

	<b>Спецификация требований к программному обеспечению для Проекта СКОТ</b>	Автор: TEAM18 № документа: TEAM18-v0.2 Дата: 2023-10-10 Страниц: 100
-----------------------------------------------------------------------------------	------------------------------------------------------------------------------------	-------------------------------------------------------------------------------

## Содержание

СОДЕРЖАНИЕ	1
ИСТОРИЯ ИЗМЕНЕНИЙ	2
1 ВВЕДЕНИЕ	4
1.1 Цели	4
1.2 Границы применения	4
1.3 Термины, аббревиатуры, сокращения	5
1.4 Ссылки	5
1.5 Краткий обзор	5
2 ОБЩЕЕ ОПИСАНИЕ	5
2.1 Описание изделия	5
2.1.1 Интерфейсы системы	5
2.1.2 Интерфейсы пользователя	5
2.1.3 Интерфейсы аппаратных средств ЭВМ	5
2.1.4 Интерфейсы программного обеспечения	6
2.1.5 Интерфейсы коммуникаций	6
2.1.6 Ограничения памяти	6
2.1.7 Действия	6
2.1.8 Требования настройки рабочих мест	6
2.2 Функции изделия	7
2.3 Характеристики пользователей	7
2.4 Ограничения	7
2.5 Предположения и зависимости	7
2.6 Распределение требований	7
3 ДЕТАЛЬНЫЕ ТРЕБОВАНИЯ	8
3.1 Функциональные требования	8
3.1.1 <Functional Requirement One>	8
3.2 Надежность	8
3.2.1 <Reliability Requirement One>	8
3.3 Производительность	8
3.3.1 <Performance Requirement One>	8
3.4 Ремонтопригодность	5
3.4.1 <Maintainability Requirement One>	5
3.5 Ограничения проекта	5
3.5.1 <Design Constraint One>	5
3.6 Требования к пользовательской документации	5
3.7 Используемые приобретаемые компоненты	5
3.8 Интерфейсы	5
3.8.1 Интерфейс пользователя	5
3.8.2 Аппаратные интерфейсы	5
3.8.3 Программные интерфейсы	5
3.8.4 Интерфейсы коммуникаций	5
3.9 Требования лицензирования	5
3.10 Применимые стандарты	5
ИНДЕКС	5

## История изменений

[illegible]

# 1 Введение

## 1.1 Цели

Этот документ определяет спецификацию требований к программному обеспечению системы контроля оборудования и техники (СКОТ). Системе необходимо хранить перечень ТО с полным набором инструкций необходимых действий и материалов, а также любых технических манипуляций с ОБ (СУЕТА, ТО, эксплуатация).

Спецификация предназначена для работников взаимодействующих с ОБ и персонала занимающимся ТО и СУЕТОЙ оборудования.

## 1.2 Границы применения

СКОТ - система контроля оборудования и техники.

ПО будет выполнять следующие функции:

- Управление графиками обслуживания оборудования в заводе.
- Отслеживание регламентных работ (ТО) для каждого оборудования.
- Отслеживание незапланированных ремонтов (СУЕТА) для каждого оборудования.
- Составление набора материалов и запчастей для каждого ТО.
- Расчет продолжительности обслуживания.

ПО не будет выполнять функции, отличные от управления и контроля оборудования, регламентного обслуживания и незапланированного ремонта.

Применение ПО Система контроля оборудования включает:

- Улучшенный мониторинг и контроль за графиками обслуживания оборудования.
- Автоматизированное составление наборов материалов и запчастей для каждого ТО.
- Точное планирование продолжительности обслуживания, учитывая разные критерии.

Важные преимущества включают повышение эффективности обслуживания, снижение риска отказов оборудования и оптимизацию использования ресурсов.

## 1.3 Термины, аббревиатуры, сокращения

СКОТ	Система Контроля Оборудования и Техники
ТО	Техническое Обслуживание, регламентные работы
ОБ	Оборудование
СУЕТА	Незапланированный ремонт (Служба Устранения Экстренных Технических Аварий)
ПО	Программное Обеспечение
ЖУРНАЛ	Информация о отдельной единице ОБ
БД	База Данных
ЧС	Чрезвычайная Ситуация

## 1.4 Ссылки

Обозначение	Расшифровка
[IEEE-830]	IEEE Std 830-1998

## 1.5 Краткий обзор

Данный документ структурирован согласно [IEEE-830].

