Сложность модуля в зависимости от области применения - CPLX | Высокий | Высокая вложенность операторов с составными операторами. Однородная распределённая разработка | 1,09 |
| 4 Требуемая повторная используемость - RUSE | Номинальный | На уровне проекта | 1,00 |
| 5 Документирование требований жизненного цикла (ЖЦ) – DOCU | Номинальный | Оптимизированы к требованиям ЖЦ | 1,00 |
| 6 Ограничение времени выполнения платформы - TIME | Высокий | 70% | 1,11 |
| 7 Ограничение оперативной памяти платформы - STOP | Высокий | 70% | 1,06 |
| 8 Изменчивость платформы - PVOL | Низкий | Значительные изменения - каждые 12 месяцев, незначительные - каждый месяц | 0,87 |
| 9 Возможности аналитика - ACAP | Номинальный | 55% | 1,00 |
| 10 Возможности программиста - PCAP | Номинальный | 55% | 1,00 |
| 11 Опыт работы с приложениями - AEXP | Номинальный | 1 год | 1,00 |
| 12 Опыт работы с платформой - PEXP | Номинальный | 1 год | 1,00 |
| 13 Опыт работы с языком и утилитами - LTEX | Высокий | 3 года | 0,91 |
| 14 Использование программных утилит - TOOL | Высокий | Развитые утилиты ЖЦ, умеренная интеграция | 0,86 |
| 15 Требуемый график разработки - SCED | Номинальный | 100% | 1,00 |

Рассчитан поправочный множитель (МР) по факторам таблицы 4.8:

(4.14)

Определим трудоёмкость проектирования ПИ:

Реальный уровень качества программного изделия в процессе его эксплуатации оценивается количеством содержащихся в нём дефектов (ошибок). В целях соизмеримости программных изделий, разработанных на различных языках, плотность дефектов (дефектность) в таких случаях обычно рассчитывается на единицу размера программного кода “тысяча строк эквивалентного ассемблерного кода” KAELOC. В этом случае объём ПИ конкретного языка программирования, в нашем случае C#, в KLOC умножается на соответствующий коэффициент пересчёта КП­ (КП=2,5):

Качество разрабатываемого ПИ с позиции требований потребителя оценивается из условия, что распределение вероятностей строк кода размером в KAELOC, содержащих дефекты и принятых за случайные величины, подчиняются нормальному закону распределения. Значение сигмы показывает, как часто может возникнуть дефект. Чем больше сигм, тем менее вероятно возникновение дефектов, тем выше надёжность продукта, а потому выше степень удовлетворения требований потребителя.

Соотношение поля допуска с полем разброса (в “сигмах”) связывают с числом дефектов на единицу объёма ПИ размером KAELOC (Дi). В данном случае уровень качества - 5σ (Дi=0,233), а в проектируемом - 6σ (Дi+1=0,0034).

В соответствии с объёмом строк KAELOC в ПИ определён КВД=1,5, КНД=3,5, КДУ=0,75. Определим затраты на проектирование

Результаты расчётов элементов единовременных затрат по сравниваемым вариантам сводятся в таблицу 4.9.

Таблица 4.9 - Единовременные затраты по вариантам

|  |  |  |
| --- | --- | --- |
| Наименование элементов единовременных затрат | Величина по элементам, р. | |
| базовый | проектный |
| Стоимость комплекта оборудования с учётом необходимой офисной мебели |  |  |
| Стоимость запасов в оборотные средства | 144,00 | 72,00 |
| Стоимость потребной площади здания |  |  |
| Затраты на проектирование | - |  |
| Итого единовременных затрат | 21048.21 | 36255.32 |

Единовременные затраты на проектный вариант превышают базовый, но необходимо учитывать, что существующий вариант системы не автоматизирован, что значительно уменьшает её стоимость. Проектный вариант системы, не смотря на свои затраты на проектирование, достаточно перспективен и поможет повысить производительность труда разработчика или менеджера, за счёт уменьшения времени, затраченного на взаимодейсвтие внутри команды, что, в свою очередь, уменьшит время, которое будет отводиться на разработку проекта.

## 4.4 Расчёт годовых текущих издержек

Годовые текущие издержки в разрезе вариантов сравнения рассчитываются по следующим статьям:

, (4.15)

где ИЗП - годовые затраты на заработную плату разработчика или менеджера с начислениями, р.;

ИМ - годовые затраты на материалы за вычетом реализованных отходов, р.;

ИЭ - годовые затраты на силовую электроэнергию, р.;

ИРО - годовые затраты на ремонт и содержание оборудования, р.;

ИРЗ - годовые затраты на ремонт и содержание зданий, р.;

ИНР - годовые накладные расходы по управлению и обслуживанию производства, р.

