**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики-процессов управления**

**Программа бакалавриата**

**“Большие данные и распределенная цифровая платформа”**

**Техническое задание**

**на разработку “Умного электронного сейфа”**

**по дисциплине «Интернет вещей»**

**Студент гр. 23Б15-пу**

**Абрахин Е. Д.**

**Бек В. А.**

**Преподаватель**

**Дик Г. Д.**

**Санкт-Петербург**

**2025 г.**

Оглавление

[1. Общие положения 4](#_Toc211530742)

[1.1. Полное наименование системы и её условное обозначение 4](#_Toc211530743)

[1.2. Организационная структура и роли 4](#_Toc211530744)

[1.3. Сроки начала и окончания работ 4](#_Toc211530745)

[1.4. Источники и порядок финансирования 4](#_Toc211530746)

[1.5. Результаты разработки 5](#_Toc211530747)

[1.6. Термины и определения 5](#_Toc211530748)

[2. Назначение и цели создания системы 6](#_Toc211530749)

[2.1. Назначение 6](#_Toc211530750)

[2.2. Основные цели проекта 6](#_Toc211530751)

[2.3. Ограничения и допущения 6](#_Toc211530752)

[2.4. Критерии успешности проекта 7](#_Toc211530753)

[3. Характеристика объекта автоматизации 8](#_Toc211530754)

[3.1. Объект и границы автоматизации 8](#_Toc211530755)

[3.2. Процессы автоматизации 8](#_Toc211530756)

[3.3. Аппаратные компоненты (примерный перечень) 8](#_Toc211530757)

[3.4. Условия эксплуатации и сценарии использования 9](#_Toc211530758)

[4. Требования к системе 10](#_Toc211530759)

[4.1. Требования к структуре и функционированию системы (функциональные и некоторые нефункциональные требования) 10](#_Toc211530760)

[4.1.1. Функциональные требования 10](#_Toc211530761)

[4.1.2. Нефункциональные требования (ключевые метрики) 10](#_Toc211530762)

[4.1.3. Удобство и интерфейс 10](#_Toc211530763)

[4.2. Требования к техническому обеспечению (КТС) 10](#_Toc211530764)

[4.3. Требования к информационному обеспечению системы 11](#_Toc211530765)

[4.4. Требования к программному обеспечению (ПО) 12](#_Toc211530766)

[4.4.1. Общие требования 12](#_Toc211530767)

[4.4.2. Безопасность ПО 12](#_Toc211530768)

[5. Требования к программной документации 13](#_Toc211530769)

[5.1. Перечень обязательных документов 13](#_Toc211530770)

[5.2. Требования к оформлению документации 13](#_Toc211530771)

[6. Стадии и этапы разработки 14](#_Toc211530772)

[6.1. Общая структура этапов 14](#_Toc211530773)

[6.2. Управление изменениями ТЗ 14](#_Toc211530774)

[7. Порядок контроля и приёмки 15](#_Toc211530775)

[8. Дополнительно 16](#_Toc211530776)

[8.1. Обозначения и сокращения 16](#_Toc211530777)

# Общие положения

# Полное наименование системы и её условное обозначение

**Полное наименование**: система «Умный электронный сейф» (далее — «Система»).

**Условные обозначения**: умный сейф, электронный сейф, сейф. В тексте допускается использование как полного, так и сокращённого наименования.

# Организационная структура и роли

Проект выполняется в рамках учебной дисциплины «Интернет вещей». Структура и распределение ролей проекта:

**Заказчик / Куратор проекта (Преподаватель)** — принимает результаты, утверждает основные решения по оценке рисков и безопасности, утверждает акты приёмки.

**Разработчики аппаратной части / программного обеспечения** — проектирование схемы, подбор компонентов, сборка прототипа, тестирование КТС, реализация прошивки микроконтроллера, серверной части/хранилища и интерфейсов (Абрахин Е. Д., Бек В. А.)

**Инженер по тестированию / Приёмщик** — подготовка и проведение тестов, подготовка программ и методик испытаний, оформление актов сдачи-приёмки. (Бек В. А.)

**Владелец репозитория (GitHub)** — отвечает за структуру репозитория, лицензию, инструкции по сборке и развёртыванию. (Абрахин Е. Д.)