Раздел 2 содержит описание поставляемой системы и схему её использования в Организации. Раздел 3 содержит функциональные и нефункциональные требования, предъявляемые к системе и необходимые для её проектирования.

## 2 Общее описание

### 2.1 Описание изделия

#### 2.1.1 Интерфейсы системы

Веб-интерфейс: Система предоставляет веб-интерфейс для управления и мониторинга оборудования.

API: Предоставляется программный интерфейс приложения для интеграции с другими системами или автоматизации процессов.

#### 2.1.2 Интерфейсы пользователя

Графический интерфейс пользователя (GUI): Пользователи могут взаимодействовать с системой через удобный графический интерфейс, чтобы создавать, редактировать и просматривать данные об оборудовании и ТО.

Консольный интерфейс: Для более технических пользователей или администраторов доступен консольный интерфейс для выполнения команд и скриптов.

#### 2.1.3 Интерфейсы аппаратных средств ЭВМ

Взаимодействие с оборудованием:

- Коммуникация с оборудованием по временному графику: Интерфейс должен поддерживать настройку и управление временными параметрами оборудования, такими как заданные промежутки времени для обслуживания.
- Коммуникация с оборудованием по наработке: Интерфейс должен обеспечивать поддержку мониторинга наработки оборудования и уведомления о необходимости обслуживания на основе наработки.
- Одновременное взаимодействие с несколькими устройствами: Интерфейс должен иметь возможность обработки и управления несколькими устройствами одновременно, чтобы обеспечить эффективное планирование и координацию обслуживания.

Характеристики конфигурации:

- Номера портов: Программное обеспечение должно иметь возможность работы через определенные порты для коммуникации с оборудованием.
- Наборы инструкций: Интерфейс должен поддерживать соответствующие наборы инструкций, необходимые для взаимодействия с разными типами оборудования.

Поддерживаемые устройства:

- Интерфейс должен поддерживать широкий спектр устройств, которые требуют контроля и обслуживания, включая разнообразные типы оборудования, запчасти и материалы.

Способы поддержки и протоколы:

- Интерфейс должен обеспечивать возможность взаимодействия с оборудованием через различные методы связи, такие как последовательный порт, сетевой протокол или другие протоколы, поддерживаемые конкретными устройствами.

#### 2.1.4 Интерфейсы программного обеспечения

Для работы системы требуются:

- базы данных: PostgreSQL, Redis
- веб-сервер: Nginx
- среда выполнения языка JavaScript: Node.js

#### 2.1.5 Интерфейсы коммуникаций

Локальные сетевые протоколы:

- Ethernet: Интерфейс должен поддерживать протокол Ethernet для коммуникации с оборудованием, подключенным к локальной сети. Это обеспечит передачу данных между программным обеспечением и оборудованием через стандартные сетевые возможности.
- TCP/IP: Интерфейс должен поддерживать протокол TCP/IP для установки соединения и передачи данных по сети. TCP/IP является широко распространенным протоколом в сетевых коммуникациях и обеспечивает надежное и согласованное взаимодействие.

Беспроводные протоколы:

- Wi-Fi: Если в вашем заводе используется беспроводная сеть, то интерфейс должен поддерживать протокол Wi-Fi для беспроводной связи с оборудованием. Это позволит программному обеспечению обмениваться данными с оборудованием посредством беспроводной сети.
- Bluetooth: Если у вас есть оборудование, которое поддерживает связь по Bluetooth, то интерфейс должен поддерживать соответствующий протокол для установки связи и передачи данных через Bluetooth.

Протоколы удаленного доступа:

- SSH (Secure Shell): Интерфейс должен поддерживать протокол SSH для безопасного удаленного доступа к оборудованию. Это позволит операторам или техническим специалистам подключаться к оборудованию из удаленных мест и выполнять необходимые действия по контролю и обслуживанию.
- VPN (Virtual Private Network): Если ваш завод использует VPN для безопасного удаленного подключения, то интерфейс должен обеспечивать поддержку соответствующих VPN-протоколов, чтобы обеспечить защищенную коммуникацию с оборудованием посредством виртуальной частной сети.

#### 2.1.6 Ограничения памяти

Для развертки базы данных необходимо минимум 10 гигабайт на сервере.

Для оперативной памяти минимум 4 гигабайт.

#### 2.1.7 Действия

- Улучшенный мониторинг и контроль за графиками обслуживания оборудования.
- Автоматизированное составление наборов материалов и запчастей для каждого ТО.
- Точное планирование продолжительности обслуживания, учитывая разные критерии.

