# 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 системы 3

2.1.2 Интерфейсы пользователя 3

2.1.3 Интерфейсы аппаратных средств ЭВМ 3

2.1.4 Интерфейсы программного обеспечения 3

2.1.5 Интерфейсы коммуникаций 3

2.1.6 Ограничения памяти 4

2.1.7 Действия 4

2.1.8 Требования настройки рабочих мест 4

2.2 Функции изделия 4

2.3 Характеристики пользователей 4

2.4 Ограничения 4

2.5 Предположения и зависимости 4

2.6 Распределение требований 4

3 Детальные требования 4

3.1 Функциональные требования 4

3.1.1 <Functional Requirement One> 5

3.2 Надежность 5

3.2.1 <Reliability Requirement One> 5

3.3 Производительность 5

3.3.1 <Performance Requirement One> 5

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Версия** | **Описание** | **Автор(ы)** |
| 2023-10-04 | 0.1 | Начальная ревизия | Филатов Максим, Никитин Кирилл, Кочетов Николай |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 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

Целью данной системы является проверка студентов в онлайн универе посредством оценки работ по практике одних студентов другими и в определении функциональных требований и характеристик системы онлайн-обучения, а также определении ожидаемого поведения системы и ограничений, которым она должна соответствовать.

Также данный документ предназначен для команды разработчиков, тестировщиков, менеджеров проекта, стейкхолдеров и других участников, связанных с планированием, разработкой, тестированием и внедрением системы онлайн-обучения.

[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

1. Изделие программного обеспечения, которое будет произведено: cистема онлайн-обучения, которая будет называться "EduPlatform".
2. Изделие программного обеспечения будет предоставлять возможность студентам и преподавателям обмениваться ресурсами для обучения и оценивать работы в рамках курсов.   
   Система позволит загружать работы студентов, автоматически выбирать рецензентов для их оценки, рассчитывать итоговые оценки и передавать результаты студентам и преподавателям.   
   Если оценки сильно различаются, система перенаправит работу на проверку преподавателем.   
   Система не будет предоставлять автоматическую оценку работ студентов без участия других студентов или преподавателей.
3. Применения определяемого программного обеспечения, включая важные преимущества, объекты и цели:   
   Система "EduPlatform" будет использоваться в качестве платформы для организации и осуществления процесса обучения, взаимодействия студентов и преподавателей, а также контроля успеваемости студентов. Преимуществами для студентов является доступность и гибкость онлайн-обучения, а для преподавателей – упрощение процесса проверки работ и обратной связи со студентами.
4. Данный документ соответствует предыдущим утверждениям и высокоуровневым спецификациям требований, определенным для системы онлайн-обучения "EduPlatform".

[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

|  |  |
| --- | --- |
| Система онлайн-обучения | Программное обеспечение, предоставляющее платформу для обучения, взаимодействия и оценки работ студентов и преподавателей в рамках курсов. |
| Студент | Пользователь системы, зарегистрированный на курсе с целью изучения и выполнения заданий. |
| Преподаватель | Пользователь системы, отвечающий за создание и управление курсами, заданиями и оценкой работ студентов. |
| Курс | Набор тем и образовательных материалов, предназначенных для обучения студентов в рамках системы. |
| Рецензент | Студент, выбранный системой для оценки работы другого студента. |
| EduPlatform | Название системы онлайн-обучения |
| СТПО | Спецификация требований программного обеспечения |
|  |  |
|  |  |

[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.  
  
Перспектива продукта заключается в том, чтобы сделать онлайн-обучение более качественным, гибким, доступным и персонализированным для студентов и преподавателей, интегрировать новые технологии и решения, постоянно развивать и совершенствовать функциональность и возможности системы в соответствии с изменяющимися потребностями пользователя и трендами образовательного процесса.  
  
**Перспективы системы:**  
1) Расширение функциональности платформы для поддержки различных типов контента, включая мультимедийные форматы, виртуальную реальность и дополненную реальность, чтобы обеспечить более интерактивные и гибкие образовательные возможности для студентов и преподавателей.