Годовые затраты на заработную плату разработчика или менеджера с начислением по рабочим местам рассчитываются по рабочим местам рассчитываются по формуле:

(4.16)

где ЗТ - часовая тарифная ставка 1-го разряда, ЗТ=1,19 р.;

Ктi - тарифный коэффициент разряда по i-задаче, КТ11=2,65;

КПi - коэффициент премирования по i-задаче, КП=0,5;

КД­ - коэффициент, учитывающий дополнительную заработную плату, КД=0,1;

КСС - коэффициент, учитывающий отчисления на социальные нужды, КСС=0,35;

КН - коэффициент, учитывающий налоги на заработную плату, КН=0,1.

Определим годовые затраты на заработную плату разработчиков

Годовые затраты на материалы рассчитываются по формуле:

(4.17)

где Pmj - цена приобретения используемого j-го материала, р.;

РOj - цена реализуемых отходов, р.;

HMj - норма расхода j-х видов материала, шт. (кг);

HОj - норма реализуемых отходов, шт. (кг).

АРМ менеджера при подготовке отчётов предусматривает расход бумаги, тонера для заправки картриджа принтера, канцелярских товаров. Определён расход материалов по решаемой задаче:

1. Расчёт расхода бумаги формата А4 для ежедневной печати документов произведён исходя из количества бумаги на решение одной задачи и годовой задачи решений. На решение задачи требуется в среднем 2 листа бумаги формата А4 с учётом величины годовой программы Аг = 5865 в год потребуется 15552 листов бумаги.

Для базисного варианта потребность в бумаге на 30% больше по причине значительного количества операций выполняемых вручную (20218 листов).

С учётом количества листов бумаги в пачке, для базового варианта необходимо 19 пачек бумаги, для проектного - 41 пачка.

1. Расходные материалы по принтеру: заправочный комплект Pantum TN-420 H рассчитан на 3000 страниц. В соответствии с базовым вариантом необходима 6.74 заправки в год, для проектного – 5.184 заправки в год.
2. В базовом и проектном вариантах предусмотрены канцтовары (30% от стоимости бумаги и тонера).

Определены годовые затраты на материалы:

Годовые издержки на потребляемую электроэнергию, если оборудование работает в режиме полной занятости в течение рабочего дня, рассчитывается по формуле:

(4.18)

где Wi - потребляемая мощность i-го оборудования (таблица 7.8), кВт;

Fд - годовой действительный фонд работы единицы оборудования, ч.;

Рэ - цена (тариф) за 1 кВт·ч потреблённой энергии, Рэ = 0,29567р/кВт·ч.

Потребляемая мощность используемого оборудования представлена в таблице 4.10.

Таблица 4.10 - Потребляемая мощность оборудования

|  |  |
| --- | --- |
| Оборудование | Потребляемая мощность, кВт |
| Моноблок Тесла Директор / Q24B2 | Корпус Q24b2 | 0,040 |
| МФУ PANTUM M6500W | 0,45 |
| Итого | 0,49 |

Таблица 4.10 составлена по данным технической документации предложенного оборудования. Определим годовые издержки на потребляемую электроэнергию:

Годовые издержки на ремонт и содержание оборудования рассчитывают по формуле:

(4.19)

Определим годовые издержки на ремонт и содержание оборудования

Годовые издержки на ремонт и содержание зданий рассчитываются по формуле:

, (4.20)

где Нрз - норматив на ремонт и содержание здания, Нрз=2,5.

Определим годовые издержки

Годовые накладные расходы состоят из статей затрат на управление и обслуживание производства ИУ, ИОС, воду на бытовые нужды ИВБ, тепловой энергии на горячую воду ИТЭГВ, отопление ИТЭОТ, вентиляцию ИТЭВТ:

, (4.21)

Годовые расходы на управление и обслуживание производства определяются по формуле:

, (4.22)

где ККУ - коэффициент, учитывающий косвенные расходы по управлению (ККУ=0,25)

Для следующих статей затраты для базового и проектируемого вариантов равны. Годовые затраты электроэнергии на освещение рассчитываются по формуле:

(4.22)

где Ws - норма освещённости, Ws =0,03 кВт/м2;

S - площадь производственных и служебно-бытовых помещений, S=14м2;

FО - годовой действительный фонд освещения, Fо =2400 ч.

Годовые затраты воды на бытовые нужды рассчитываются по формуле:

, (4.23)

где РВБ - цена воды на бытовые нужды, РВБ =3,3749 р./м3;

НВБ - норма расхода воды на бытовые нужды за сутки на одного работника, НВБ = 0,025 м3

Чр - численность работников, Чр=2 чел.

