МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «КубГУ»)

Факультет компьютерных технологий и прикладной математики  
Кафедра информационных технологий

**ОТЧЁТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЕ №3 ПО ДИСЦИПЛИНЕ «ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»**

Работу выполнила студентка 4ИТ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. А. Арабова

Проверил доцент кафедры ИТ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Н.Полетайкин

Краснодар

2023

Тема: Техническое задание на создание программного продукта.

Цель: Освоение методики предварительного анализа разрабатываемой программы; освоение задач формулирования функциональных и нефункциональных требований к программной реализации отдельных задач и к программе в целом; выработка навыков разработки технического задания.

Задание

1. Установить назначение и общую цель создания программы.

2. Определить структуру программы и состав функциональных задач.

3. Разработать функциональные требования к программе:

* требования к входным и выходным данным;
* требования к программной реализации задач;
* специальные требования к математическому обеспечению программной
* реализации задач;

4. Разработать модель требований в нотации UML

5. Разработать требования к информационному обеспечению (к базе данных).

6. Разработать требования к инструментальному программному обеспечению

(к системе управления базой данных (СУБД), к средству разработки

программ (IDE), средствам автоматизированного проектирования ПО)

7. Установить нефункциональные требования к программе. Дать не менее 5

наиболее очевидных для данной системы требований из перечня:

* требования к надежности;
* требования к эффективности;
* требования к безопасности;
* требования к эргономичности и удобству использования;
* требования к численности и квалификации персонала и режиму работы;
* требования к переносимости;
* требования к сопровождаемости;
* требования к особенностям поставки;
* требования к защите информации от несанкционированного доступа;
* требования по сохранению информации при авариях;
* требования к соответствию стандартам качества.

Индивидуальная тема: Сервис проверки корректности технической документации.

1. **Назначение и цель создания сервисa проверки технической документации**

Сервис проверки корректности технической документации – программа для проверки структуры технической документации и выявлении несоответствий с ГОСТом. Сервис предполагается для автоматизирования процесса проверки необходимых файлов.

Целью создания данного сервиса является минимизировать человеческий труд, тем самым уменьшив время проверки и улучшив ее качество.

1. **Разрабатываемый сервис должен решать такие задачи:**

* задача корректного составления структуры шаблона,
* задача проверки содержания технической документации,
* задача выделения ошибок,
* задача визуализации результата

**3.Функциональные требования**

**Требования к задаче «Задача корректного составления структуры шаблона»**

1. Шаблон должен быть составлен по ГОСТу
2. Шаблон должен содержать все обязательные страницы технической документации

**Требования к задаче «Задача проверки написанного текста»**

1. Написанный текст в содержании должен соответствовать ГОСТу и содержать обязательные пункты и подпункты

**Требования к задаче «Задача выделения ошибок»**

1. Найденные ошибки в содержании, нехватке обязательных страниц и прочие несоответствия должны быть выделены

**Требования к задаче «Задача визуализации результата»**

1. В консоли написанного сервиса должны быть выведены ошибки, найденные в загруженном файле технической документации

**Диаграмма требований (вариантов использования) UML представлена на рисунке 1:**

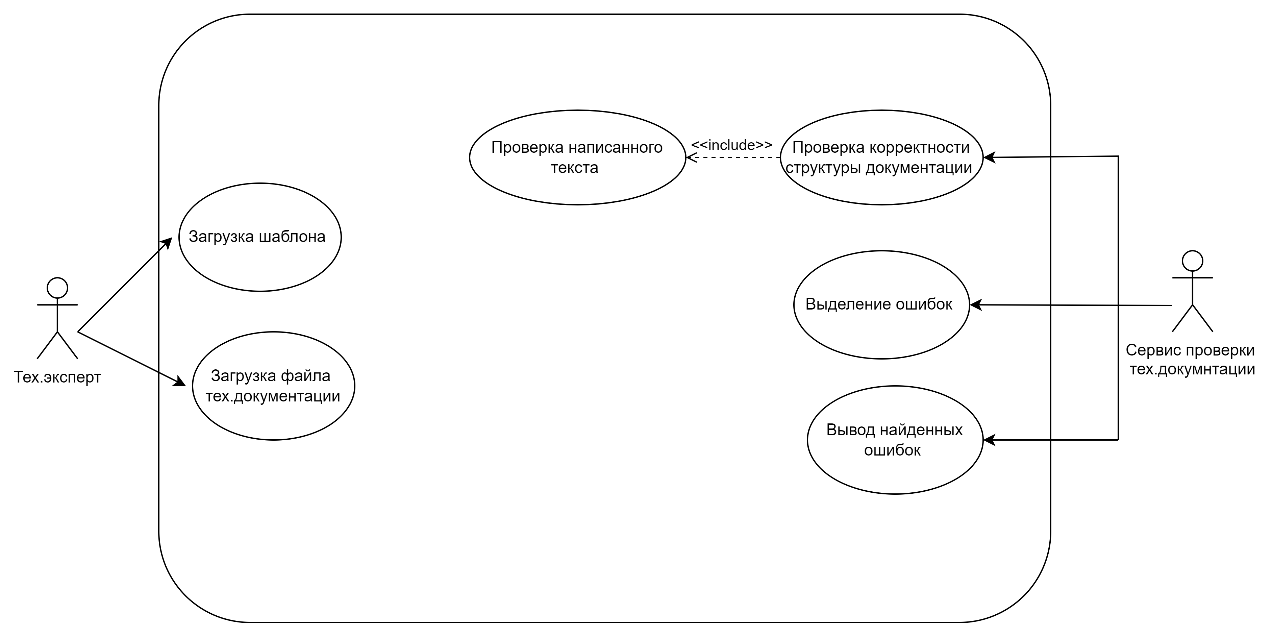


Рисунок 1 – Модель требований в нотации UML

**Требования к информационному обеспечению**

1. Установленный интерпретатор Python
2. Библиотеки Python: `tkinter`, `pathlib`, `docx`, `docx2txt`.
3. Файлы DOCX : техническая документация и файл-шаблон

**Требования к инструментальному программному обеспечению:**

Требования к операционной среде:

Необходимо обеспечить поддержку операционной системы Windows 10 и новее.

Требования к инструментальным средствам разработки ПО:

Язык программирования для создания приложения: Python

**Нефункциональные требования к ПО:**

1. Сервис должен быстро выполнять задачи по анализу и сравнению документов, вне зависимости от размеров файлов.
2. Сервис должен надежно анализировать и сравнивать документы, не вызывая сбоев или ошибок.
3. Интерфейс пользователя должен быть простым и понятным, обеспечивая удобный способ выбора файлов для сравнения и анализа.
4. Сервис должен эффективно использовать ресурсы, чтобы снизить энергопотребление и уменьшить воздействие на окружающую среду.
5. система должна иметь возможность модификации обеспечивающей части системы.