2) Использование методов искусственного интеллекта и машинного обучения для автоматического анализа и оценки выполненных работ, а также для обнаружения плагиата и фильтрации некачественных материалов.

3) Разработка мобильных приложений для платформы, доступных на разных операционных системах (iOS, Android), что позволит пользователям удобно и на ходу воспользоваться всеми возможностями системы.

4) Разработка и внедрение системы персонализированных рекомендаций для студентов, основанных на их предыдущих интересах, успехах и предпочтениях.

5) Улучшение системы оценки производительности преподавателей и качества обучения на основе рекомендаций и обратной связи от студентов.

6) Расширение взаимодействия с другими образовательными платформами и институтами для создания партнерских программ и обмена опытом.  
  
**Функции продукта:**

1) Регистрация и авторизация пользователей (студентов и преподавателей) на платформе

2) Создание и управление курсами, темами и заданиями преподавателями.

3) Выбор и просмотр курсов и тем студентами.

4) Загрузка выполненных работ студентами.

5) Автоматический выбор рецензентов для оценки работ.

6) Оценивание работ студентами (рецензентами) по заданной шкале (например, пятибалльной).

7) Расчёт итоговой оценки системой с учетом оценок рецензентов.

8) Передача работы преподавателю для оценивания в случае неоднозначных оценок.

9) Отображение результатов оценок студентам и преподавателям.

10) Формирование статистики и аналитики по курсам, студентам и преподавателям.

**Пользовательские характеристики:**

1) Студенты - основные пользователи системы, их функции включают выбор курсов, выполнение заданий и оценку работ других студентов.

2) Преподаватели - управляют курсами, темами и заданиями, проверяют работы в случаях неоднозначных оценок, просматривают статистику и аналитику по своим курсам и студентам.  
  
**Ограничения целостности:**

1) Механизмы проверки на плагиат должны быть встроены в систему для повышения качества предоставляемых работ.

2) Учетная запись преподавателя должна быть связана с одним или несколькими курсами, а учетная запись студента должна быть связана с выбранными студентом курсами.  
  
**Допущения и зависимости:**

1) Студенты и преподаватели имеют стабильный интернет-доступ для использования системы.

2) Все пользователи считаться компетентными в работе с компьютером и интернет-технологиями, имеющими достаточное знание темы для выполнения или оценки заданий.

**Подмножества требований:**

1) Возможность импорта и экспорта данных по курсам, студентам и оценкам

2) Интеграция с ресурсами образовательных материалов, видео и текстовых форматов.

3) Возможность коммуникации между студентами и преподавателями в рамках системы через личные сообщения или обсуждение тем на форуме.

4) Поддержка на платформе системы обратной связи для решения возникающих вопросов и конфликтных ситуаций.

5) Возможность интеграции со сторонними сервисами для авторизации (например, Google, UNN или VK Group).

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

# Общее описание

**Описание изделия:**

Система онлайн-обучения "EduPlatform" является веб-приложением, предназначенным для организации, проведения и управления курсами, позволяет студентам и преподавателям взаимодействовать, обмениваться знаниями и оценивать выполненные работы.

**Функции изделия**:

1. Регистрация и авторизация пользователей.

2. Создание, редактирование и управление курсами, темами и заданиями преподавателями.

3. Выбор, поиск и просмотр курсов и тем студентами.

4. Загрузка выполненных работ студентами.

5. Автоматический выбор рецензентов для оценки работ.

6. Оценивание работ студентами (рецензентами).

7. Рассчет итоговых оценок системой.

8. Передача работы преподавателю для оценивания в случае расхождения между оценками.

9. Отображение результатов оценок студентам и преподавателям.

10. Формирование статистики и аналитики по курсам, студентам и преподавателям.

**Характеристики пользователей:**

1. Студенты - основные пользователи платформы, взаимодействуют с курсами, темами, выполняют и оценивают работы.