41.02р

Годовые затраты тепловой энергии на горячую воду рассчитываются по формуле:

(4.24)

где Ртэ - цена (тариф) за теплоэнергию, Ртэ= 129,536р/Гкал;

qТХВ - удельная тепловая характеристика воды, qТХВ = 1 ккал/(м3·ч·℃);

VВГ - объём потребления горячей воды за 1 час, VВГ = 3 л;

tВГ, tВХ - температура горячей воды в системе tВГ=+65℃, холодной воды tВХ =+5℃;

FВГ - период теплоснабжения горячей водой,

(4.25)

Годовые затраты тепловой энергии на отопление рассчитываются по формуле:

*,* (4.26)

где qЗДТХ – удельная тепловая характеристика здания qЗДТХ =0,40, ккал/(м3 ч оС);

VЗД – объем помещения здания по наружному обмеру (VЗД = SH, где высота помещения Н= 3,5м), м3;

tЗДВН, tЗДН – температура воздуха внутри помещения и снаружи соответственно (tЗДВН = + 20, tЗДН = - 10), оС;

FОТ – отопительный период за год (FОТ = 4320 ч), ч.

Годовые затраты тепловой энергии на вентиляцию рассчитываются по формуле:

, (4.27)

где qТХВТ - удельная тепловая характеристика вентиляция здания, qТХВТ =0,15 ккал/(м3·ч·℃);

tВНВТ, tНВТ - температура воздуха вытяжного и снаружи tВТВН = + 20, tВТН = - 1,5), оС

FВТ - период работы вентиляционной системы, FВТ = 1400ч

КТЭП - коэффициент, учитывающий потери теплоэнергии, КТЭП = 1,18

Расчёт имеет вид:

Определены годовые накладные расходы по вариантам:

Результаты расчётов за год по статьям текущих издержек сведены в таблице 4.11.

Таблица 4.11 - Годовые текущие издержки по вариантам

|  |  |  |
| --- | --- | --- |
| Наименование статей издержек | Величина издержек, р. | |
| базовый | проектный |
| Затраты на заработную плату разработчика с начислениями |  |  |
| Затраты на материалы | 1274.84 | 991.42 |
| Затраты на силовую электроэнергию |  |  |
| Затраты на ремонт и содержание оборудования |  |  |
| Затраты на ремонт и содержание зданий |  | .12 |
| Накладные расходы |  |  |
| Итого годовых текущих издержек | 39349.75 | 20226.17 |

Годовые текущие издержки снизились на 19123.17 по сравнению с базовым вариантом. Снижение текущих издержек при использовании информационной системы оптимизации взаимодействия произошло за счёт уменьшения трудоёмкости решения задачи, соответствующего увеличения производительности труда разработчика и менеджера в контексте взаимодействия внутри команды.

## 4.5 Расчёт показателей экономической эффективности

Для технических решений в области совершенствования информационной системы, имеющих внутрипроизводственную значимость, годовой экономический эффект определятся по следующей формуле:

(4.28)

где *ЗБГ, ЗПГ* – годовые приведенные затраты по базовому и проектному варианту.

Величина приведённых затрат определяется по формуле:

(4.29)

где Ен - нормативный коэффициент эффективности, Ен=0,1;

К, Кi - единовременные затраты (таблица 4.7) суммарные и по элементам, р.;

pi - норма реновации единовременных затрат, которая рассчитывается как обратная величина срока службы tСЛi по i-элементам (оборотных элементов и затрат на проектирование tСЛ=4 года) с учётом морального износа:

, (4.30)

И - годовые издержки (таблица 7.9), р.

В таблицу 4.12 внесены нормы реновации единовременных затрат по элементам в соответствии с выбранной величиной срока службы по i-м элементам.

Таблица 4.12 - Норма реновации элементов единовременных затрат

|  |  |  |
| --- | --- | --- |
| Наименование элементов единовременных затрат | Срок службы tСЛi по i-м элементам | Норма реновации |
| Стоимость комплекта машин и оборудования с учётом необходимой офисной мебели | 5 | 0,164 |
| Стоимость запасов в оборотные средства | 4 | 0,215 |
| Затраты на проектирование | 4 | 0,215 |
| Стоимость потребной площади здания | 50 | 0,000859 |

Так как единовременные затраты по проектному варианту превышают базовый, рассчитаем срок окупаемости:

(4.31)

Период окупаемости меньше нормативного, следовательно, подтверждается целесообразность проектного варианта оцениваемых технических решений.

## 4.6 Организация внедрения системы

Дипломный проект на тему “АСОИ “Автоматизация процесса взаимодействия команды разработчиков программного обеспечения”. Разработка серверной части” выполнен с целью автоматизации процесса взаимодействия команды разработки. Разработанная система с автоматизированным местом разработчика или менеджера позволит повысить производительность труда специалиста и избежать ошибок при разработке программного обеспечения.

График внедрения АСОИ приведён в таблице 4.13.

