

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук

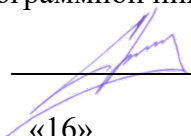
Департамент программной инженерии


СОГЛАСОВАНО

УТВЕРЖДАЮ

Научный руководитель,
доцент департамента
программной инженерии, канд. пед. наук

Академический руководитель
образовательной программы
«Программная инженерия»


_____ С. А. Виденин
«16» _____ мая 2025 г.


_____ Н. А. Павлов
«16» _____ мая 2025 г.

ИНФОРМАЦИОННО-АНАЛИТИЧЕСКАЯ СИСТЕМА УПРАВЛЕНИЯ
ИНЦИДЕНТАМИ В ОБЛАСТИ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ


Техническое задание
ЛИСТ УТВЕРЖДЕНИЯ
RU.17701729.05.05-01 ТЗ 01-1-ЛУ

Исполнители:

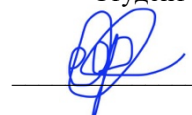
Студент группы БПИ-216

 / А.И. Шарипов /

Студент группы БПИ-216

 / С. А. Пономарев /

Студент группы БПИ-214

 / Е. К. Фортов /

«16» мая 2025

Москва 2025

Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

УТВЕРЖДЕН

RU.17701729.05.05-01 ТЗ 01-1-ЛУ

**ИНФОРМАЦИОННО-АНАЛИТИЧЕСКАЯ СИСТЕМА УПРАВЛЕНИЯ
ИНЦИДЕНТАМИ В ОБЛАСТИ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ**

Техническое задание

RU.17701729.05.05-01 ТЗ 01-1

Листов 27

Инв. № полл.	Полп. и дата	Взам. Инв. №	Инв. № лубл.	Полп. и дата

АННОТАЦИЯ

Техническое задание – это основной документ, оговаривающий набор требований и порядок создания программного продукта, в соответствии с которым производится разработка программы, ее тестирование и приемка.

Настоящее Техническое задание на разработку «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности» содержит следующие разделы: «Глоссарий», «Введение», «Основание для разработки», «Назначение разработки», «Требования к программе», «Требования к программным документам», «Техникоэкономические показатели», «Стадии и этапы разработки», «Порядок контроля и приемки» и приложения [7].

В разделе «Глоссарий» содержатся определения терминов и понятий, используемых в настоящем Техническом задании.

В разделе «Введение» указано наименование и краткая характеристика области применения приложения.

В разделе «Основания для разработки» указан документ на основании, которого ведется разработка и наименование темы разработки.

В разделе «Назначение разработки» указано функциональное и эксплуатационное назначение программного продукта.

Раздел «Требования к программе» содержит основные требования к функциональным характеристикам, к надежности, к условиям эксплуатации, к составу и параметрам технических средств, к информационной и программной совместимости, к маркировке и упаковке, к транспортировке и хранению, а также специальные требования.

Раздел «Требования к программным документам» содержит предварительный состав программной документации и специальные требования к ней.

Раздел «Технико-экономические показатели» содержит ориентировочную экономическую эффективность, предполагаемую годовую потребность, экономические преимущества разработки приложения.

Раздел «Стадии и этапы разработки» содержит стадии разработки, этапы и содержание работ.

В разделе «Порядок контроля и приемки» указаны общие требования к приемке работы.

Настоящий документ разработан в соответствии с требованиями:

- 1) ГОСТ 19.101-77 Виды программ и программных документов [1];
- 2) ГОСТ 19.102-77 Стадии разработки [2];
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов [3];
- 4) ГОСТ 19.104-78 Основные надписи [4];
- 5) ГОСТ 19.105-78 Общие требования к программным документам [5];
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];
- 7) ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению [7].

Изменения к данному Техническому заданию оформляются согласно ГОСТ 19.60378 [8], ГОСТ 19.604-78 [9].