2. Преподаватели - создают и управляют курсами, темами, заданиями, оценивают работы и контролируют образовательный процесс.

**Ограничения:**

1. Система должна быть совместима с различными браузерами и операционными системами.

2. Система должна поддерживать аутентификацию и авторизацию пользователей безопасным способом.

3. Система должна иметь защиту от несанкционированного доступа и атак.

**Предположения и зависимости:**

1. Все пользователи имеют стабильный доступ в интернет.

2. Все пользователи имеют достаточные знания и опыт для пользования системой.

3. Возможное изменение законодательных актов или технологических требований, влияющих на систему.

**Поднаборы требований:**

1. Возможность работы с различными типами контента (текст, изображения, видео, презентации).

2. Механизмы для обнаружения плагиата и проверки качества предоставленных работ.

3. Возможность масштабирования системы для удовлетворения возрастающих потребностей пользователей.

4. Интеграция с сервисами и системами сторонних разработчиков API или авторизации

5. Поддержка мобильных устройств, включая создание соответствующих мобильных приложений.

• product functions

• user characteristics

• constraints

• assumptions and dependencies

• requirements subsets]

## Описание изделия

Система "EduPlatform" является независимым и полностью автономным продуктом, предназначенным для онлайн-обучения студентов и взаимодействия студентов с преподавателями. Однако, данная система может быть впоследствии расширена и интегрирована с другими продуктами, связанными с онлайн-обучением и образовательными платформами.

**Интерфейсы системы:**

Система будет взаимодействовать с веб-сервером и базой данных для хранения информации о студентах, преподавателях, курсах и работах студентов.

**Интерфейсы пользователя:**

Пользовательский интерфейс будет представлен в виде веб-приложения, доступного через браузер и адаптированного для различных устройств (компьютеры, планшеты, смартфоны).

**Интерфейсы аппаратных средств ЭВМ:**

Система будет работать на любых устройствах, поддерживающих современные веб-браузеры и имеющих доступ в Интернет.

**Интерфейсы программного обеспечения:**

Система "EduPlatform" будет взаимодействовать с серверной частью, реализованной на основе веб-сервера и базы данных. Для разработки клиентской части могут быть использованы современные фреймворки и библиотеки.

**Интерфейсы коммуникаций:**

Система будет использовать протоколы HTTP/HTTPS для передачи данных между клиентом и сервером.

**Ограничения памяти:**

Требования к памяти зависят от количества пользователей, курсов и работ, которое система должна обрабатывать и хранить. Однако, современные технологии облачных хранилищ и оптимизации баз данных помогут снизить нагрузку на память.

**Функционирование (Operations):**

Система должна обеспечивать непрерывное и стабильное функционирование для всех пользователей. Возможны периодические технические работы для обновления и обслуживания системы.

**Требования настройки рабочих мест:**

Система будет работать на любых устройствах с поддержкой современных веб-браузеров и доступом в Интернет, без дополнительных требований к установке специального программного обеспечения.

### Интерфейсы системы

**1. Веб-сервер:** ответственный за обработку запросов от клиентов и обеспечение взаимодействия с базой данных. Веб-сервер будет реализован на основе современных технологий, таких как Node.js, Ruby on Rails, Django или PHP.

**2. База данных:** Ответственная за хранение информации о студентах, преподавателях, курсах и работах студентов. Будет использоваться реляционная или нереляционная база данных в зависимости от предпочтений разработчиков и требований системы (например, MySQL, PostgreSQL, MongoDB).

**3. API (Application Programming Interface):** предоставляет стандартизованный набор методов для обращения клиентской части системы к серверной части. API будет разработано с использованием RESTful принципов и обеспечит интеграцию между клиентом и сервером через JSON формат.

**4. Клиентская часть системы:** веб-приложение, доступное через браузер и отвечающее за взаимодействие студентов и преподавателей с системой. Клиентская часть будет разработана с использованием современных технологий (React, Angular, Vue.js) и будет адаптирована для различных устройств (компьютеры, планшеты, смартфоны).