Таблица 4.13 – План-график внедрения разработанного программного продукта

|  |  |  |
| --- | --- | --- |
| Наименование операции | Исполнитель | Время, дни |
| 1 Установка необходимого ПО | Системный администратор или разработчик | 1 |
| 2 Настройка | Разработчик | 2 |
| 3 Тестирование | Разработчик и пользователь | 1 |
| 4 Обучение пользователя | Разработчик | 1 |
| Итого | - | 5 |

## 4.7 Заключение по разделу

Основные технико-экономические показатели дипломного проекта, которые определяют сравнительную экономическую эффективность принятых технических решений, сведены в таблицу 4.14.

Таблица 4.14 - Технико-экономические показатели по сравнительным вариантам

|  |  |  |
| --- | --- | --- |
| Наименование показателей | Величина по вариантам | |
| базовый | проектный |
| 1 | 2 | 3 |
| 1 Годовое количество решаемых задач |  |  |
| 1.1 Количество решаемых технический заданий | 4725 | 5865 |
| 2 Норма решения задачи, мин |  |  |
| 2.1 Решение одного задания | 37.51 | 18.9 |
| 3 Уровень качества программного изделия iσ | 5 | 6 |
| 4 Потребляемая мощность вычислительных средств, кВт | 0,41 | 0,41 |
|  |  |  |

Продолжение таблицы 4.14

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 5 Единовременные затраты, р. | 21048.21 | 36255.32 |
| 6 Годовые текущие издержки, р. | 39349.75 | 20226.17 |
| 7 Годовые приведённые затраты, р. |  |  |
| 8 Срок окупаемости | - | 0.7 |
| 9 Продолжительность освоения, дней | - | 5 |
| 10 Продолжительность использования, лет | - | Не менее 10 |

Анализ технико-экономических показателей позволил выявить значительную экономию годовых текущих затрат на 12695.23 рублей при отсутствии роста единовременных затрат на проектирование программного изделия. Окупаемость проекта составляет 0.7 года. Внедрение данной автоматизированной системы обеспечит повышение эффективности работы менеджеров и разработчиков, уменьшение рисков, которые присутствуют в классическом подходе к взаимодействию внутри команд. Таким образом, в ходе выполнения данного раздела дипломного проектирования, была обоснована экономическая целесообразность внедрения изделия относительно действующего в настоящее время подхода к разработке и взаимодействию команд разработки программного обеспечения.

**5 Охрана труда**

Преддипломная практика проходила в ИОО «ЭПАМ Системз» в качестве инженера-программиста.

## 5.1 Система управления охраной труда на предприятии

## 5.1.1 Политика в области охраны труда

Основной вид работ для работников IT-компании - работа с так называемым офисным оборудованием. Требования по охране труда, которые должны соблюдаться работниками при использовании в работе офисного оборудования, установлены в Типовой инструкции N 25. В рамках этого документа под офисным оборудованием понимаются персональные электронные вычислительные машины (ПЭВМ), копировально-множительная техника, сканирующие устройства, которые, анализируя какой-либо объект (изображение, текст), создают цифровую копию его изображения ([ч. 2 п. 1](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365ABE1E7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25).

К выполнению работ с использованием офисного оборудования допускаются работники, прошедшие в установленном законодательством порядке инструктажи по охране труда (далее - инструктажи), в соответствии с требованиями [главы 3](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097729E80537333F930E6A93D93935B46895BB96036264A9E0E7BEDAF077561E283B1AC54289E6E9A5M2S) Инструкции N 175 ([п. 2](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365ABEFE7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25).

Для целей проведения инструктажей наниматель в соответствии с предписаниями [Инструкции](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B8709409772AEE0936333F930E6A93D93935B46895BB96036367A0E3E7BEDAF077561E283B1AC54289E6E9A5M2S) N 176 должен разработать инструкцию по охране труда (ИОТ) при использовании в работе офисного оборудования, руководствуясь наряду с прочим Типовой [инструкцией](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365ABE4E7BEDAF077561E283B1AC54289E6E9A5M2S) N 25.

В некоторых случаях можно организовывать инструктажи, основываясь только на положениях Типовой [инструкции](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365ABE4E7BEDAF077561E283B1AC54289E6E9A5M2S) N 25, без утверждения собственного ЛПА. Такое допускается тогда, когда отсутствует специфика, которую необходимо отразить в ЛПА организации, и требования Типовой [инструкции](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365ABE4E7BEDAF077561E283B1AC54289E6E9A5M2S) N 25 являются достаточными и соответствующими выполняемым в данной организации работам ([ч. 2 п. 4](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B8709409772AEE0936333F930E6A93D93935B46895BB96036367A0EFE7BEDAF077561E283B1AC54289E6E9A5M2S) Инструкции N 176).