#### 2.1.8 Требования настройки рабочих мест

Рабочее место должно иметь минимальное разрешение экрана, которое обеспечит отображение всех необходимых элементов интерфейса программного обеспечения без искажений или обрезания информации.

Изменение конфигурации интерфейса: Программное обеспечение должно предоставлять возможность настроить интерфейс в соответствии с конкретными потребностями рабочего места

или задачи. Это должно включать изменение расположения и размеров элементов интерфейса, добавление или удаление определенных функций или настройку предпочтений пользователя.

Поддержка различных языков: Если на заводе работают люди из разных стран или языковых групп, то программное обеспечение должно поддерживать многоязычный интерфейс. Это позволит настраивать программу на специфический язык для каждого рабочего места, облегчая взаимодействие с программой для всех пользователей.

Гибкость настроек времени и форматов: Рабочее место должно предоставлять возможность настройки временных параметров и форматов даты/времени в соответствии с требованиями местной временной зоны или корпоративных стандартов.

## **2.2 Функции изделия**

- Управление графиками обслуживания оборудования в заводе.
- Отслеживание регламентных работ (ТО) для каждого оборудования.
- Отслеживание незапланированных ремонтов (СУЕТА) для каждого оборудования.
- Составление набора материалов и запчастей для каждого ТО.
- Расчет продолжительности обслуживания.

## **2.3 Характеристики пользователей**

- Образовательный уровень: Пользователи должны иметь различные уровни образования, от среднего до высшего. Они должны обладать базовыми навыками работы с компьютером и программным обеспечением.
- Опыт: Пользователи должны иметь разный опыт работы с оборудованием и программным обеспечением. Некоторые должны быть опытными и хорошо знакомыми с процессами обслуживания оборудования, в то время как другие должны быть новичками и нуждаются в дополнительной поддержке и руководстве.
- Техническая квалификация: Пользователи должны иметь разный технический уровень и квалификацию. Они должны быть инженерами, техниками, операторами оборудования и другими специалистами. Некоторые пользователи должны обладать специализированными знаниями в области обслуживания и ремонта определенного оборудования.

## **2.4 Ограничения**

- Доступ к БД должен быть разделен на несколько уровней привилегий от работника до админа.
- В случае ЧС связанных с отключением или ошибкой(ПО) , ПО должно сохранить все данные, для последующего восстановления в ограниченный срок.
- БД должна поддерживать редактирование несколькими пользователями если они не редактируют один ЖУРНАЛ.

## **2.5 Предположения и зависимости**

- На территории предприятия должна быть организована локальная сеть для подключения всего ОБ
- Связь с мировым интернетом осуществляется только через центральный сервер
- Пользователи прошли обучение по работе с ПО и знакомы с техникой безопасности каждого отдельного ОБ

## **2.6 Распределение требований**

- Поддержка Безопасности ПО уровня AA+.ru

### 3 Детальные требования

Для каждой функции из списка детальных требований создано описание с использованием данного шаблона:

**Идентификатор требования:** Однозначно идентифицирует требование во всех документах СКОТ.

**Наименование:** Определяет функциональную группу, к которой относится требование.

**Описание:** Описание требований.

**Приоритет:** Определяет порядок, в котором должны выполняться требования. Приоритеты обозначены (от высшего к низшему) “1”, “2” и “3”.

**Риск:** Указывает риск невыполнения требования. Это показывает, насколько конкретное требование критично для системы. Существуют следующие уровни риска и связанное с ними воздействие на систему, если требование не выполняется или выполняется неправильно:

- **Критический (К)** – приведет к нарушению основной функциональности системы.

Система не может быть использована, если это требование не выполнено.

- **Высокий (В)** – повлияет на основную функциональность системы. Некоторые функции системы могут быть недоступны, но в целом системой можно пользоваться.

- **Средний (С)** – повлияет на некоторые функции системы, но не на основную функциональность. Система может использоваться с некоторыми ограничениями.

- **Низкий (Н)** – система может использоваться без ограничений, но с некоторыми обходными путями.

#### 3.1 Функциональные требования

##### 3.1.1 График обслуживания

Идентификатор требования: И-СНОВА-3-СЕНТЯБРЯ

Наименование: График обслуживания

Описание: Система должна поддерживать возможность задания графика обслуживания для каждого оборудования на заводе. Пользователь должен иметь возможность выбрать тип графика обслуживания (по времени, по наработке или одновременный) для каждого оборудования. Система должна автоматически генерировать уведомления о предстоящих работах по обслуживанию в соответствии с заданным графиком.