**5. Сервис аутентификации:** отвечает за проверку идентификационных данных пользователей и обеспечивает безопасность системы. Сервис аутентификации может использовать стандартные протоколы, такие как OAuth 2.0, для интеграции с внешними сервисами (например, Google, UNN, VK Group).

**6. Сервис уведомлений:** отвечает за информирование студентов и преподавателей о новых событиях, связанных с курсами, работами и оценками. Уведомления могут быть отправлены по электронной почте, внутри системы или через мобильное приложение.

Каждый из упомянутых интерфейсов должен быть проектирован и реализован с применением лучших практик разработки программного обеспечения и учитывая требования системы, пользователей и бизнес-логики. Взаимодействие между интерфейсами должно быть оптимизировано для обеспечения высокой производительности, масштабируемости и стабильности системы.

### Интерфейсы пользователя Здесь необходимо определить следующие характеристики пользовательских интерфейсов:

**a) Логические характеристики каждого интерфейса:**

- Адаптивный дизайн: интерфейс должен корректно отображаться на различных устройствах (компьютеры, планшеты, смартфоны) и разрешениях экрана.

- Интуитивная навигация: меню, кнопки, и ссылки должны быть легко доступными и понятными для пользователей без дополнительных объяснений (например, использование иконок с подписями).

- Вывод информации: система должна предоставлять информацию о курсах, работах студентов и оценках в удобном и структурированном виде (например, таблицы, списки, графики).

- Сообщения об ошибках и прогрессе: система должна информировать пользователей о возможных ошибках (например, неправильно заполненные поля или сбои) и о статусе обработки запросов (например, индикация загрузки).

**b) Аспекты оптимизации интерфейса:**

- Минимизация количества шагов для выполнения задач: интерфейс должен быть разработан таким образом, чтобы пользователи могли выполнять свои задачи за минимальное количество шагов и времени.

- Дружественность к пользователю: интерфейс должен быть привлекательным, удобным, и разработанным с учетом особенностей целевой аудитории (например, возраст, опыт использования подобных систем).

- Доступность функций: необходимые функции должны быть легко доступными и понятными для пользователей. При возникновении ошибок система должна предоставлять краткие и полные объяснения об ошибках, чтобы помочь пользователям устранить проблему без затруднений.

- Обучаемость: пользователи должны быстро осваивать систему. Например, студент должен способен загрузить и отредактировать свою работу за 10 минут после 30 минут обучения.

Таким образом, интерфейсы пользователя системы "EduPlatform" должны быть разработаны с учетом требований и ожиданий пользователей, а также с последующей проверкой и анализом пользовательского опыта для оптимизации взаимодействия с системой.

### Интерфейсы аппаратных средств ЭВМ

Для корректной работы системы "EduPlatform" необходимо определить логические характеристики каждого взаимодействия между программным обеспечением и аппаратными компонентами системы. В данном случае, поскольку система представляет собой веб-приложение, основные требования к аппаратным интерфейсам направлены на поддержку работы веб-браузеров и подключение к Интернету.

**1. Поддержка веб-браузеров:** система должна корректно работать в современных веб-браузерах, таких как Google Chrome, Mozilla Firefox, Safari, и Microsoft Edge. Браузеры должны поддерживать последние стандарты и технологии веб-разработки, такие как HTML, CSS, и JavaScript.

**2. Интернет-соединение:** для доступа к системе необходимо наличие стабильного интернет-соединения со скоростью, достаточной для загрузки веб-страниц, обработки запросов, и передачи файлов.

**3. Распределение нагрузки и оптимизация производительности:** серверная часть системы должна быть оптимизирована для работы с различными аппаратными решениями, включая облачные сервера и по возможности адаптироваться к изменяемой нагрузке со стороны пользователей.

**4. Интеграция с аппаратными компонентами:** веб-приложение должно корректно работать с различными устройствами ввода (клавиатура, мышь, сенсорный экран) и воспроизведения мультимедиа (аудио и видео).