В процессе использования в работе офисного оборудования на работающих могут воздействовать следующие вредные и (или) опасные производственные факторы ([ч. 1](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365AAE6E7BEDAF077561E283B1AC54289E6E9A5M2S), [2 п. 3](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365ADE7E7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25):

— повышенный уровень электромагнитных излучений;

— повышенный уровень ионизирующих излучений;

— повышенный уровень статического электричества;

— повышенная напряженность электростатического поля;

— повышенная или пониженная ионизация воздуха;

— повышенная яркость света;

— прямая и отраженная блесткость;

— повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека;

— статические перегрузки костно-мышечного аппарата и динамические локальные перегрузки мышц кистей рук;

— перенапряжение зрительного анализатора;

— другие - в зависимости от условий труда, в которых применяется офисное оборудование, и характера работы.

При работе с офисным оборудованием работники должны соблюдать определенные требования по охране труда. Это, как правило, требования перед началом работы, при ее выполнении и по окончании, а также при аварийных ситуациях. Рассмотрим ниже некоторые из них.

При использовании в работе офисного оборудования работникам необходимо ([п. 5](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365ACE4E7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25):

— пользоваться исправными выключателями, розетками, штепсельными вилками и другой электроарматурой;

— знать и соблюдать требования эксплуатационных документов организаций - изготовителей используемого офисного оборудования.

Перед началом работы с использованием офисного оборудования работнику следует ([п. 8](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365AFE7E7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25):

— проверить целостность питающих и соединительных кабелей, разъемов и штепсельных соединений, защитного заземления (зануления);

— проверить оснащенность рабочего места (места для выполнения работы (оказания услуги)) и убедиться в устойчивости положения офисного оборудования на рабочей поверхности;

— проверить отсутствие видимых повреждений офисного оборудования;

— включить офисное оборудование в электрическую сеть;

— расположить клавиатуру ПЭВМ на поверхности рабочего стола на расстоянии 100 - 300 мм от края, обращенного к работающему, или на специальной регулируемой по высоте поверхности, отделенной от основной столешницы;

— разместить экран видеомонитора на расстоянии 600 - 700 мм от глаз, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов (далее - оптимальное расстояние от экрана видеомонитора до глаз) так, чтобы уровень глаз при вертикально расположенном экране видеомонитора приходился на центр или 2/3 высоты экрана.

Перед началом работы с использованием офисного оборудования не допускается ([п. 9](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365AEE2E7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25):

— использовать для подключения офисного оборудования розетки, удлинители, не оснащенные заземляющим контактом (шиной);

— устанавливать системный блок в закрытых нишах мебели, непосредственно на полу;

— приступать к работе с ПЭВМ:

— при мелькании изображения на экране видеомонитора;

— обнаружении неисправности офисного оборудования, кабелей или проводов, разъемов, штепсельных соединений;

— отсутствии или неисправности защитного заземления (зануления) офисного оборудования.

При выполнении работы с использованием офисного оборудования работник должен ([п. 11](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365A1E0E7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25):

- использовать офисное оборудование исключительно по назначению;

- держать открытыми его вентиляционные отверстия;

- соблюдать оптимальное расстояние от экрана видеомонитора до глаз;

- поддерживать рациональную рабочую позу и оптимальное размещение на рабочей поверхности используемого офисного оборудования с учетом его количества и конструктивных особенностей, характера выполняемой работы.

При выполнении работы с использованием офисного оборудования не допускается ([п. 14](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036365A0E0E7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25):

- работать мокрыми руками и способствовать попаданию влаги на поверхность офисного оборудования;

- прикасаться к панелям с разъемами офисного оборудования, разъемам питающих и соединительных кабелей, экрану видеомонитора при включенном питании;

- ставить на кабель предметы, натягивать, перекручивать и перегибать его;

- переключать и отключать питание во время выполнения активной задачи, а также часто переключать питание;

- самостоятельно вскрывать и ремонтировать офисное оборудование.

По окончании работы с использованием офисного оборудования работнику следует ([п. 15](consultantplus://offline/ref=4C10C8852D6375549E84BB3AD98B2E1BBBF5B87094097928E80537333F930E6A93D93935B46895BB96036364A9EFE7BEDAF077561E283B1AC54289E6E9A5M2S) Типовой инструкции N 25):

- корректно закрыть все активные задачи;

- отключить офисное оборудование от электрической сети;

## 5.1.2 Ответственность

Таблица 5.1 — Матрица распределения ответственности

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование функции | Директор офиса | Административный менеджер | Специалист по охране труда |
| 1 | 2 | 3 | 4 |
| Организация деятельности по обеспечению охраны труда | Р, К | О, К | И, О, У |
|  |  |  |  |

Продолжение таблицы 5.1

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Подготовка документов по вопросам охраны труда | Р, К | У | И, О, У |
| Проведение инструктажей по охране труда | Р, К | У, К | И |
| Разработка предложений и мероприятий по повышению уровня охраны труда | Р, К | О, У | И |
| Проведение проверки по состоянию условий охраны труда на рабочих местах | Р, К | У | И |
| Проведение и проверка знаний работников | Р, К | И, К | О, И |
| Примечание: Р — руководство; И — информирование; К — контроль; У — участие; О — ответственность | | | |