ГЛОССАРИЙ.....	4
1. ВВЕДЕНИЕ.....	7
1.1. Наименование программы.....	7
1.2. Краткая характеристика области применения.....	7
2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ.....	9
2.1. Документы, на основании которых ведется разработка.....	9
2.2. Наименования темы разработки.....	9
3. НАЗНАЧЕНИЕ РАЗРАБОТКИ.....	10
3.1. Функциональное назначение.....	10
3.2. Эксплуатационное назначение.....	10
4. ТРЕБОВАНИЯ К ПРОГРАММЕ.....	13
4.1. Требования к функциональным характеристикам.....	13
4.1.2. Организация входных данных.....	15
4.1.3. Организация выходных данных.....	16
4.2. Требования к временным характеристикам.....	16
4.3. Требования к интерфейсу.....	16
4.4. Требования к надежности.....	16
4.5. Условия эксплуатации.....	16
4.6. Требования к составу и параметрам технических средств.....	16
4.7. Требования к информационной и программной совместимости.....	17
4.8. Требования к маркировке и упаковке.....	17
4.9. Требования к транспортировке и хранению.....	17
5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ.....	18
5.1. Предварительный состав программной документации.....	18
5.2. Специальные требования к программной документации.....	19
6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ.....	20
6.1. Ориентировочная экономическая ценность.....	20
6.2. Предполагаемая потребность.....	20
6.3. Экономические преимущества разработки по сравнению с отечественными и.....	21
зарубежными аналогами.....	21
6.3.1 Российские аналоги.....	21
6.3.2 Зарубежные аналоги.....	21
7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ.....	22
8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ.....	24
8.1. Виды испытаний.....	24
8.2. Общие требования к приемке работы.....	24
СПИСОК ИСТОЧНИКОВ.....	25
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....	26

ГЛОССАРИЙ

1. Аргус – название описываемой системы
2. Авторизация - процесс проверки прав доступа пользователя или системы к определённой функциональности. Осуществляется в первую очередь через проверку JWT-токена в прокси-сервисе.
3. Аутентификация - процесс подтверждения подлинности пользователя, который пытается войти в систему (проверка логина/пароля или иного фактора). В системе реализуется через прокси-сервис с выдачей JWT-токенов.
4. Бизнес-процесс - логическая последовательность действий, определяющая, как следует обрабатывать конкретный тип инцидентов. В системе каждый бизнес-процесс связан с графом статусов и набором тикетов, возникающих при срабатывании датчика или создании инцидента вручную.
5. Датчик - физическое или программное устройство, которое фиксирует конкретное событие (например, задымление, протечку воды) и отправляет сигнал в систему. При срабатывании датчика система автоматически запускает соответствующий бизнес-процесс.
6. Дежурство - механизм назначения ответственных лиц на временные интервалы в системе. Позволяет определить, кто именно из группы пользователей активно отвечает за инциденты в конкретный момент.
7. Документирование - фиксация всех действий, статусов и уведомлений, связанных с инцидентом (тикетом). Обеспечивает прозрачность истории решения и simplifies последующий анализ.
8. Жизненный цикл инцидента - последовательность стадий, которую проходит инцидент от момента возникновения до полного решения/закрытия. В системе этот цикл определяется статусами и графами статусов.
9. Инцидент - любое нештатное событие — авария, поломка, пожар, утечка, которое требует оперативного реагирования. В контексте системы инциденты оформляются в виде тикетов и обрабатываются по бизнес-процессам.
10. ITSM (IT Service Management) - методология, связанная с управлением IT-сервисами и инцидентами внутри ИТ-инфраструктуры. Примеры решений: ServiceNow, Jira ServiceManagement. Система имеет некоторые параллели с ITSM, но сфокусирована также на физических датчиках и производственных инцидентах.