**5. Поддержка мобильных устройств:** система должна быть адаптирована для работы на мобильных устройствах, таких как смартфоны и планшеты, с различными разрешениями экрана и операционными системами (iOS, Android).

В целом, интерфейсы аппаратных средств ЭВМ для системы "EduPlatform" должны быть максимально адаптированы к современным технологиям и устройствам пользователей для обеспечения качественного функционирования системы и удобства ее использования.

### Интерфейсы программного обеспечения

Для корректной работы и взаимодействия системы "EduPlatform" с другими программными компонентами, необходимо определить следующие интерфейсы программного обеспечения:

**1. Система управления базами данных** (Database Management System, DBMS):

a) Наименование: MySQL / PostgreSQL / MongoDB (в зависимости от выбранной базы данных)

b) Мнемоническое наименование: DB

c) Номер спецификации: зависит от версии и документации выбранной DBMS

d) Номер версии: ссылаться на последнюю стабильную версию выбранной DBMS

e) Источник: официальный сайт выбранной DBMS (например, mysql.com, postgresql.org, mongodb.com)

**2. Веб-сервер:**

a) Наименование: Node.js / Ruby on Rails / Django / PHP (в зависимости от выбранной технологии серверной части)

b) Мнемоническое наименование: WebServer

c) Номер спецификации: зависит от версии и документации выбранного веб-сервера

d) Номер версии: ссылаться на последнюю стабильную версию выбранного веб-сервера

e) Источник: Официальный сайт выбранного веб-сервера (например, nodejs.org, rubyonrails.org, djangoproject.com, php.net)

**3. Фреймворк клиентской части:**

a) Наименование: React / Angular / Vue.js (в зависимости от выбранной технологии клиентской части)

b) Мнемоническое наименование: FrontendFramework

c) Номер спецификации: зависит от версии и документации выбранного фреймворка

d) Номер версии: ссылаться на последнюю стабильную версию выбранного фреймворка

e) Источник: Официальный сайт выбранного фреймворка (например, reactjs.org, angular.io, vuejs.org)

Для каждого интерфейса:

**a) Назначение взаимодействующих программ:**

- DBMS: Обеспечение хранения, обработки и доступа к данным системы.

- WebServer: Обработка запросов клиентской части, обеспечение взаимодействия с базой данных и другими сервисами.

- FrontendFramework: Реализация пользовательского интерфейса и взаимодействие с серверной частью через API.

**b) Определение интерфейса в терминах содержания и формата сообщений:**

- DBMS: SQL-запросы для реляционных баз данных (MySQL, PostgreSQL) или JSON-запросы для нереляционных баз данных (MongoDB).

- WebServer: RESTful API с использованием JSON для обмена данными между клиентской и серверной частями.

- FrontendFramework: HTTP/HTTPS запросы к API серверной части и обработка полученных данных для отображения пользователю.

Таким образом, система "EduPlatform" предполагает интеграцию и взаимодействие с рядом сторонних программных компонентов, обеспечивающих ее корректную работу и удобство использования.

### Интерфейсы коммуникаций

В системе "EduPlatform" используются следующие интерфейсы коммуникаций и протоколы для поддержания взаимодействия между компонентами системы и обеспечения ее корректной работы:

**1. Протокол HTTP/HTTPS:** используется для передачи данных между клиентской частью (веб-браузер пользователей) и серверной частью системы (веб-сервер, API). Протокол HTTPS обеспечивает шифрование данных и безопасность взаимодействия.

**2. WebSocket:** дополнительно может использоваться для обеспечения двунаправленного общения между клиентом и сервером в реальном времени, например, для функций чата или обновлений информации без необходимости перезагрузки страницы.

**3. Протоколы сетевого уровня (TCP/IP):** используются для обеспечения обмена данными между серверами, хостами и другими сетевыми устройствами, а также для поддержания стабильной работы системы.

