**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**

**Отчет о выполнении лабораторной работы №3**

**по дисциплине**

**«Технологии проектирования программного обеспечения»**

Направленность Информационные технологии

Работу выполнил студент группы 4ИТ/2 И.А. Молчанов

(подпись)

Работу проверил доц. каф. ИТ, к.т.н., доц. А.Н. Полетайкин

(подпись)

Краснодар

2023

**ВВЕДЕНИЕ**

**Тема работы**: Техническое задание на создание программного продукта.

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

Для выполнения данный работы были поставлены следующие **задачи**:

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 программной̆ реализации задач;
* специальные требования к математическому обеспечению программной̆ реализации задач;

1. Разработать модель требований в нотации UML
2. Разработать требования к информационному обеспечению (к базе данных).
3. Разработать требования к инструментальному программному обеспечению (к системе управления базой̆ данных (СУБД), к средству разработки программ (IDE), средствам автоматизированного проектирования программного обеспечения)
4. Установить нефункциональные требования к программе:

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

**Индивидуальная тема**: Техническое задание на создание программного продукта для анализа новостных сообщений на признак фейка.

* 1. **Назначение подсистемы и цели её создания**

Назначение системы анализа новостных сообщений на признак фейка заключается в определении достоверности информации, представленной в новостных статьях или сообщениях. Основная цель создания такой системы - борьба с распространением дезинформации и фейковых новостей.

Система анализа новостных сообщений на признак фейка имеет несколько целей:

1. Выявление фейковых новостей: система помогает автоматически обнаруживать новостные сообщения, содержащие ложную или искаженную информацию. Она анализирует различные аспекты сообщения, такие как источник, стиль письма, фактическую информацию и сравнивает ее с известными достоверными источниками данных. Это позволяет идентифицировать потенциальные фейковые новости и предупреждать пользователей о возможных манипуляциях информацией.
2. Предотвращение распространения дезинформации: система помогает предотвратить распространение фейковых новостей, предупреждая пользователей об их недостоверности. Это может быть особенно полезно в социальных сетях и онлайн-платформах, где фейковые новости могут быстро распространяться и влиять на общественное мнение. Путем предоставления пользователю информации о недостоверности определенной новости система помогает повысить осведомленность и критическое мышление.
3. Поддержка журналистов и факт-чекеров: система анализа новостных сообщений на признак фейка может быть полезной для журналистов и факт-чекеров, облегчая их работу по проверке достоверности информации. Автоматический анализ может помочь выявить потенциальные фейковые новости и предложить дополнительные ресурсы для проверки фактов, что помогает сократить время и усилия, затрачиваемые на ручную проверку каждого сообщения.
4. Улучшение информационной грамотности: система также может способствовать повышению информационной грамотности пользователей, помогая им развивать критическое мышление и умение анализировать достоверность новостей. Предоставление информации о фейковых новостях и методах их идентификации может помочь пользователям развить навыки отбора достоверной информации и более осознанно оценивать новости.

* 1. **Определение структуры программы и состава функциональных задач**

Задачи, программную реализацию которых предполагается осуществить, представлены в таблице 1.

|  |  |
| --- | --- |
| **Задача** | **Описание** |
| Сбор данных | Сбор новостных сообщений из различных источников, таких как новостные веб-сайты, социальные сети, блоги. |
| Предварительная обработка данных | Выполнение предварительной обработки данных, которая включает в себя очистку текста от лишних символов, удаление стоп-слов, приведение текста к нормализованному виду. |
| Анализ и классификация | Основной анализ новостных сообщений на признак фейка с помощью нейронной сети. |
| Проверка достоверности источников | Проверка достоверности источников новостей. Здесь может быть использована база данных достоверных и недостоверных источников. |
| Вывод результатов | Отображение результатов анализа пользователю. |

Таблица 1 – Структура программы и состав её функциональных задач

* 1. **Функциональные требования**

**3.1 Требования к входным и выходным данным**

**3.1.1. Требования к задаче «Сбор данных»**

Источники, из которых будут собираться новостные сообщения: новостные веб-сайты, социальные сети, блоги, форумы. Требуется составить список релевантных источников, которые будут использоваться для сбора данных.

**3.1.2. Требования к задаче «Предварительная обработка данных»**

Разработка процедур для удаления или корректировки неточностей, ошибок. Например, удаление дубликатов, заполнение пропущенных значений, исправление ошибок формата, проведение лемматизации и нормализации, чтобы гарантировать качество данных.

**3.1.3. Требования к задаче «Анализ и классификация»**

Необходимо определить значимые признаки, которые будут использоваться для классификации данных. Это может включать выбор и фиксацию признаков, создание новых признаков на основе имеющихся данных и другие методы инженерии признаков, чтобы максимизировать информативность и разделяющую способность признаков. Среди таких методов могут использоваться: Bag of Words (мешок слов), токенизация, метод наивного Байеса. Также произвести процесс обучения нейронной сети на обучающих данных. Необходимо провести: разделение данных на обучающую и тестовую выборки, настройку параметров модели, использование методов кросс-валидации для оценки качества модели.

**3.1.4. Требования к задаче «Проверка достоверности источников»**

Необходимо создать механизм для сбора информации об источниках данных. Это может включать сбор метаданных, таких как автор, организация, дата публикации и другие сведения, а также описание источника, его репутацию и достоверность. Требуется обеспечить возможность получения и хранения этой информации.

**3.1.5. Требования к задаче «Вывод результатов»**

Необходимо использовать один из способов представления результатов, например, в виде текстового отчета, графиков, таблиц, диаграмм, интерактивных визуализаций. Требуется выбрать наиболее подходящий формат, который будет удобен и понятен пользователям.