11. JWT (JSON Web Token) - стандарт открытого формата токенов, используемый для передачи данных аутентификации и авторизации между клиентом и сервером. В системе применяется для проверки прав пользователя при каждом запросе (через прокси-сервис).
12. Микросервисная архитектура (Microservices) - стиль проектирования ПО, при котором система состоит из набора небольших, независимых сервисов (микросервисов). В «Аргус» выделены отдельные сервисы для работы с датчиками, статусами, тикетами/бизнес-процессами и т.д.
13. Прокси-сервис (Proxy Service) - центральная точка входа в «Аргус», проверяющая JWT-токен при каждом запросе и маршрутизирующая его к нужному микросервису. Отвечает за безопасность, аутентификацию и авторизацию.
14. Приоритет (Priority) - уровень важности задачи или инцидента, определяющий порядок и срочность его обработки (например High, Medium, Low). В «Аргус» задаётся при создании тикета и влияет на порядок решения.
15. Прозрачность решения (Transparency) - свойство системы, при котором все действия, статусы и взаимодействия по инцидентах доступны для отслеживания в любой момент. В «Аргус» достигается за счёт сохранения истории тикетов, статусов, комментариев и уведомлений.
16. SCADA-система (Supervisory Control and Data Acquisition) - класс систем, предназначенных для сбора данных и управления автоматикой на оборудовании (к примеру, Siemens WinCC). По сравнению с «Аргус» SCADA-системы ориентированы на физический контроль и считывание параметров, но не обеспечивают гибкие бизнес-процессы с дежурствами и тикетами.
17. SLA (Service Level Agreement) - соглашение об уровне обслуживания, регламентирующее допустимое время реакции и решения инцидентов. В «Аргус» для каждого статуса можно задать время на выполнение (escalationSLA), при превышении которого включается механизм уведомлений всей дежурной группы (фоллбэк).
18. Статус - конкретное состояние инцидента (тикета), например: «Новый», «В работе», «Вызов службы», «Закрыт». Набор статусов формирует граф статусов, по которому «движется» тикет.
19. Тикет - основная рабочая сущность для фиксации и ведения инцидента (содержит описание, приоритет, исполнитель, сроки, историю комментариев). Создаётся в системе

автоматически по сигналу датчика или вручную пользователем и проходит через все необходимые статусы в соответствии с бизнес-процессом.

20. Уведомление - механизм доставки оповещений (по SMS, e-mail) ответственным лицам при возникновении или переходе тикетов в новый статус. Может настраиваться с разными интервалами и каналами в зависимости от статуса и роли пользователя.

21. Фоллбэк - механизм автоматической эскалации задачи, если она не решается в установленный срок. В «Аргус» при срабатывании фоллбэка уведомляется вся дежурная группа или назначенный список ответственных.

22. Граф статусов - модель в виде ориентированного графа, где вершины — это статусы, а рёбра — возможные переходы между ними. В «Аргус» определяет, по каким стадиям проходит тикет и кто будет ответственным на каждой стадии.

23. Эскалация - процесс передачи инцидента на другой уровень ответственности или более высокому руководству, если решение затягивается или выходит за рамки SLA. В системе «Аргус» может срабатывать автоматически при нарушении сроков.

1. ВВЕДЕНИЕ

1.1. Наименование программы

Наименование программы – «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности».

Наименование программы на английском языке – «Information and Analytical Incident Management System in the Field of Security».

1.2. Краткая характеристика области применения

Информационно-аналитическая система управления инцидентами в области обеспечения безопасности представляет собой специализированное программное решение, предназначенное для координации, мониторинга и автоматизации реагирования на инциденты безопасности. Область её применения охватывает объекты инфраструктуры, где применяются датчики безопасности (например, пожарные, водные, охранные датчики), а также компании и организации, ответственные за эффективное управление рисками и угрозами.

Ключевое функциональное предназначение системы заключается в автоматизации обработки тревожных событий, управлении бизнес-процессами, связанных с инцидентами, и координации действий всех участников процесса реагирования. Система сочетает элементы трекера задач (по аналогии с Jira) и платформы для интеграции данных от датчиков, обеспечивая реализацию заранее настроенных протоколов реагирования без задержек.

Основные функции:

- Автоматизация событий: Декодировка сигналов от датчиков с автоматическим запуском предопределённых сценариев реагирования.
- Координация процессов безопасности: Назначение исполнителей, управление тикетами, распределение приоритетов и статусов.
- Регуляция доступа: Контроль авторизации пользователей и обеспечение безопасности работы системы.
- Уведомления и взаимодействие: Обеспечение прозрачного взаимодействия между участниками инцидентов с помощью автоматических уведомлений, комментариев и текущих статусов задач.

Сфера применения системы включает промышленные объекты, коммерческие здания, серверные и складские помещения, транспортную и критически важную инфраструктуру, где требуется управление множеством событий безопасности.

2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

2.1. Документы, на основании которых ведется разработка

Основанием для разработки является учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем программы тема ВКР.

2.2. Наименования темы разработки

Наименование темы разработки – ”Информационно-аналитическая система управления инцидентами в области обеспечения безопасности””.

Разработка программы ведется в рамках выпускной квалификационной работы в соответствии с учебным планом подготовки бакалавров (Национальный исследовательский университет ”Высшая школа экономики”, факультет компьютерных наук, департамент программной инженерии), по направлению 09.03.04 ”Программная инженерия”.

