# Содержание

Содержание 1

История изменений 2

1 Введение 3

1.1 Цели 3

1.2 Границы применения 3

1.3 Термины, аббревиатуры, сокращения 3

1.4 Ссылки 3

1.5 Краткий обзор 3

2 Общее описание 3

2.1 Описание изделия 3

2.1.1 Интерфейсы пользователя 4

2.1.2 Интерфейсы аппаратных средств ЭВМ 4

2.1.3 Интерфейсы программного обеспечения 4

2.1.4 Интерфейсы коммуникаций 4

2.1.5 Ограничения памяти 4

2.1.6 Требования настройки рабочих мест 4

2.2 Функции изделия 4

2.3 Характеристики пользователей 5

2.4 Ограничения 5

3 Детальные требования 5

3.1 Функциональные требования 6

3.1.1 Пользователь 6

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 8

3.4.1 <Maintainability Requirement One> 8

3.5 Ограничения проекта 8

3.5.1 <Design Constraint One> 8

3.6 Требования к пользовательской документации 8

3.7 Используемые приобретаемые компоненты 8

3.8 Интерфейсы 8

3.8.1 Интерфейс пользователя 8

3.8.2 Аппаратные интерфейсы 8

3.8.3 Программные интерфейсы 8

3.8.4 Интерфейсы коммуникаций 8

3.9 Требования лицензирования 8

3.10 Применимые стандарты 8

Индекс 8

# История изменений

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Версия** | **Описание** | **Автор(ы)** |
| 2022-10-17 | 0.1 | Начальная ревизия | Козлов Алексей  Шуин Евгений  Николаев Александр |
| 2022-10-19 | 1.0 | Добавили введение и общее описание | Козлов Алексей  Шуин Евгений  Николаев Александр |
| 2022-11-11 | 1.1 | Изменили пункт 2.3, 2.1.1  Добавили пункт 3.1, 3.1.1 | Николаев Александр |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Введение

[The introduction of the **Software Requirements Specification (SRS)** should provide an overview of the entire **SRS**. It should include the purpose, scope, definitions, acronyms, abbreviations, references, and overview of the **SRS**.]

[Note: The Software Requirements Specification (**SRS**) captures the complete software requirements for the system, or a portion of the system. This document describes a typical **SRS** outline for a project using only traditional natural-language style requirements – with **no use-case modelling.**.]

[Many different arrangements of an **SRS** are possible. Refer to [IEEE830-1998] for further elaboration of these explanations, as well as other options for organizing an **SRS**.]

## Цели

Разработка системы проверки качества ТО, а также план инспекций ремонтных работ и затраченных для этого определенных ресурсов и деталей.

[Specify the purpose of this **SRS**. The **SRS** should fully describe the external behaviour of the application or subsystem identified. It also describes non-functional requirements, design constraints and other factors necessary to provide a complete and comprehensive description of the requirements for the software.]

## Границы применения

Система проверки качества позволяет отследить выполнение ТО, определить срок эксплуатации оборудования, выполнить отправку данных оператору системы, для дальнейшего проведения ТО в случае надобности ремонта, замены оборудования.

Дополнительно система может выдавать отчет о количестве проведенных ТО, затраченных материалах, запчастях и проведенных ремонтах с некоторым интервалом.

[A brief description of the software application that the **SRS** applies to; the feature or other subsystem grouping; what Use-Case model(s) it is associated with; and anything else that is affected or influenced by this document.]

## Термины, аббревиатуры, сокращения

|  |  |
| --- | --- |
| ТО | Техническое обслуживание |
| ИН | Индивидуальный номер |
| ИНА | Индивидуальный номер оборудования |
| БД | База данных |
| ЭВМ | Электронно-вычислительная машина |
| RAM | Random Access Memory |
| ПК | Персональный компьютер |
|  |  |
|  |  |

[This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the **SRS**. This information may be provided by reference to the project Glossary.]

## Ссылки

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

[This subsection should provide a complete list of all documents referenced elsewhere in the **SRS**. Each document should be identified by title, documentation number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]

## 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.

[This subsection should describe what the rest of the **SRS** contains and explain how the document is organized.]