**3.2 Требования к программной реализации задач**

Для обеспечения корректной работы и сохранения целостности данных, информация о новостных сообщениях и их атрибутах должна быть храниться в базе данных. База данных должна обеспечивать эффективное хранение и доступ к данным. Пользовательский интерфейс программы должен быть интуитивно понятным и удобным в использовании. Код программы должен быть оптимизирован для обеспечения эффективной работы. Информационные объекты, используемые в программе, должны быть понятными для пользователей. Например, атрибуты новостных сообщений и факторы, используемые для анализа фейковости, должны иметь понятные и описательные наименования.

* 1. **Построение модели требований в нотации UML**

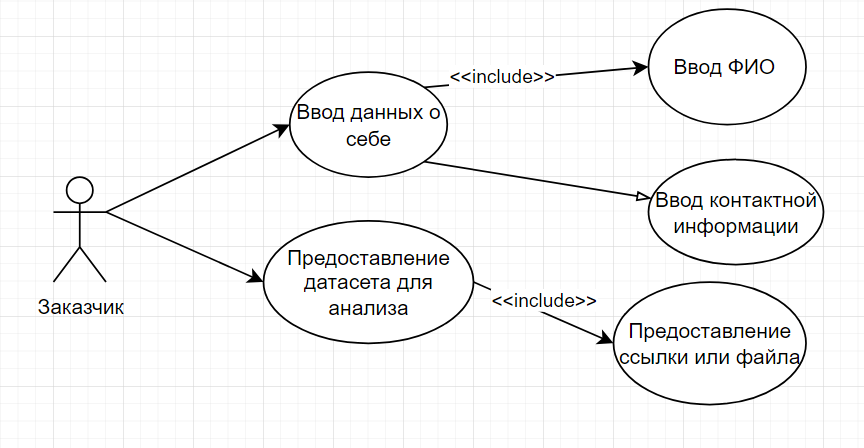


Рисунок 1 – UML-заказчик

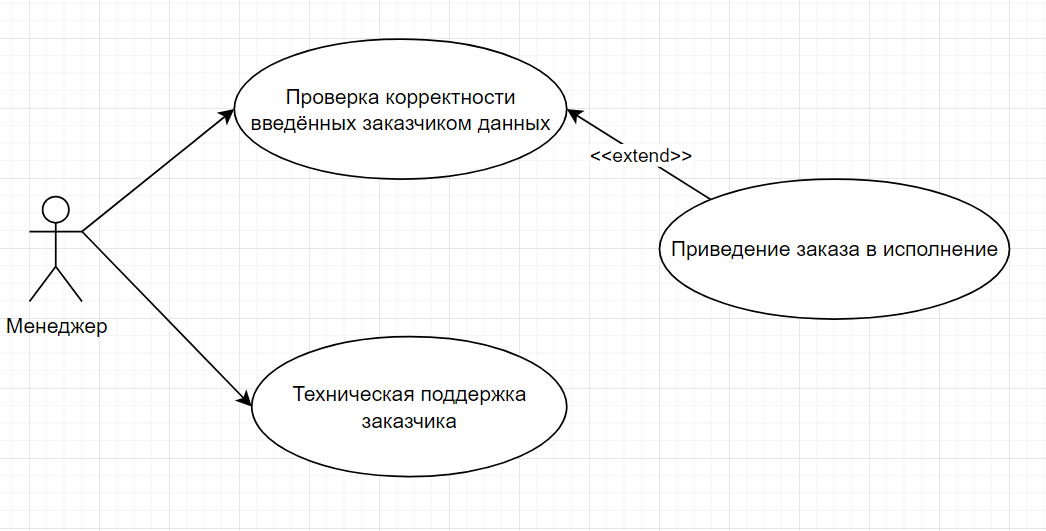


Рисунок 2 – UML-менеджер

* 1. **Требования к базе данных и к инструментальному программному обеспечению**

**5.1. Требования к информационному обеспечению**

• должна отсутствовать дублированная информация;

• должна быть обеспечена возможность ввода информации;

• должна быть защита от несанкционированного доступа к данным;

• должна быть обеспечена конфиденциальность информации;

• должна быть возможность получения данных с помощью языка запросов высокого уровня, без использования прикладных программ;

• БД должна иметь возможность неограниченно использоваться.

**5.2. Требования к инструментальному программному обеспечению**

Ниже будут представлены требования к системе управления базой данных (СУБД), к средству разработки программ (IDE), средствам автоматизированного проектирования ПО

• язык программирования: Python 3.10;

• выбрать среду разработки: PyCharm, JupyterNotebook, GoogleCollab;

• выбрать СУБД: MySQL, PostgreSQL.

* 1. **Нефункциональные требования**

1. *Защищенность –* система должна обеспечена защитой данных от вредоносных действий, таких как уничтожение, искажение или намеренное подделывание.
2. *Надежность –* система должна сохранять значения всех параметров в пределах установленных значений со временем. Важно, чтобы сбои в работе системы не приводили к опасным последствиям.
3. *Производительность* – система должна выполнять определенный объем работы за единицу времени.
4. *Удобный интерфейс –* интерфейс должен быть удобным, интуитивно понятным и продуманным, чтобы обеспечить простоту использования программного обеспечения. Дизайн не должен пугать клиентов излишней сложностью или изощренностью.
5. *Живучесть –* система должна сохранять или быстро восстанавливать способность к выполнению своих функций.

**Вывод**: проделав лабораторную работу №3, я освоил методики предварительного анализа разрабатываемой̆ программы, а также задачи формулирования функциональных и нефункциональных требований к программной реализации отдельных задач и к программе в целом, приобрёл навыки разработки технического задания, а также для своей индивидуальной темы установил назначение и общую цель создания программы, определил её структуру и разработал модель требований в нотации UML.