3. НАЗНАЧЕНИЕ РАЗРАБОТКИ

3.1. Функциональное назначение

Информационно-аналитическая система управления инцидентами в области обеспечения безопасности предназначена для оперативного управления событиями, возникающими на объектах инфраструктуры, где установлены различные датчики безопасности (например, пожарные, водные и другие датчики). Система сочетает в себе функции трекера задач (аналогичного Jira) с интеграцией данных от датчиков для автоматического инициирования predefined процессов реагирования на события тревожного характера.

Система предоставляет интуитивно понятный интерфейс для управления датчиками, бизнес-процессами, тикетами и интеграцией с внешними датчиками. Кроме того, она обеспечивает отправку уведомлений в режиме реального времени и обладает гибкостью настройки очередей реагирования, что исключает непредусмотренные временные задержки при угрозах безопасности.

Система предназначена для выполнения следующих функций:

- **Автоматизация тревожных событий:** все сигналы от датчиков (например, срабатывание пожарной сигнализации или сигнал о затоплении) автоматически декодируются, привязываются к конкретным аварийным протоколам и запускают заранее определенные процессы реагирования.

- **Координация бизнес-процессов безопасности:** система управляет процессами и очередями задач для каждого инцидента, автоматически координирует работу ответственных должностных лиц (например, пожарная инспекция, службы охраны) с помощью тикетов.

- **Регулирование прав доступа и безопасность:** все запросы — как для мониторинга, так и для управления — проходят через прокси-слой, который проверяет авторизацию каждого пользователя в системе.

- **Уведомления и взаимодействие:** уведомления, комментарии и автоматические статусы задач облегчают взаимодействие между участниками инцидентов и обеспечивают прозрачность управления инцидентами для обеспечения безопасности.

3.2. Эксплуатационное назначение

Эксплуатационное назначение системы заключается в обеспечении работы комплекса программных и аппаратных средств для управления и реагирования на события, связанные с

безопасностью на объекте. Как система безопасности, ее основная функция — получение сигналов от различных датчиков, управление событиями и инцидентами, координация участников инцидентных процессов, а также автоматизация обработки бизнес-процессов, связанных с реагированием на события.

1. Мониторинг безопасности объектов

Основной эксплуатационной задачей системы является сбор информации с датчиков, установленных на объекте. Каждый датчик связан с конкретной зоной или объектом контроля (например, серверная, склад, производственный цех), при этом тип событий могут быть различными (пожар, затопление, несанкционированный доступ).

2. Автоматизация работы с инцидентами

После срабатывания датчика система автоматически создает инцидент (тикет), который фиксируется в системе в соответствующей очереди задач. Для каждого тикета задается ответственный сотрудник, параметры и приоритет решения, в зависимости от характера инцидента.

Система координирует выполнение действий по реагированию: отслеживает текущий статус инцидента, назначает исполнителей, уведомляет их о необходимости действий и контролирует выполнение проверок и реагирования.

3. Обработка сигналов и автоматизация процессов реагирования

Эксплуатация системы также включает автоматический запуск заданных бизнес-процессов с привлечением необходимых ресурсов и служб при срабатывании датчиков.

Например, при обнаружении пожара датчиком дыма система автоматически уведомит пожарную службу и сотрудников охраны, создаст инцидент с необходимыми исполнителями.

4. Управление очередями работы (бизнес-процессы)

Операторы системы могут настроить очереди инцидентов различной природы (например, «Пожар», «Протечка», «Взлом») и для каждой очереди регламентировать шаги, которые необходимо предпринять для устранения инцидента.

5. Гибкие уведомления и уведомления на основе ролей

В рамках эксплуатации система позволяет настроить схемы уведомлений в зависимости от типа инцидента, его критичности, а также на основе ролевой модели ответственности.

Например, при наступлении инцидента критического уровня вся цепь командования (дежурный оператор, менеджер службы безопасности, руководитель объекта) должна быть оперативно уведомлена в зависимости от их роли в процессе реагирования.

4. ТРЕБОВАНИЯ К ПРОГРАММЕ

4.1. Требования к функциональным характеристикам

4.1.1. Состав выполняемых функций

Для удобства чтения раздел «Состав выполняемых функций» имеет сокращенную нумерацию: ссылаясь на разделы, содержащиеся в «Составе выполняемых функций», из других разделов или других документов, стоит использовать расширенную нумерацию, т. е. пункт «1. Регистрация» стоит нумеровать как «4.1.1.1. Регистрация».