Приоритет: 1

Риск: Критический

### 3.1.2 Регламентные работы (ТО)

Идентификатор требования: РУТИНА

Наименование: Регламентные работы (ТО)

Описание: Система должна поддерживать возможность создания регламентных работ (ТО) для каждого оборудования. Каждое ТО должно содержать информацию о необходимых материалах, запчастях и их количестве. Длительность выполнения каждого ТО должна быть указана.

Приоритет: 1

Риск: Высокий

### 3.1.3 Управление материалами и запчастями

Идентификатор требования: ЗАВХОЗ

Наименование: Управление материалами и запчастями

Описание: Система должна поддерживать инвентаризацию материалов и запчастей, необходимых для выполнения регламентных работ. Пользователь должен иметь возможность добавлять, редактировать и удалять материалы и запчасти из системы. Система должна отображать информацию о доступных запасах материалов и запчастей и автоматически уведомлять о необходимости пополнения запасов.

Приоритет: 2

Риск: Средний

### 3.1.4 Интерфейс пользователя

Идентификатор требования: WEB-МОРДА

Наименование: Интерфейс пользователя

Описание: Интерфейс системы должен быть интуитивно понятным и легким в использовании. Пользователь должен иметь возможность легко навигировать по системе, просматривать и редактировать графики обслуживания, регламентные работы и информацию о материалах. Система должна предоставлять возможность генерации отчетов о выполненных регламентных работах и доступных материалах.

Приоритет: 3

Риск: Низкий

### 3.1.5 Отчетность о выполненных работах

Идентификатор требования: СЕКРЕТАРЬ

Наименование: Отчетность о выполненных работах

Описание: Система должна позволять генерировать отчеты о выполненных регламентных работах для каждого оборудования. Отчеты должны содержать информацию о дате и времени выполнения работ, продолжительности и использованных материалах.

Пользователь должен иметь возможность экспортировать отчеты в различных форматах, таких как PDF или Excel.

Приоритет: 2

Риск: Средний

### 3.1.6 Уведомления о графике обслуживания

Идентификатор требования: БУДИЛЬНИК

Наименование: Уведомления о графике обслуживания

Описание: Система должна отправлять уведомления пользователям о предстоящих работах по обслуживанию по заданному графику. Уведомления должны быть отправлены



по электронной почте или через мобильное приложение. Пользователь должен иметь возможность настроить предпочтительный способ получения уведомлений.

Приоритет: 1

Риск: Критический

### **3.1.7 Интеграция с другими системами**

Идентификатор требования: ПОДРУЖАЙКА

Наименование: Интеграция с другими системами

Описание: Система должна иметь возможность интеграции с другими системами, такими как система учета запасов или система планирования ресурсов предприятия (ERP).

Интеграция позволит автоматически обновлять информацию о доступных материалах и запчастях, а также синхронизировать данные о выполненных регламентных работах с другими системами. Приоритет: 2

Риск: Высокий

### **3.1.8 Мобильное приложение**

Идентификатор требования: АВРОРА

Наименование: Мобильное приложение

Описание: Система должна иметь мобильное приложение, доступное для установки на смартфоны и планшеты. Мобильное приложение должно предоставлять возможность просмотра и управления графиком обслуживания, отчетами о выполненных работах, а также получения уведомлений о предстоящих работах.

Приоритет: 1

Риск: Низкий

### **3.1.9 История изменений**

Идентификатор требования: ШПИОН

Наименование: История изменений

Описание: Система должна сохранять историю изменений в графике обслуживания и выполненных работах. Пользователи должны иметь возможность просмотра предыдущих версий графика и отчетов, а также видеть информацию о том, кто и когда внес изменения. История изменений поможет отслеживать и анализировать выполненные работы.

Приоритет: 2

Риск: Низкий

### **3.1.10 Пользовательские роли и права доступа**

Идентификатор требования: СИСТЕМА-РАБ-ВЛАДЕЛЕЦ

Наименование: Пользовательские роли и права доступа

Описание: Система должна поддерживать различные пользовательские роли и права доступа. Администратор системы должен иметь возможность определять роли пользователей и назначать им соответствующие права доступа. Администратор должен иметь полный доступ ко всем функциям системы, в то время как обычные пользователи имеют ограниченные права.