### Ограничения памяти

Ограничения памяти для системы "EduPlatform" зависят от количества пользователей, курсов, работ и других данных, которые система должна обрабатывать и хранить. В качестве ориентиров для определения ограничений памяти, могут быть использованы следующие параметры:

**1. Оперативная память:** необходимый объем оперативной памяти (RAM) для сервера зависит от нагрузки и масштабируемости системы. Рекомендуется мониторить использование памяти и при необходимости увеличивать ее объем для обеспечения стабильности работы системы.

**2. Постоянная память:** размер требуемого дискового пространства для базы данных и хранения пользовательских данных (файлы, изображения) зависит от количества зарегистрированных пользователей, созданных курсов, и загруженных работ. Рекомендуется использовать технологии сжатия данных, оптимизации баз данных и облачные хранилища для экономии пространства и обеспечения доступности данных.

**3. Память клиентского устройства:** система должна быть оптимизирована для работы на устройствах с различным объемом оперативной памяти и дискового пространства. Клиентская часть системы должна корректно работать на устройствах с низким объемом памяти, минимизировать использование памяти и не создавать наращивания данных за счет кеширования или частичной загрузки информации при навигации.

Таким образом, система "EduPlatform" должна быть разработана с учетом возможных ограничений памяти и адаптирована к различным условиям использования для обеспечения стабильности и производительности.

### Действия

В "EduPlatform" предусмотрены следующие типы действий:

**a) Различные способы действий в организации пользователя:**

- загрузка работ студентами

- выбор студентов-рецензентов системой

- оценка работ рецензентами

- расчет итоговой оценки системой

- перенаправление работы для проверки преподавателю, если оценки сильно различаются

- получение оценок и отзывов студентами

- мониторинг успеваемости и предоставление отчетов преподавателям

**b) Периоды диалоговых действий и периоды оставленных без отклика действий:**

- периоды активного взаимодействия пользователей с системой (при работе с оценками и отзывами)

- периоды ожидания обработки данных системой (при расчете итоговой оценки, подборе рецензентов и отправке работы на проверку преподавателю)

**c) Функции поддержки обработки данных:**

- хранение загруженных работ, оценок и отзывов

- статистический анализ данных для определения средних оценок, распределения оценок и выявления аномалий

- создание резюме результатов для студентов и преподавателей

**d) Действия резервного копирования и восстановления:**

- автоматическое резервное копирование данных о загруженных работах, оценках и отзывах

- возможность восстановления данных в случае потери или повреждения

- создание архивных копий результатов текущего периода обучения для использования в будущем

Таким образом, "EduPlatform" будет выполнены действия, связанные с процессами оценивания работ студентами, их анализом и предоставлением результатов обратной связи пользователям. Кроме того, система обеспечивает безопасность и сохранность данных, что является одной из важных функций для успешной работы образовательной платформы.

### Требования настройки рабочих мест

a) Для корректной работы "EduPlatform" требуется определить следующие требования:

- разрешение экрана: оптимальное разрешение экрана для комфортной работы с системой должно быть не менее 1280x720 пикселей (телефоны, планшеты, компьютеры, ноутбуки)

- используемый браузер: обеспечить совместимость системы с наиболее распространенными браузерами (Google Chrome, Mozilla Firefox, Safari, Microsoft Edge)

- скорость интернет-соединения: минимальная рекомендуемая скорость интернет-соединения должна быть не менее 1 Mbps для стабильной работы системы

b) Характеристики рабочего места или решаемых задач, которые должны быть изменены для настройки программного обеспечения на специальную конфигурацию:

- операционная система: убедиться, что рабочие места пользователей оснащены поддерживаемыми операционными системами (Windows, macOS, Linux)

- антивирусное ПО и файервол: настройка защитных систем таким образом, чтобы разрешить обмен данными между рабочими местами пользователей и системой "EduPlatform"

- программное обеспечение для просмотра и редактирования документов типов, используемых в "EduPlatform" (например, Microsoft Office, Adobe Acrobat Reader или их аналоги)

- настройка прокси-сервера или VPN соединения для безопасного доступа к системе, если это необходимо это для выполнения корпоративных политик безопасности

В целом, настройка рабочих мест пользователей "EduPlatform" должна быть сфокусирована на обеспечении комфортных условий для работы с системой, поддержании совместимости программного обеспечения и обеспечении надежной и безопасной связи с платформой.