Система должна обладать набором компонент, обеспечивающих ее ключевую функциональность, Требования к этим компонентам описаны ниже:

1. Компонент, отвечающий за пользователей и группы, должен:
 - 1.1. Обеспечивать возможность регистрации пользователя по почте и паролю с помощью пользовательского интерфейса
 - 1.2. Обеспечивать возможность авторизации пользователя по почте и паролю с помощью пользовательского интерфейса
 - 1.3. Обеспечивать управление пользователями: позволять создавать, редактировать, просматривать комплексную информацию о пользователях с помощью пользовательского интерфейса
 - 1.3.1. Информация о пользователях должна содержать в себе: ФИО, адрес электронной почты, номер телефона и другую информацию
 - 1.4. Обеспечивать управление пользователями: позволять создавать, редактировать, просматривать комплексную информацию о группах с помощью пользовательского интерфейса
 - 1.4.1. Группа должна содержать в себе несколько пользователей и информацию о них
 - 1.5. Обеспечивать сбор нескольких пользователей в группы, их добавление и удаление оттуда с помощью пользовательского интерфейса
2. Компонент, отвечающий за дежурства, должен:
 - 2.1. Обеспечивать управление дежурствами: позволять создавать, редактировать, просматривать комплексную информацию о дежурствах с помощью пользовательского интерфейса
 - 2.2. У группы пользователей должна быть возможность задавать график дежурств с помощью пользовательского интерфейса
 - 2.2.1. Дежурство группы пользователей – это временной отрезок, в рамках которого выделяется один ответственный пользователь

- 2.2.2. Если у группы пользователей задан график дежурств, то обязательно должен быть один ответственный пользователь в текущий момент времени
- 2.2.3. Дежурство должно быть периодичным и цикличным
- 3. Компонент, отвечающий за прием сигналов от датчиков должен:
 - 3.1. Обеспечивать управление датчиками: позволять создавать, редактировать, просматривать комплексную информацию о датчиках с помощью пользовательского интерфейса
 - 3.1.1. Информация о датчиках должна содержать в себе описание датчика, его местоположение, за что он отвечает, какой бизнес процесс должен запуститься при его срабатывании и другую информацию
 - 3.2. Реализовывать получение и обработку сигналов от датчиков
 - 3.3. При получении сигнала от датчика запускать заданный бизнес процесс, который закреплен за датчиком
- 4. Компонент, отвечающий за процессы должен:
 - 4.1. Обеспечивать управление процессами: позволять создавать, редактировать, просматривать комплексную информацию об очередях с помощью пользовательского интерфейса
 - 4.1.1. Информация об очередях должна содержать в себе граф статусов задач в нем, ответственные за очередь и другую информацию
 - 4.2. Обеспечивать возможность добавлять тикеты в заданную очередь
 - 4.2.1. При создании тикета должны автоматически определиться ответственные за эту очередь и призваться в тикет
- 5. Компонент, отвечающий за тикеты должен:
 - 5.1. Обеспечивать управление тикетами: позволять создавать, редактировать, просматривать комплексную информацию о тикетах с помощью пользовательского интерфейса
 - 5.1.1. Информация о тикете должна содержать в себе следующее:
 - 5.1.1.2. Название
 - 5.1.1.3. Описание
 - 5.1.1.4. Статус
 - 5.1.1.5. Дата создания
 - 5.1.1.7. Автор
 - 5.1.1.8. Исполнитель
 - 5.1.1.9. Комментарии