## 5.2 Наблюдение за состоянием условий труда по системе Элмери

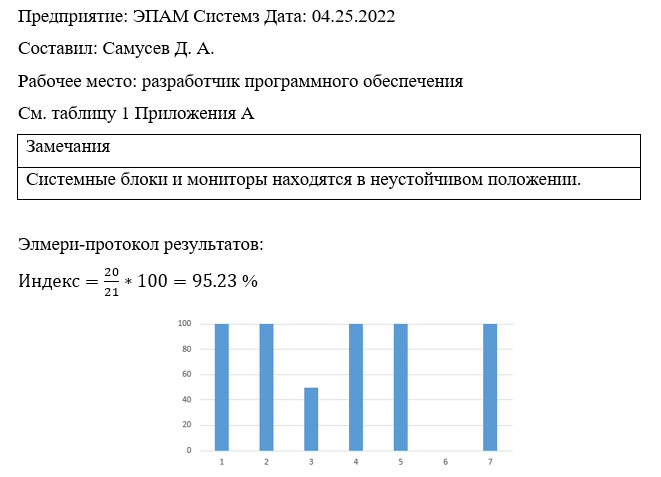


Рисунок 5.1 — Рабочее место разработчика программного обеспечения

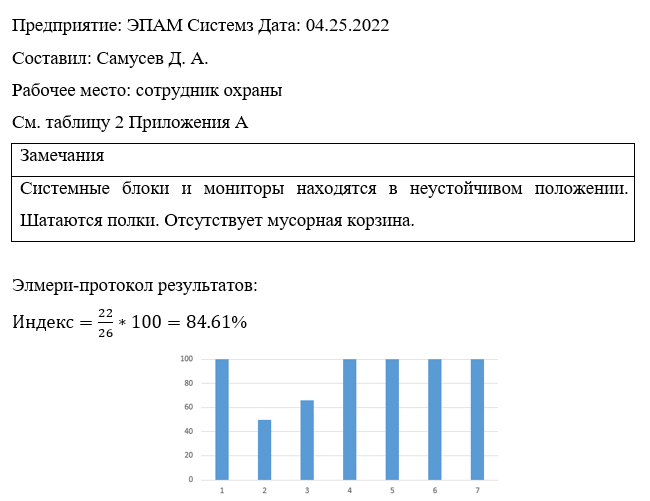


Рисунок 5.2 — Рабочее место сотрудника охраны

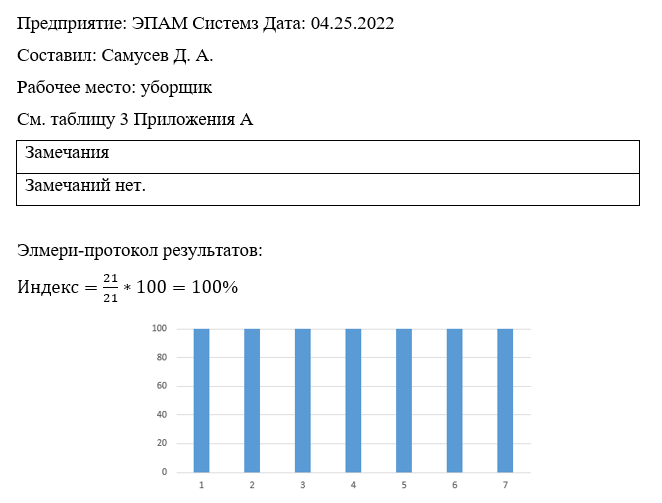


Рисунок 5.3 — Рабочее место уборщика

На основании полученных протоколов можно сделать вывод, что предприятие удовлетворяет практически всем требованиям условий труда. Были приняты всевозможные меры по обеспечению защиты сотрудников от рисков, связанных с работой с офисным оборудованием. Вентиляция помещений происходит на должном уровне. Зимой и летом исправно работают обогреватели и кондиционеры, что благотворно сказывается на макроклимате офиса. Системные блоки и сервер предприятия не нарушают шумовых норм, т.к. на предприятии используется достаточно совершенное оборудование, способное сохранять работоспособность длительное время без перегрева и лишних шумов. Сервер предприятия расположен в отдельном помещении, что изолирует дополнительные шумы. Так же на предприятии используются совершенные мониторы, которые оказывают минимальное излучение и воздействие на глаза. Учитываются такие факторы как, наклон монитора, расстояние до глаз, разрешение, яркость, матрица, подсветка, частота кадров. Все вышеперечисленные параметры сотрудник может регулировать для максимизации комфорта работы за компьютером.