## Функции изделия

В этом разделе представлены основные функции, выполняемые программным обеспечением "EduPlatform". Функции организованы в группы для обеспечения понятности структуры.

**1. Управление курсами и пользователями:**

a) Создание и редактирование курсов и учебных модулей.

b) Управление доступом студентов и преподавателей к курсам.

c) Мониторинг успеваемости студентов и предоставление отчетов.

**2. Загрузка и хранение работ студентов:**

a) Предоставление пользовательского интерфейса для загрузки работ студентами.

b) Хранение загруженных работ в облачном хранилище.

c) Поддержка различных типов файлов для загружаемых работ.

**3. Оценивание работ студентами и системой:**

a) Автоматический выбор рецензентов для оценки работ.

b) Предоставление интерфейса для студентов-рецензентов для оценки и предоставления отзывов.

c) Расчет итоговой оценки на основе оценок рецензентов и определение дальнейшего действия (например, пересылка преподавателю в случае сильно различающихся оценок).

**4. Обратная связь и коммуникация:**

a) Получение оценок и отзывов студентами.

b) Функция обсуждения работ и отзывов между студентами и преподавателями.

c) Возможность получения поддержки и решения технических проблем.

Таким образом, "EduPlatform" управляет процессом обучения и взаимодействия студентов и преподавателей, организует проверку и оценку работ студентов, обеспечивает хранение данных и позволяет обмениваться отзывами для достижения успеха в обучении.

## Характеристики пользователей

Рассмотрим общие характеристики пользователей, для которых предназначена "EduPlatform":

**1. Студенты:**

- Образовательный уровень: студенты могут иметь различный образовательный уровень, включая среднее, среднее специальное, высшее и послевузовское образование.

- Опыт: студенты могут не иметь опыта работы в своей области или наоборот, обладать определенным профессиональным опытом.

- Техническая квалификация: студенты должны обладать базовыми знаниями и навыками в работе с компьютером, интернетом и офисными приложениями.

**2. Преподаватели:**

- Образовательный уровень: преподаватели должны иметь высшее образование, а также возможно, научные степени и звания.

- Опыт: преподаватели должны обладать опытом работы в своей области, а также опытом преподавания и наставничества.

- Техническая квалификация: преподаватели должны иметь продвинутые навыки в работе с компьютером, интернетом и офисными приложениями, а также опыт использования образовательных платформ.

Определение характеристик пользователей помогает обеспечить наиболее удобный и доступный интерфейс, а также оптимальный набор функциональных возможностей "EduPlatform". Эти характеристики будут влиять на требования к системе и на формирование функций, рассмотренных в разделе 3 СТПО.

## Ограничения

В этом разделе приведем общее описание факторов, ограничивающих выбор разработчика для "EduPlatform":

**a) Регулирующая политика:** соблюдение правил и норм местного законодательства, а также правил и политик, принятых в образовательных учреждениях, которые будут использовать систему.

**b) Ограничения аппаратных средств ЭВМ:** система должна быть готова к работе на различных устройствах (ПК, мобильные устройства) с определенными минимальными и рекомендуемыми системными требованиями.

**c) Интерфейсы с другими приложениями:** "EduPlatform" должна обеспечивать интеграцию с внешними системами, такими как системы управления базами данных, системы аутентификации пользователей, а также популярные службы облачного хранения.

**d) Параллельная работа:** система должна поддерживать одновременную работу нескольких пользователей с обеспечением надежного доступа и сохранности данных.

**e) Функции протоколирования:** обеспечить регистрацию действий пользователей и системных событий для контроля и анализа работы системы.

**f) Функции управления:** наличие административных функций для управления и контроля доступа.

**g) Требования к языкам высокого уровня:** система должна быть разработана на современных и распространенных языках программирования, например, Python, Java или JavaScript.