5.1.1.10. Процесс

- 5.2. Обеспечивать возможность добавлять комментарии к тикету с помощью пользовательского интерфейса
- 5.3. Обеспечивать отправку уведомлений тем, кого упомянули в тикете
- 5.4. Обеспечивать смену статуса у тикета в соответствии с заданным бизнес процессом тикета с помощью пользовательского интерфейса
- 6. У компонента, отвечающего за уведомления:
 - 6.1. Уведомления должны иметь возможность быть реализованы следующими способами:
 - 6.1.1. Отправка sms по номеру телефона
 - 6.1.2. Отправка сообщения по электронной почте
 - 6.2. Должна быть возможность настраивать уведомления по типу, уведомлять пользователей по заданному типу с помощью пользовательского интерфейса
 - 6.3. Должна быть возможность управлять уведомлениями: создавать, с помощью интерфейса API
 - 6.4. Должна быть возможность настраивать уведомления по группам с помощью интерфейса API: уведомлять дежурного в группе
 - 6.5. Должна быть возможность задавать правила уведомлениям с помощью интерфейса API: когда, с какой периодичностью, каких пользователей уведомлять
 - 6.6. Должна быть возможность привязывать уведомление к статусу с помощью пользовательского интерфейса: в этом случае все тикеты из этой очереди должны уведомлять пользователей по заданным правилам
- 7. Прокси
 - 7.1. Система должна содержать в себе прокси, через которую будут проходить все запросы во внутренние сервисы и проверяться авторизация пользователя
- 8. Пользовательский интерфейс
 - 8.1. Весь функционал системы, с которым предполагается взаимодействие пользователя, должен быть обёрнут в удобный и понятный веб-интерфейс.

4.1.2. Организация входных данных

Входными данными для приложения являются строки, числа и прочие данные, введенные пользователем при взаимодействии с пользовательским интерфейсом с помощью соответствующих элементов интерфейса. Так же к входным данным относятся данные, посылаемые датчиками в систему.

4.1.3. Организация выходных данных

На выход подаются данные в формате, выбранном разработчиком. Доступ к выходным данным осуществляется через пользовательский интерфейс.

4.2. Требования к временным характеристикам

Время оповещения ответственных людей с момента срабатывания какого-либо датчика при стабильном подключении к сети Интернет со скоростью передачи данных в 25Мбит/сек и более не должно превышать 3 секунд.

4.3. Требования к интерфейсу

Интерфейс должен быть удобным для администрирования и интуитивно понятным. Дизайнерские решения остаются на усмотрение разработчика. Веб-страница должна корректно отрисовываться на экранах размера 1366 пикселей в ширину и 700 пикселей в высоту, а также на любых других экранах, размеры которых превышают эти значения. Вне зависимости от экранов в рамках допустимых размеров элементы веб-страницы не должны блокироваться и должны корректно выполнять свою функциональную роль.

4.4. Требования к надежности

Некорректные действия и вводимые данные пользователя не должны влиять на работоспособность как визуальной, так и серверной части системы.

Приложение должно обеспечивать сохранность данных пользователей посредством средств криптографии и соблюдением мер по сохранности данных.

4.5. Условия эксплуатации

Не требует специального обслуживания, может эксплуатироваться пользователем любого уровня подготовки.

4.6. Требования к составу и параметрам технических средств

Требования к составу и параметрам технических средств к системе в целом не предъявляются. Требования к отдельным компонентам системы описаны в следующих документах: настоящем Техническом задании «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Серверная часть: процессы интеграции, обработки данных и управления», Техническом задании «Информационно-аналитическая система управления инцидентами в области обеспечения

безопасности: управление участниками системы, уведомлениями и дежурствами» и настоящем Техническом задании «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Клиентская часть».

4.7. Требования к информационной и программной совместимости

Требования к информационной и программной совместимости к системе в целом не предъявляются. Требования к отдельным компонентам системы описаны в следующих документах: настоящем Техническом задании «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Серверная часть: процессы интеграции, обработки данных и управления», Техническом задании «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности: управление участниками системы, уведомлениями и дежурствами» и настоящем Техническом задании «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Клиентская часть».

4.8. Требования к маркировке и упаковке

Система распространяется в виде электронного пакета, содержащего программную документацию, приложение (исполняемые и прочие необходимые для работы файлы).

4.9. Требования к транспортировке и хранению

Программный продукт может храниться и транспортироваться на любом носителе информации или в облачном хранилище.