# Сроки начала и окончания работ

В документе указаны ориентировочные сроки:

* начало работ — декабрь 2025 г.,
* окончание — не позднее мая 2026 г.

# Источники и порядок финансирования

Проект реализуется в рамках образовательной программы «Интернет вещей» факультета Прикладной математики — процессов управления СПбГУ. Финансирование и выдача аппаратных комплектов/компонентов — в пределах каких-нибудь средств.

# Результаты разработки

Результаты проекта представляются в виде набора артефактов:

* Аппаратный прототип (фотографии/сборочные схемы, спецификация компонентов).
* Репозиторий на GitHub с открытой лицензией (исходный код прошивки, клиентских интерфейсов, скрипты сборки), README, инструкции по сборке и развёртыванию.
* Документация: общее описание системы, пользовательское руководство, инструкция по сборке/установке, программа и методика испытаний, отчёт по тестированию.
* Тестовые материалы: сценарии и результаты испытаний, логи.
* Акт сдачи-приёмки и презентация.

Все электронные материалы передаются на согласованном носителе и публикуются (в части ПО и документации) в открытом репозитории. Критерии приёмки и порядок проведения испытаний описаны в разделе «Порядок контроля и приёмки».

# Термины и определения

Краткие определения ключевых терминов (имеющиеся в документе и расширенные для однозначности):

* **Идентификация** — процесс установления личности/учётной записи пользователя по уникальному идентификатору (ID).
* **Аутентификация** — подтверждение прав доступа пользователя (например, проверка цифрового кода).
* **Прошивка (firmware)** — программное обеспечение микроконтроллера, управляющее аппаратной частью сейфа.
* **База данных (БД)** — структурированное хранилище событий, конфигурации и шаблонов разрешённых пользователей.
* **КТС** — комплекс технических средств (микроконтроллер, датчики, модуль связи и т.п.).

# Назначение и цели создания системы

# Назначение

Прототип «Умного электронного сейфа» служит для повышения надёжности и защищённости хранения ценностей посредством цифровых методов контроля доступа и оповещения; демонстрации подхода к интеграции IoT-технологий в локальные устройства безопасности; освоения студентами практики проектирования киберфизических систем.

# Основные цели проекта

* Разработать аппаратно-программный прототип сейфа с цифровым замком и системой идентификации пользователей.
* Реализовать средства обнаружения несанкционированного доступа (вибрация, вскрытие, многократный неверный ввод кода) и механизмы реагирования (сигнализация, уведомления).
* Обеспечить возможность дистанционного мониторинга и уведомлений через IoT-каналы (Wi-Fi / Bluetooth) и продемонстрировать простой мобильный интерфейс.
* Обеспечить открытость разработки: весь код — в репозитории GitHub под свободной лицензией, с инструкциями для воспроизведения и сборки. (Информация о репозитории и лицензии согласована с заказчиком/руководителем проекта.)
* Исследовать и оценить показатели надёжности, автономности и удобства использования (включая замер времени реакции, устойчивость к сбоям, время автономного питания).

# Ограничения и допущения

* Проект — прототип; не предполагается сертификация или коммерческая эксплуатация без доработок.
* Аппаратная и программная части могут изменяться по мере освоения технологий; архитектура должна быть модульной.
* Код публикуется с открытой лицензией; при использовании сторонних библиотек следует соблюдать их лицензии.
* Из-за студенческого статуса возможны компромиссы по уровню защищённости: проекты ориентируются на демонстрацию принципов, а не на полную защищённость промышленного уровня.

# Критерии успешности проекта

* Прототип открывает и закрывает сейф по корректному коду; логируются события доступа и тревоги.
* Уведомления успешно доставляются через выбранный канал (Wi-Fi или Bluetooth) в большинстве тестовых сценариев.
* Система работает автономно не менее заявленного времени (см. п. 4.1 и 4.2).

# Характеристика объекта автоматизации

# Объект и границы автоматизации

Объект автоматизации — электронный сейф с микроконтроллером, набором датчиков и модулем беспроводной связи, реализующий хранение ключевых функций управления замком, мониторинга состояния и оповещения владельца. Система — киберфизический комплекс: аппаратная часть (корпус, замок, датчики, батарея) + ПО (прошивка, интерфейс, база событий).