**h) Протоколы интерфейсов синхронизации сигналов:** система должна поддерживать соответствующие протоколы для обеспечения стабильного пересылки данных и коммуникаций между пользователями и сервером.

**i) Требования надежности:** "EduPlatform" должна обеспечивать высокую надежность и доступность для пользователей, а также устойчивость к различным типам сбоев.

**j) Критичность приложения:** система должна поддерживать высокие стандарты, потому что образовательный процесс и карьера студентов зависят от качества работы "EduPlatform".

**k) Соображения безопасности и секретности:** обеспечение защиты и конфиденциальности персональных данных пользователей, а также интеллектуальной собственности, связанной с контентом курсов и проверяемыми работами.

## Предположения и зависимости

В этом разделе определены факторы, которые влияют на требования "EduPlatform", но не являются непосредственными ограничениями. Изменения в этих факторах могут повлиять на требования, представленные в документе СТПО.

**1. Операционная система:** предполагается, что на аппаратных средствах ЭВМ пользователей будет установлена совместимая операционная система (Windows, macOS, Linux). Если доступна другая операционная система, требования к "EduPlatform" могут измениться.

**2. Сетевая инфраструктура:** предполагается наличие стабильного доступа пользователей к интернету для работы с "EduPlatform". Изменение качества или доступности интернет-соединения может повлиять на функциональность системы и удовлетворение требований пользователей.

**3. Технологические инструменты:** предполагается использование современных инструментов разработки программного обеспечения, фреймворков и библиотек, которые обеспечивают быстрое и качественное развертывание "EduPlatform". Обновление или замена используемых технологий может изменить требования к системе.

**4. Стандарты обучения и нормативные документы:** система основана на существующих стандартах обучения и нормативных документах, которые определяют образовательные процессы. Изменения в этих стандартах и документах могут повергнуть требования к "EduPlatform".

**5. Интеграция с внешними сервисами:** "EduPlatform" предполагает возможность интеграции с различными внешними сервисами, например, системами облачного хранения или аутентификации пользователей. Изменения в доступности или спецификациях этих сервисов могут влиять на требования к системе.

Оценка предположений и зависимостей позволяет адаптироваться к возможным изменениям, а также предоставляет возможность для пересмотра и корректировки требований к "EduPlatform" при необходимости.

## Распределение требований

В данном разделе приводятся требования, реализация которых может быть отложена до будущих версий системы "EduPlatform".

**1. Мобильное приложение:** хотя основной функционал "EduPlatform" должен быть расположен в веб-приложении, доступном через интернет-браузер, в последующих версиях можно разработать полноценное мобильное приложение для улучшения доступности и удобства использования системы на мобильных устройствах.

**2. Интеграция с онлайн-сервисами для совместной работы:** в будущих версиях системы можно добавить возможность интеграции с популярными онлайн-сервисами для совместной работы над документами или проектами, например, Google Docs или Microsoft Office 365.

**3. Расширение функционала обратной связи:** добавление возможностей видеоконференции, текстового и голосового чатов, чтобы студенты и преподаватели могли обмениваться мнениями и знаниями в режиме реального времени.

**4. Встроенные учебные ресурсы и курсы:** в дополнение к создаваемым академическими модулям, "EduPlatform" может предоставить доступ к каталогу внешних образовательных ресурсов и курсов, например, интеграция с ресурсами открытого образования.

**5. Система уведомлений:** реализация сложной системы уведомлений для пользователей о новых событиях или действиях, связанных с образовательным процессом и их курсами.

**6. Поддержка нескольких языков:** возможность использования "EduPlatform" пользователями, говорящими на различных языках с добавлением переведенного интерфейса и контента.

Учет возможности распределения требований позволяет разработчикам системы сфокусироваться на реализации ключевых функций в первой версии "EduPlatform", а также планировать долгосрочное развитие и улучшение продукта.

# Детальные требования

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

[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.]

### <Functional Requirement One>

[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