Возможностями по увеличению безопасности работников на предприятии ИОО “Эпам Системз” являются:

- повышение устойчивости офисного оборудования, чтобы исключить возможность случайных поломок и травм.

- дополнительное закрепление держащих поверхностей таких как: полки, столы, тумбы.

- контроль за обязательным перерывом в течении 10 минут каждые 2 часа, для предоставления возможности на отдых для организма.

**6 Энергосбережение и ресурсосбережение при внедрении новых технологий**

В наши дни энергосбережение является первой в списке важных задач. Этот факт привязан к острой нехватке большинства ресурсов, увеличивающейся цене их добычи и приобретения и, конечно, к вопросам экологии Земли.

Экономия энергии – есть эффективное использование энергетическо-ресурсов во время использовании инновационных систем, которые реализованы в технологически, экономически выгодны, не увеличивают риск нанесения вреда экологии. Данная формулировка была дана на Международной энергетической конференции (МИРЭК).

Энергосбережение в каждой сфере является по сути мероприятиями, направленными на снижение лишних затрат энергии. Подробные исследования энергозатрат в производственных и потребительских сферах указывает, что 90% расходов приходится на энергопотребление, в то время как расход на энергопередачу составляет всего 8-10%. Именно потому большинство усилий по энегросбережению сосредоточены как раз в сферах потребления энергии.

Главная роль по повышению эффективности потребления энергии приурочено современным технологиям энергосбережения. Технология энергосбережения – свежий или модернизированный процесс, который отличается более эффективным коэффициентом полезного потребления ТЭР (топливно-энергетический ресурс).

Будь то предприятие, или частное лицо бытового уровня – все нуждаются в применении технологий по энергосбережению. Этот шаг продвинет нас в разрешении большинства экологических вопросов – изменение климата окружающей среды, загрязнение воздуха выбросами заводов, критическая нехватка природных ресурсов. Как правило организации используют технологии, дающие существенный эффект энергосбережения. К таким относят: сушка, пар, теплообменники, альтернативные источники энергии, тригенерация, сжатый воздух, когенерация и другие.

Энергосберегающий режим весьма актуален в использовании устройств, которые некоторое время могут работать в режиме пониженной нагрузки – насосы, моторы, системы охлаждения. Есть большое множество приспособлений, способных сократить потери во время работы электрооборудования, например – конденсаторы, частотно регулируемые электроприводы. Такие приводы обладают механизмами оптимизации потребления энергии: гибкое регулирование частоты оборотов, которая зависит от действительной нагрузки. Этот механизм экономит от 30-45% расходуемой энергии. Более того, зачастую даже нет необходимости в замене двигателя, что может быть очень полезным при совершенствовании рабочих производственных предприятий.

Учёные разработали устройство, которое позволяет захватывать тепло, уходящее через трубу при сжигании газа, и использовать его для производства дополнительной энергии, способной обеспечить питанием высотные дома. Технологии энергосбережения имеют комплексное влияние на строительство. Под этим влиянием подразумевают утепление, стекольные пакеты, устройства обогрева и кондиционирования. Самая развитая и популярная технология энергосбережения с огромным потенциалом для повышения эффективности с области постройки жилищ – котельные установки. Эти технологии могут значительно сократить использование энергетических носителей, уменьшить расходы на техническое обслуживание. Более того, в некоторых случаях можно заметить повышение КПД. Так же, замена котельных установок зачастую позволяет организациям переходить на чистое топливо (газ, древесные гранулы), а не использовать дорогой уголь или мазут.

Более того, большей экономии можно достичь, если заменить отдельно стоящие тепловые пункты поместить в помещении, обеспеченном инновационными бесшумными моторами, которые так же и достаточно компактны.

Для обеспечения здания вентиляционной системой используется система рекуперации тепла выработанного воздуха и переменной производительности приточно-вытяжных агрегатов. Эти системы предоставляют возможность не использовать впустую тепло, полученное человеком из осветительных приборов, офисного инвентаря. Таким образом сокращается потребление тепла от источника извне. Конструкторы ИТ-оборудований обращают отдельное внимание энергоэффективности системам, так как данный параметр имеет значительное влияние на цену использования. Именно он обычно разрешает выбор в пользу того или иного устройства. Спрос на технологии энергосбережения объясняется повышением стоимости электричества или недостатка мощности, которые подводятся на предприятия.

В свою же очередь, организации, производящие серверные платформы так же вынуждены решать тяжёлую задачу. А именно обеспечение совершенствование отделов обработки данных, сталкивающихся с ограниченными площадями и мощностями. Им приходится учитывать, помимо расходов финансов, возникающие вопросы, из-за недостатка подведённых мощностей и требований избавления от избыточного тепла. А это сказывается на выборе как раз в пользу энергоэффективных механизмов.