# Общее описание

• product perspective

• product functions

• user characteristics

• constraints

• assumptions and dependencies

• requirements subsets]

## Описание изделия

Применение программы на промышленных предприятиях, имеющих оборудование, требующее регулярное прохождение ТО или замену деталей, оборудования.

### Интерфейсы пользователя

Интерфейс пользователя содержит два приложения:

* Мобильное приложение, предоставляет пользователю следующие возможности:
  + Просмотр состояния оборудования;
  + Создание отчёта.
* ПК приложение, предоставляет пользователю следующие возможности:
  + Просмотр состояния оборудования;
  + Создание отчёта;
  + Вносить изменения в БД о состоянии оборудования, необходимости ремонта/замене деталей.

### Интерфейсы аппаратных средств ЭВМ

Для работы с приложением требуется архитектура типа «клиент-сервер» топологии «Звезда».

### Интерфейсы программного обеспечения

Система включает в себя работу с БД, содержащую:

Информация о существующем оборудовании:  
 - Сроке эксплуатации  
 - Дату последнего ТО

- Необходимость ТО

Информация о проведенном ТО обслуживании может быть взята из БД для формирования отчетности.

### Интерфейсы коммуникаций

Система должна обеспечивать интерфейс взаимодействия с устройством «Принтер».

Система должна отправлять отчет по ТО на электронный адрес.

### Ограничения памяти

Система должна исправно работать на RAM с объемом ОЗУ не более 8гб.

Освобождение RAM предусматривается только во время выполнения профилактических работ**.**

### Требования настройки рабочих мест

Рабочее место должно состоять из следующих элементов:

* Компьютер;
* Клавиатура;
* Мышь;
* Установленная на компьютер БД, содержащая всю необходимую информацию об оборудовании;
* Установленное приложение отслеживания состояния оборудования и приложение создания отчётности;

## Функции изделия

- Хранение списка оборудования, материалов, запчастей, журнала ремонта и прохождения ТО

- Подсчет необходимых материалов для проведения ТО или ремонта

- Редактирование БД

- Формирование графика периодической инспекции оборудования

- Формирование отчетности по ТО и ремонтным работам

- Формирования графика будущих ТО

- Формирование графика инспекции оборудования

- Формирования списка необходимых запчастей для проведения ремонтных работ

## Характеристики пользователей

В системе имеется два типа пользователя:

* Оператор системы: специалист по работе с БД и компьютерными системами, имеющий достаточный опыт в сфере работы с ТО.
* Мастер: специалист по ТО и ремонту оборудования.

## Ограничения

К системе предъявляются следующие требования и ограничения:

* Код приложений должен быть написан на языке C++;
* В качестве БД необходимо использовать PostgreSQL;
* Система должна быстро реагировать на поломку оборудования;

# Детальные требования

Этот раздел содержит все требования к программному обеспечению, как функциональные, так и нефункциональные. Функциональные требования сгруппированы в соответствии с моделью варианта использования.

Требование обладает следующими свойствами:

**Идентификатор требования** Уникально идентифицирует требование во всех документах.

**Название** Определяет функциональную группу, к которой относится требование. Дает требованию символическое имя.

**Описание** Определение требования.

**Приоритет** Определяет порядок, в котором требования должны быть реализованы. Приоритеты обозначаются (от высшего к низшему) «1», «2» и «3». Требования приоритета 1 должны быть реализованы в первом продуктивном выпуске системы. Требования приоритета 2 и ниже подлежат специальному релиз-соглашению, которое выходит за рамки этого документа.

**Риск** Определяет риск невыполнения требования. Он показывает, как конкретное требование имеет решающее значение для системы. Существуют следующие уровни риска и связанное с ним влияние на систему, если требование не выполняется или выполняется неправильно:

Критический (C) – нарушит основной функционал системы. Система не может быть использована, если это требование не выполнено.

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

Средний (M) — повлияет на некоторые функции системы, но не на основной функционал. Систему можно использовать с некоторыми ограничениями.

Низкий (L) – систему можно использовать без ограничений, но с некоторыми обходными путями.

This section of the **SRS** should contain all the software requirements to a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements. When using use-case modelling, these requirements are captured in the Use-Cases and the applicable supplementary specifications.]

## Функциональные требования

В этом разделе описаны основные функциональные требования к системе управления проектами. Требования структурированы по функциональным областям.