5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

5.1. Предварительный состав программной документации

- 1) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности». Техническое задание (ГОСТ 19.201-78) [7];
- 2) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Серверная часть: процессы интеграции, обработки данных и управления». Техническое задание (ГОСТ 19.201-78) [7];
- 3) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности: управление учатниками системы, уведомлениями и дежурствами». Техническое задание (ГОСТ 19.201-78) [7];
- 4) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Клиентская часть». Техническое задание (ГОСТ 19.201-78) [7];
- 5) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности». Программа и методика испытаний (ГОСТ 19.301-78) [10];
- 6) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Серверная часть: процессы интеграции, обработки данных и управления». Программа и методика испытаний (ГОСТ 19.301-78) [10];
- 7) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности: управление учатниками системы, уведомлениями и дежурствами». Программа и методика испытаний (ГОСТ 19.301-78) [10];
- 8) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Клиентская часть». Программа и методика испытаний (ГОСТ 19.301-78) [10];
- 9) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Серверная часть: процессы интеграции, обработки данных и управления». Руководство программиста (ГОСТ 19.505-79) [12];
- 10) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности: управление учатниками системы, уведомлениями и дежурствами». Руководство программиста (ГОСТ 19.505-79) [12];
- 11) «Информационно-аналитическая система управления инцидентами в области

обеспечения безопасности. Клиентская часть». Руководство оператора (ГОСТ 19.505-79) [12];

12) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Серверная часть: процессы интеграции, обработки данных и управления». Текст программы (ГОСТ 19.401-78) [13];

13) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности: управление учатниками системы, уведомлениями и дежурствами». Текст программы (ГОСТ 19.401-78) [13];

14) «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности. Клиентская часть». Текст программы (ГОСТ 19.401-78) [13];

5.2. Специальные требования к программной документации

1) Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 и ГОСТами к каждому виду документа (см. п. 5.1).

2) Пояснительная записка должна быть загружена в систему Антиплагиат через LMS

«НИУ ВШЭ». Лист, подтверждающий загрузку пояснительной записки, сдается в учебный офис вместе со всеми материалами не позже, чем за день до защиты курсовой работы.

3) Вся документация также воспроизводится в печатном виде, она должна быть подписана академическим руководителем образовательной программы 09.03.04

«Программная инженерия», руководителем разработки и исполнителем перед сдачей курсовой работы в учебный офис, не позже одного дня до защиты.

4) Документация также сдается в электронном виде в формате .pdf или .docx, а программа – в архиве формата .zip или .rar.

5) Все документы перед защитой курсовой работы должны быть загружены в информационно-образовательную среду НИУ ВШЭ LMS (Learning Management System) в личном кабинете, дисциплина – «Дипломное проектирование 2024-25», одним архивом.

6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

6.1. Ориентировочная экономическая ценность

Для оценки экономической ценности вашей системы выделим основные аспекты, которые принесут пользу предприятиям, внедряющим систему

- **Снижение затрат на реагирование на инциденты**

Интеграция с датчиками безопасности (пожарными, водными и т.д.) позволяет оперативно реагировать на угрозы, минимизируя ущерб от пожаров, затоплений или других аварий. Это сокращает расходы на восстановление и снижает риски потерь

- **Оптимизация трудозатрат**

Благодаря гибким уведомлениям и автоматизации распределения задач между дежурными сотрудниками система уменьшает количество ручных операций, что экономит время и снижает вероятность человеческих ошибок.

- **Предотвращение простоев**

В системах безопасности время реагирования критично. Ваша система способствует предотвращению простоев производств, вызванных авариями.

- **Повышение эффективности управления инцидентами**

Более прозрачное управление процессами безопасности и интеграция с проектным управлением увеличивают общую производительность.

Оценочная экономическая выгода

Внедрение системы может снизить финансовые ущербы, связанный с инцидентами, отслеживаемыми датчиками

6.2. Предполагаемая потребность

На фоне повышения требований к обеспечению безопасности объектов многие компании, особенно крупные предприятия и промышленные организации, нуждаются в современных решениях для мониторинга и оперативного реагирования.

Такие системы, как Jiga, предназначены в основном для управления инцидентами и не интегрируются с физическими датчиками или системами безопасности. Наша система закрывает этот пробел, предлагая уникальную функциональность.

Рассуждая в реалиях нашей страны, законодательные нормы в России (например,

требования пожарной и экологической безопасности) обязывают организации внедрять системы мониторинга и реагирования. Это создаёт базу для потенциального спроса.

Ключевые потребители:

1. Крупные промышленные предприятия.
2. Государственные учреждения (особенно критически важная инфраструктура).
3. Частные компании (торговые и офисные центры).
4. Строительные компании (для обеспечения безопасности на строительных объектах).