Приоритет: 1

Риск: Низкий

## 3.2 Надежность

### 3.2.1 Доступность

Идентификатор требования: ДОСТУПНОЕ-ЖИЛЬЁ

Описание: Желаемый процент времени доступности системы составляет 98%. Система должна быть доступной в течение установленных рабочих часов, с возможностью доступа к обслуживанию и операции в деградированном режиме в случае сбоев.

Приоритет: 1

Риск: Высокий

### 3.2.2 Среднее время между отказами (MTBF)

Идентификатор требования: СВИДАНИЯ-ВСЛЕПУЮ

Описание: Среднее время между отказами должно быть указано в часах и составлять 5000 часов. Это помогает определить, насколько надежной является система и частоту ее сбоев.

Приоритет: 3

Риск: Низкий

### 3.2.3 Среднее время восстановления (MTTR)

Идентификатор требования: РЕХАБ

Описание: Допустимая длительность простоя системы после отказа должна составлять не более 4 часов. Это помогает определить скорость восстановления системы и минимизацию времени простоя.

Приоритет: 2

Риск: Средний

### 3.2.4 Точность

Идентификатор требования: В-ЯБЛОЧКО

Описание: требуется точность до 0.01 единицы или соответствие стандартам ISO-9001.

Приоритет: 1

Риск: Высокий

### 3.2.5 Максимальная частота ошибок или дефектов

Идентификатор требования: РУССКИЙ-ЛИБЕРАЛ

Описание: Максимальная частота ошибок должна составлять 0.1 ошибки на тысячу строк кода.

Приоритет: 2

Риск: Средний

### 3.2.6 Частота ошибок или дефектов

Идентификатор требования: БАГ-СЧЁТЧИК

Описание: Ошибки и дефекты должны быть категоризированы как незначительные, значительные и критические. Необходимо определить, что считается "критической" ошибкой, полная потеря данных или полная невозможность использования определенных частей функциональности системы.

Приоритет: 3

Риск: Низкий

### 3.3 Производительность

#### 3.3.1 Время ответа

Идентификатор требования: ОЖИДАНИЕ

Наименование: Время ответа

Описание: Среднее время ответа на запросы пользователя должно составлять не более 250 миллисекунд. Максимальное время ответа не должно превышать 1 секунду.

Приоритет: 1

Риск: Высокий

#### 3.3.2 Пропускная способность

Идентификатор требования: ПРОПУСКНОЙ

Наименование: Пропускная способность

Описание: Система должна обеспечивать обработку не менее 200 транзакций в секунду, с учетом возможных пиковых нагрузок.

Приоритет: 1

Риск: Высокий

#### 3.3.3 Емкость системы

Идентификатор требования: ТОЛПА

Наименование: Емкость системы

Описание: Максимальное количество клиентов, обслуживаемых одновременно, должно быть не менее 1000. Система должна способствовать масштабируемости, чтобы удовлетворить потребности в случае роста числа клиентов.

Приоритет: 1

Риск: Высокий

#### 3.3.4 Резервное копирование и восстановление

Идентификатор требования: НА-ВСЯКИЙ-СЛУЧАЙ

Наименование: Резервное копирование и восстановление

Описание: Должна быть обеспечена ежедневная резервная копия данных и возможность восстановления в случае сбоев в течение 4 часов.

Приоритет: 1

Риск: Высокий

#### 3.3.5 Документация и обучение

Идентификатор требования: УЧИТЕЛЬ

Наименование: Документация и обучение

Описание: Обеспечьте наличие подробной документации по работе с системой и обучение персонала для эффективного администрирования и обслуживания.

Приоритет: 1

Риск: Высокий

#### 3.3.6 Мониторинг и управление

Идентификатор требования: ВЕРТУХАЙ

Наименование: Мониторинг и управление

Описание: Предусмотрите систему мониторинга производительности и управления, которая позволит оперативно реагировать на проблемы в работе системы.

Приоритет: 1

Риск: Высокий

## 3.4 Ремонтопригодность

### 3.4.1 Структура и документация

Идентификатор требования: СПРАВКА-ПО-РЕМОНТУ

Наименование: Структура и документация

Описание: Код должен быть хорошо документирован с использованием стандартов документирования, таких как Doxygen или Javadoc. Используйте понятные и информативные комментарии и описания функций, классов и переменных. Создайте справочные документы и руководства для администраторов и обслуживающего персонала.

Приоритет: 1

Риск: Высокий

### 3.4.2 Соглашения об именах

Идентификатор требования: КОД-ПАТРУЛЬ

Наименование: Соглашения об именах

Описание: Придерживайтесь четких соглашений об именовании переменных, функций, классов и модулей кода. Используйте camelCase для переменных и PascalCase для классов.