# Процессы автоматизации

* **Идентификация и аутентификация пользователя** — ввод цифрового кода, проверка по локальной БД шаблонов/учётных записей, предоставление/отказ в доступе.
* **Управление электромеханическим замком** — открытие/закрытие, аварийная блокировка при множественных ошибках, защита от попыток механического вмешательства.
* **Мониторинг питания** — измерение уровня заряда аккумулятора, уведомления о низком заряде, управление энергопотреблением.
* **Обнаружение угроз** — обработка датчиков вибрации, дверного контакта, звуковых аномалий; определение подозрительных событий.
* **Реагирование на инциденты** — активация звуковой/световой сигнализации, запись события в лог, отправка уведомления владельцу.
* **Интеграция с мобильными устройствами**— отправка статуса, получение команд управления (при наличии сети).
* **Логирование и хранение событий** — локальная БД с возможностью резервного экспорта/синхронизации.

# Аппаратные компоненты (примерный перечень)

* Микроконтроллер с поддержкой Wi-Fi/Bluetooth и низким энергопотреблением (описанные требования по управлению устройствами см. 4.2).
* Модуль управления замком (электромеханический или сервопривод).
* Датчики: вибрации/удара, датчик открытия дверцы, опционально датчик звука.
* Система сигнализации: светодиодная индикация, зуммер ≥85 дБ.
* Аккумуляторная батарея (7.4 В, ≥1000 mAh) с защитой от глубокого разряда.

# Условия эксплуатации и сценарии использования

* Домашнее/общежитское/лабораторное окружение с переменной доступностью сети.
* Работа в автономном режиме при отсутствии сети.
* Экстренные сценарии: многократный неверный ввод кода, физическая попытка вскрытия, разряд батареи.

# Требования к системе

# Требования к структуре и функционированию системы (функциональные и некоторые нефункциональные требования)

# Функциональные требования

Система должна обеспечивать следующие функции:

* Ввод и проверка цифрового кода пользователя.
* Идентификация и аутентификация пользователей.
* Управление замком: открытие/закрытие, аварийная блокировка при множественных ошибках ввода.
* Контроль уровня заряда и уведомление о необходимости подзарядки/замены батареи.
* Обнаружение попыток несанкционированного доступа: обработка данных от датчиков вибрации/шока, открытия, звука.
* Активация сигнализации (звуковой/световой) и отправка уведомлений пользователю по Wi-Fi/Bluetooth.
* Локальное логирование всех ключевых событий (входы, попытки взлома, потеря питания, обновления).
* Работа в автономном режиме при отсутствии связи с сетью (локальные функции и логирование).

# Нефункциональные требования (ключевые метрики)

* Время проверки и обработки кода: не более 1 секунды.
* Задержка срабатывания замка после правильного кода: не более 0,5 секунды.
* Время активации сигнализации при попытке взлома: не более 0,2 секунды.
* Длительность автономной работы от батареи: не менее 2 месяцев при типовой нагрузке.
* Устойчивость к сбоям питания: корректное восстановление состояния после перезапуска, без повреждения данных.

# Удобство и интерфейс

Наличие простого и понятного веб-или мобильного интерфейса для мониторинга состояния сейфа и получения уведомлений. Документация по интерфейсам (API) и примеры использования.

# Требования к техническому обеспечению (КТС)

**Общие требования** к компонентам (минимальные/ориентировочные):

* **Микроконтроллер**: поддержка управления не менее чем 8 периферийными устройствами, поддержка Wi-Fi или Bluetooth, режимы низкого энергопотребления; потребление в режиме ожидания — не более 70 мА.
* **Модуль управления замком**: электромеханический или сервоприводной; время срабатывания ≤ 0,5 с; ресурс не менее 100 000 циклов; питание 5–12 В.
* **Датчики безопасности**: датчик вибрации/удара, датчик открытия дверцы, опционально — датчик звука.
* **Световая и звуковая сигнализация**: LED-индикация; зуммер ≥ 85 дБ; время отклика ≤ 0,2 с.
* **Аккумуляторная батарея**: напряжение 7.4 В, ёмкость ≥ 1000 mAh, защита от глубокого разряда и короткого замыкания; автономная работа — не менее 6 месяцев (при заявленной нагрузке).
* **Корпус**: металл или прочный полимер, предотвращающий несанкционированный доступ к электронике при несанкционированном вскрытии.
* **Коммуникационный модуль**: поддержка Wi-Fi (предпочтительно 802.11 b/g/n) или Bluetooth; каналы должны поддерживать шифрование.