[This section describes the functional requirements of the system for those requirements which are expressed in the natural language style. For many applications, this may constitute the bulk of the **SRS** Package and thought should be given to the structure of this section. This section is typically structured by feature, but alternative structures may also be appropriate, for example, structure by user or by subsystem. Functional requirements may include feature sets, capabilities, and security.

Where application development tools, such as requirements tools, modelling tools, etc., are employed to capture the functionality, this section will refer to the availability of that data, indicating the location and name of the tool that is used to capture the data.]

### Пользователь

**Идентификатор требования** Т1.01.01

**Название** Основная функциональность/пользователь

**Описание** Система должна поддерживать концепцию пользователя системы. У пользователя системы есть логин и пароль. Логин должен быть уникальным в системе. Пароль должен иметь достаточную сложность.

**Приоритет** 1

**Риск** C

**Идентификатор требования** Т1.01.02

**Название** Основная функциональность/тип пользователя

**Описание** В системе каждый пользователь имеет свой тип, который влияет на доступные ему функции.

**Приоритет** 1

**Риск** C

|  |  |
| --- | --- |
| Тип пользователя | Функции |
| Оператор системы | Просмотр списка оборудования, отслеживание состояния/ обновление/экспорт/импорт оборудования, назначение/переназначение оборудования мастеру. |
| Мастер | Просмотр списка оборудования, отслеживание состояния/ обновление/экспорт/импорт оборудования, ремонт оборудования. |

**Идентификатор требования** Т1.01.03

**Название** Основная функциональность/хранение

**Описание** Система должна хранить список всех пользователей, которым разрешено работать в системе, список всех типов пользователей в постоянном хранилище. Система должна иметь возможность хранить не менее 300 пользователей.

**Приоритет** 1

**Риск** C

**Идентификатор требования** Т1.01.04

**Название** Основная функциональность/смена пароля

**Описание** Система должна предоставить аутентифицированному пользователю возможность изменить свой пароль и сохранить изменение в БД.

**Приоритет** 1

**Риск** C

[The requirement description.]

## Надежность

[Requirements for reliability of the system should be specified here. Some suggestions follow:

• Availability—specify the percentage of time available ( xx.xx%), hours of use, maintenance access, degraded mode operations, etc.

• Mean Time Between Failures (MTBF) — this is usually specified in hours, but it could also be specified in terms of days, months or years.

• Mean Time To Repair (MTTR)—how long is the system allowed to be out of operation after it has failed?

• Accuracy—specify precision (resolution) and accuracy (by some known standard) that is required in the system’s output.

• Maximum Bugs or Defect Rate—usually expressed in terms of bugs per thousand of lines of code (bugs/KLOC) or bugs per function-point( bugs/function-point).

• Bugs or Defect Rate—categorized in terms of minor, significant, and critical bugs: the requirement(s) must define what is meant by a “critical” bug; for example, complete loss of data or a complete inability to use certain parts of the system’s functionality.]

### <Reliability Requirement One>

[The requirement description.]

## Производительность

[The system’s performance characteristics should be outlined in this section. Include specific response times. Where applicable, reference related Use Cases by name.

• response time for a transaction (average, maximum)

• throughput, for example, transactions per second

• capacity, for example, the number of customers or transactions the system can accommodate

• degradation modes (what is the acceptable mode of operation when the system has been degraded in some manner)

• resource utilization, such as memory, disk, communications, etc.

### <Performance Requirement One>

[The requirement description goes here.]

## Ремонтопригодность

[This section indicates any requirements that will enhance the maintainability of the system being built, including coding standards, naming conventions, class libraries, maintenance access, maintenance utilities.]

### <Maintainability Requirement One>

[The requirement description goes here.]

## Ограничения проекта

[This section should indicate any design constraints on the system being built. Design constraints represent design decisions that have been mandated and must be adhered to. Examples include software languages, software process requirements, prescribed use of developmental tools, architectural and design constraints, purchased components, class libraries, etc.]

### <Design Constraint One>

[The requirement description goes here.]

## Требования к пользовательской документации

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

## 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.]

## 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.]

### Интерфейс пользователя

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

### Аппаратные интерфейсы

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

### 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.]

### Интерфейсы коммуникаций

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

## Требования лицензирования

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

## 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