У большинства ЦОД пределы энергопотребления и занимаемых площадей почти иссякли. Одновременно с этим необходимо и дальше расширять производительности систем. Для преодоления этих проблем необходимы по-настоящему энергоэффективные сервера и механизмы хранения информации, которые действительно нужны там, где наблюдается повышенная плотность размещения оборудования. Энергоэффективность оборудований на предприятии действительно эффективна во время разработки крупных решений. Она существенно уменьшает локальный нагрев и, в итоге, снижает показатель шумов, так как воздушный поток, охлаждающий сервера, значительно ослабевает. Энергоэффективность уже на протяжении долгих лет становится мировой темой во всех областях технологического развития и информационных технологий. В наши дни всё больше производителей обращают внимание на данный фактор во время конструирования собственных систем.

**Заключение**

В ходе выполнения технического задания спроектирована и реализована информационная система автоматизации взаимодействия внутри команды разработчиков. Созданная ИС позволяет посредством интерфейса получать оперативный и удобный доступ к данным.

Внедрение разработанной информационной системы даст возможность снизить затраты на маркетинговые расходы, повысить надежность и долговечность хранения данных.

Разработка показала свою экономическую целесообразность, обеспечив годовой экономический эффект в размере 12695.23 рублей и срок окупаемости в 0,7 года.

Разработанная ИС позволяет оперативно формировать технические задачи и выводить их на печать. При использовании ИС значительно сократилось время работы с задачами и текущими проектами. Реализация ИС представлена в виде программы на машинном носителе (оригинал, дубликаты), эксплуатационной программной документации и рабочей документации программного обеспечения.

Список использованных источников

1. **Кузнецов, И. Н**. Охрана труда: учебное пособие / И. Н. Кузнецов. - Минск: Четыре четверти, 2010. – 175 с.
2. **Ларман, К**. Применение UML 2.0 и шаблонов проектирования / К. Ларман. – СПб.: Питер, 2016. – 256 с.
3. **Михнюк, Т. Ф**. Охрана труда: учеб. пособие для вузов / Т. Ф. Михнюк. – Минск: Минфин, 2009. – 345 с.
4. Экономика и организация производства в дипломных проектах: Учеб. пособие для машиностроительных вузов / К.М. Великанов, Э.Г. Васильева, В.Д. Власов и др.; Под общ. ред. К.М. Великанова. – 4-е изд. Перераб. и доп. – Л.: Машиностроение, 1986.
5. Энергоэффективность и энергосбережение [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Энергоэффективность. Дата доступа: 01.03.2022.
6. **Архипова З.В.**, Пархомов В.А. Информационные технологии в экономике. Учебное пособие. – Иркутск, Издательство БГУЭП, 2003. – 230 с.
7. React.js документация [Электронный ресурс]. – Режим доступа: https://ru.react.js.org. – Дата доступа: 10.04.2022.
8. Справочник React [Электронный ресурс]. – Режим доступа: https://reactdev.ru – Дата доступа: 13.04.2022.
9. **Фаулер, М**. Рефакторинг кода на JavaScript: улучшение проекта существующего кода, 2-е изд. — М.: [«Диалектика»](https://ru.wikipedia.org/w/index.php?title=%D0%94%D0%B8%D0%B0%D0%BB%D0%B5%D0%BA%D1%82%D0%B8%D0%BA%D0%B0_(%D0%B8%D0%B7%D0%B4%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D1%81%D1%82%D0%B2%D0%BE)&action=edit&redlink=1), 2019. — 464 с.
10. **Дакетт, Д**. Javascript и jQuery. Интерактивная веб-разработка. — М., 2017. — 640 с.
11. **Резиг, Д**. Секреты Javascript ниндзя 2-е изд.. — М.: [«Диалектика»](https://ru.wikipedia.org/w/index.php?title=%D0%94%D0%B8%D0%B0%D0%BB%D0%B5%D0%BA%D1%82%D0%B8%D0%BA%D0%B0_(%D0%B8%D0%B7%D0%B4%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D1%81%D1%82%D0%B2%D0%BE)&action=edit&redlink=1), 2017. — 544 с.
12. **Никсон, Р**. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 4-е изд. — СПб.. — М.: [«Диалектика»](https://ru.wikipedia.org/w/index.php?title=%D0%94%D0%B8%D0%B0%D0%BB%D0%B5%D0%BA%D1%82%D0%B8%D0%BA%D0%B0_(%D0%B8%D0%B7%D0%B4%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D1%81%D1%82%D0%B2%D0%BE)&action=edit&redlink=1), 2016. — 768 с.
13. **Флэваrан, Д**. JavaScrirt: карманный справочник, 3-е изд. — М., 2013. — 320 с.
14. Чистый код C# [Электронный ресурс]. – Режим доступа: <https://qna.habr.com/q/708315> – Дата доступа: 07.05.2022.