Примечание: при отсутствии точной модели компонентов допускается использовать похожие платы/модули при условии документирования выбранной замены и её характеристик.

# Требования к информационному обеспечению системы

* **Структурированное хранение данных.** Локальная БД должна хранить учётные записи пользователей, журналы событий (метка времени, тип события, источник события), конфигурацию устройства.
* **Целостность и непротиворечивость данных.** Механизмы простой проверки целостности и корректного восстановления после сбоя.
* **Совместимость форматов.** Форматы логов и экспортируемых данных — текстовые/JSON-совместимые, чтобы упростить анализ в сторонних инструментах.
* **Защита данных:** аутентификация доступа к интерфейсам управления; шифрование конфиденциальных данных (например, ключей) в хранилище; шифрование трафика при передаче (TLS для Wi-Fi).
* **Резервное копирование и экспорт данных.** Возможность выгружать логи и конфигурации вручную (USB/флеш) или по сети для резервного хранения.

# Требования к программному обеспечению (ПО)

# Общие требования

* ПО должно быть основано на свободно распространяемом ПО или открытых библиотеках (с учётом их лицензий).
* Архитектура ПО — модульная: отдельные слои/модули для управления замком, обработки событий/сенсоров, коммуникаций, логирования, пользовательского интерфейса.
* Проект должен содержать документацию по сборке, инструкции для запуска и тестовые сценарии.

# Безопасность ПО

* Хранение ключевых секретов должно быть защищённым; при развертывании в тестовой среде — использовать тестовые секреты.
* Поддержка защищённых каналов связи (шифрование передачи).
* Учетная запись администратора должна иметь возможность восстановления/сброса с защитой от злоупотреблений (например, аппаратный сброс с подтверждением).

# Требования к программной документации

# Перечень обязательных документов

README (корневой файл репозитория)

* Краткое назначение проекта;
* Минимальные шаги «быстрый старт» (build → run → demo);
* Структура репозитория;
* Требования к ОС/платформе.

Руководство пользователя (UserManual)

* Назначение и области применения.
* Описание интерфейсов (локальная панель, веб/мобильный интерфейс). Пошаговые сценарии: открытие/закрытие, регистрация пользователя, восстановление доступа, поведение при тревоге.
* Примеры экранов/команд.

Руководство по установке и настройке (InstallationGuide)

* Сборка: схема подключения модулей, перечень используемых плат и номеров пинов.
* ПО: сборка прошивки, установка зависимостей, развёртывание, запуск.
* Настройка сети, создание учётной записи администратора.

# Требования к оформлению документации

Текстовые документы публикуются в Markdown (для репозитория) и/или в PDF/Word для официальной сдачи. Иллюстрации — векторные или PNG/JPEG высокого разрешения. вся документация размещается в ветке docs/ репозитория.

# Стадии и этапы разработки

# Общая структура этапов

* Подготовительный этап
* Анализ и проектирование
* Разработка
* Тестирование
* Подготовка к демонстрации и приемке
* Сопровождение и доработка

# Управление изменениями ТЗ

Все изменения ТЗ оформляются как дополнения с указанием: автор изменения, суть изменения, обоснование, влияние на сроки и ресурсы. Каждая поправка получает номер и дату.

# Порядок контроля и приёмки

Заказчик (Преподаватель) принимает работу на основании предъявленных артефактов, демонстрации работы прототипа и отчётов по тестированию.

# Дополнительно

# Обозначения и сокращения

Таблица 1. Сокращения и обозначения

|  |  |
| --- | --- |
| **Обозначение / Сокращение** | **Определение** |
| СУБД | Система управления базами данных |
| UI | Пользовательский интерфейс |
| IoT | Интернет вещей |
| API | Программный интерфейс приложения |
| ПО | Программное обеспечение |
| ТЗ | Техническое задание |