6.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами

6.3.1 Российские аналоги

Многие российские системы сосредоточены либо на управлении проектами (например, 1С:Управление проектами или Яндекс.Трекер), либо на безопасности (например, системы пожарного мониторинга).

Наши преимущества:

1. Комбинация проектного управления и систем безопасности;
2. Лучшая адаптация под требования российского законодательства;
3. Возможность интеграции с датчиками, которые широко используются в России;

6.3.2 Зарубежные аналоги

Зарубежные решения, такие как Microsoft Project или Atlassian Jira, не предназначены для работы с датчиками безопасности и не имеют возможности так гибко кастомизировать уведомления, что критически важно для максимально эффективного взаимодействия системы с дежурными.

Наши преимущества:

1. Низкая стоимость владения (особенно важна для среднего и малого бизнеса);
2. Локализация и поддержка клиентов на русском языке;
3. Упрощённая интеграция с российскими устройствами и датчиками;
4. Низкая стоимость по сравнению с зарубежными решениями за счёт отсутствия валютной составляющей.

7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

Стадии и этапы разработки были выявлены с учетом ГОСТ 19.102-77 [2]:

1. Техническое задание

1.1. Обоснование необходимости разработки программы:

1.1.1. Первоначальная постановка задачи научным руководителем.

2. Разработка и утверждение технического задания:

2.1. Обсуждение и определение основных требований к программе вместе с научным руководителем.

2.2. Выбор стека технологий для разработки программы.

2.3. Определение этапов и стадий разработки и документации на нее.

2.4. Согласование технического задания с научным руководителем

3. Технический проект

3.1. Разработка технического проекта:

3.1.1. Определение формы взаимодействия пользователя с программой.

3.1.2. Разработка структуры программы.

3.1.3. Определение конфигурации технических средств.

3.2. Утверждение технического проекта:

3.2.1. Согласование технического проекта с научным руководителем.

4. Рабочий проект

4.1. Разработка программы:

4.1.1. Разработка, программирование и отладка программы.

4.2. Разработка программной документации:

4.2.1. Разработка программной документации из перечня, составленного на этапе технического задания.

4.3. Испытания программы:

4.3.1. Разработка программы и методики испытаний.

4.3.2. Корректировка программы и программной документации по итогам испытаний.

5. Внедрение

5.1. Подготовка и передача программы:

5.1.1. Подготовка и передача программы и программной документации для защиты.

5.1.2. Защита программы.

5.1.3. Передача программы в архив.

Программа и документация к ней разрабатываются к утвержденным приказом декана ФКН срокам.

8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

8.1. Виды испытаний

Производится проверка корректного выполнения программой заложенных в нее функций, то есть осуществляется функциональное тестирование программы.

Функциональное тестирование осуществляется в соответствии с документом «Информационно-аналитическая система управления инцидентами в области обеспечения безопасности». Программа и методика испытаний (ГОСТ 19.301-79), в котором указывают [14]:

- 1) перечень функций программы, выделенных в программе для испытаний, и перечень требований, которым должны соответствовать эти функции (со ссылкой на пункт 4.1.1. настоящего Технического задания);
- 2) перечень необходимой документации и требования к ней (со ссылкой на пункт 5 настоящего Технического задания);
- 3) методы испытаний и обработки информации;
- 4) технические средства и порядок проведения испытаний.

8.2. Общие требования к приемке работы

Проверка программного продукта, в том числе и на соответствие техническому заданию, осуществляется исполнителем вместе с заказчиком согласно «Программе и методике испытаний», а также пункту 5.2.

Защита выполненного проекта осуществляется комиссией, состоящей из преподавателей департамента программной инженерии, в утверждённые приказом декана ФКН сроки.

СПИСОК ИСТОЧНИКОВ

- 1) ГОСТ 19.101-77 Виды программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 2) ГОСТ 19.102-77 Стадии разработки. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 4) ГОСТ 19.104-78 Основные надписи. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 5) ГОСТ 19.105-78 Общие требования к программным документам. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 7) ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 8) ГОСТ 19.603-78 Общие правила внесения изменений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 9) ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 10) ГОСТ 19.404-79 Программа и методика испытаний. Требования к содержанию и оформлений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 11) ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 12) ГОСТ 19.505-79 Руководство оператора. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 13) ГОСТ 19.401-78 Текст программы. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 14) ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

Лист регистрации изменений

[illegible]