Приоритет: 1

Риск: Высокий

### 3.4.3 Стандарты кода

Идентификатор требования: ЧИСТЫЙ-КОД

Наименование: Стандарты кода

Описание: Следуйте установленным стандартам кода, таким как PEP 8 для Python или Google Java Style для Java. Убедитесь, что код читабелен и не содержит избыточных или ненужных конструкций.

Приоритет: 1

Риск: Высокий

### 3.4.4 Использование библиотек и фреймворков

Идентификатор требования: ФУНДАМЕНТ

Наименование: Использование библиотек и фреймворков

Описание: Используйте надежные и проверенные библиотеки и фреймворки, чтобы упростить сопровождение и обеспечить стабильность системы.

Приоритет: 1

Риск: Высокий

### 3.4.5 Разделение на модули и компоненты

Идентификатор требования: КИРПИЧИ

Наименование: Разделение на модули и компоненты

Описание: Разбейте систему на модули и компоненты, что облегчит отслеживание и обслуживание различных частей системы.

Приоритет: 1

Риск: Высокий

### 3.4.6 Логирование и отладка

Идентификатор требования: ПОЧЕМУ-НЕ-РАБОТАЕТ

Наименование: Логирование и отладка

Описание: Реализуйте систему логирования, которая позволит регистрировать события, ошибки и исключения. Логи должны быть доступны для анализа и мониторинга.

Приоритет: 1  
Риск: Высокий

### **3.4.7 Резервное копирование и восстановление**

Идентификатор требования: НА-ВСЯКИЙ-СЛУЧАЙ-СИСТЕМУ  
Наименование: Резервное копирование и восстановление  
Описание: Предусмотрите механизмы автоматического резервного копирования данных и возможность их восстановления в случае сбоев.  
Приоритет: 1  
Риск: Высокий

### **3.4.8 Тестирование и контроль версий**

Идентификатор требования: МАРИОНЕТКА  
Наименование: Резервное копирование и восстановление  
Описание: Разработайте набор тестов, включая модульные, интеграционные и функциональные тесты, для обнаружения и исправления ошибок. Используйте систему контроля версий, такую как Git, для отслеживания изменений в коде и управления версиями.  
Приоритет: 1  
Риск: Высокий

## **3.5 Ограничения проекта**

### **3.5.1 Язык ПО**

Идентификатор требования: НЕ-БРАТ-ТЫ-МНЕ  
Наименование: Язык ПО  
Описание: Поддержка языков СНГ.  
Приоритет: 2  
Риск: Средний

### **3.5.2 Компоненты ПО**

Идентификатор требования: БУРЕВЕСТИК  
Наименование: Компоненты ПО  
Описание: Использовать или самописные модули, библиотеки и фреймворки.  
Приоритет: 1  
Риск: Высокий

## **3.6 Требования к пользовательской документации**

[Describes the requirements, if any, for on-line user documentation, help systems, help about notices, etc.]

## **3.7 Используемые приобретаемые компоненты**

[This section describes any purchased components to be used with the system, any applicable licensing or usage restrictions, and any associated compatibility and interoperability or interface standards.]

## **3.8 Интерфейсы**

[This section defines the interfaces that must be supported by the application. It should contain adequate specificity, protocols, ports and logical addresses, etc. so that the software can be developed and verified against the interface requirements.]

### **3.8.1 Интерфейс пользователя**

[Describe the user interfaces that are to be implemented by the software.]

### **3.8.2 Аппаратные интерфейсы**

[This section defines any hardware interfaces that are to be supported by the software, including logical structure, physical addresses, expected behaviour, etc. ]

### **3.8.3 Программные интерфейсы**

[This section describes software interfaces to other components of the software system. These may be purchased components, components reused from another application or components being developed for subsystems outside of the scope of this **SRS** but with which this software application must interact.]

### **3.8.4 Интерфейсы коммуникаций**

[Describe any communications interfaces to other systems or devices such as local area networks, remote serial devices, etc.]

## **3.9 Требования лицензирования**

[Defines any licensing enforcement requirements or other usage restriction requirements that are to be exhibited by the software.]

## **3.10 Применимые стандарты**

[This section describes by reference any applicable standard and the specific sections of any such standards which apply to the system being described. For example, this could include legal, quality and regulatory standards, industry standards for usability, interoperability, internationalization, operating system compliance, safety, security, etc.]

## **Индекс**