**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»**

Направление подготовки 01.03.02 Прикладная математика и информатика

Работу выполнил студент группы 4ИТ/2 Д.А. Качура

(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 автоматизации моделирования вольтамперных характеристик.

* 1. **Назначение и цель создания системы моделирования вольтамперных характеристик с использованием нейронной сети**

Система моделирования вольтамперных характеристик (ВАХ) с использованием нейронной сети предназначена для анализа и предсказания электрических характеристик устройств и материалов, основываясь на вольтамперных характеристиках. ВАХ является графическим представлением зависимости между напряжением и током в электрической цепи и широко используется в области электроники, физики и материаловедения для оценки и анализа работы устройств и материалов.

Основная цель создания системы моделирования ВАХ с использованием нейронной сети состоит в разработке эффективного инструмента, который может предсказать ВАХ для различных условий и параметров, исходя из имеющихся данных.

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

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

|  |  |
| --- | --- |
| **Задача** | **Описание** |
| Регистрация в системе | Пользователь осуществляет регистрацию в системе моделирования вольтамперных характеристик, предоставляя необходимые персональные данные: ФИО, контактный номер телефона и адрес электронной почты. |
| Создание модели | Пользователь создает модель, выбирая тип компонента или устройства для моделирования вольтамперных характеристик. Он вводит соответствующие параметры, такие как значения сопротивления, напряжения или тока, а также другие характеристики, необходимые для моделирования. |
| Проверка модели | После ввода параметров модели система проверяет их корректность и целостность, а также осуществляет предварительную проверку на возможные ошибки. Если все данные верны, пользователю предоставляется возможность продолжить процесс моделирования |
| Моделирование вольтамперных характеристик | На основе введенных пользователем данных система проводит моделирование вольтамперных характеристик выбранного компонента или устройства. Это может включать расчеты и анализ электрических параметров, таких как вольтамперные графики, потребляемая мощность, эффективность и другие характеристики. |
| Визуализация результатов | Полученные результаты моделирования представляются пользователю в удобной форме, например, в виде графиков, таблиц или диаграмм. Это позволяет пользователю анализировать и интерпретировать полученные вольтамперные характеристики.выдаётся запись в журнал продаж и клиенту отправляется чек и квитанция об оплате |
| Сохранение и экспорт данных | Пользователь имеет возможность сохранить результаты моделирования для последующего использования или экспорта. Это позволяет ему в дальнейшем сравнивать различные модели, делать изменения и вносить корректировки в параметры модели. |
| Генерация отчета | Система может автоматически сгенерировать отчет о проведенном моделировании вольтамперных характеристик. Отчет может содержать подробные данные о параметрах модели, полученные результаты, аналитические выводы и рекомендации. |

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 «Регистрация на сайте»**

Идентификатор каждого клиента должен быть уникальным, для сверки клиента с его заказом на складе необходимы следующие данные о клиенте:

* фамилия, имя, отчество;
* номер телефона;
* электронная почта.

На номер телефона клиент получит SMS с кодом подтверждения профиля. На электронную почту поступит письмо об успешной регистрации аккаунта в системе.

**3.1.2. Требования к задаче «Создание модели»**

Система должна предоставлять пользователю возможность выбора типа компонента или устройства для моделирования вольтамперных характеристик.

Система должна предоставлять пользователю интерфейс для ввода соответствующих параметров модели, таких как значения сопротивления, напряжения или тока, а также других необходимых характеристик.

Система должна проверять введенные пользователем параметры на корректность и целостность.

**3.1.3. Требования к задаче «Проверка модели»**

Система должна автоматически проверять введенные пользователем параметры модели на корректность и целостность.

Система должна осуществлять предварительную проверку на возможные ошибки, например, отсутствие обязательных параметров или противоречия в значениях параметров.

Если все данные верны, система должна предоставлять пользователю возможность продолжить процесс моделирования.

**3.1.4. Требования к задаче «Моделирование вольтамперных характеристик»**

Система должна проводить моделирование вольтамперных характеристик выбранного компонента или устройства на основе введенных пользователем данных.

Моделирование может включать расчеты и анализ электрических параметров, таких как вольтамперные графики, потребляемая мощность, эффективность и другие характеристики.

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

Система должна предоставлять пользователю результаты моделирования в удобной форме, например, в виде графиков, таблиц или диаграмм.

Результаты должны быть представлены таким образом, чтобы пользователь мог анализировать и интерпретировать полученные вольтамперные характеристики.

**3.1.6. Требования к задаче «Сохранение и экспорт данных»**

Система должна предоставлять пользователю возможность сохранить результаты моделирования для последующего использования.

Пользователь должен иметь возможность экспортировать результаты в различные форматы, например, в файлы CSV или PDF.

Сохранение и экспорт данных позволяют пользователю сравнивать различные модели, вносить изменения и вести аналитическую работу.

**3.1.7. Требования к задаче «Генерация отчета»**

Система должна автоматически генерировать отчет о проведенном моделировании вольтамперных характеристик.

Отчет должен содержать подробные данные о параметрах модели, полученные результаты, аналитические выводы и рекомендации.

Система должна предоставлять пользователю возможность просмотра и сохранения сгенерированного отчета.

**3.2 Требования к программной реализации задач**

Для корректной работы и целостности данных информация будет храниться в базе данных. Интерфейс будет интуитивно понятен и разбит на задачи при помощи отдельных объектов относительно проектируемой диаграмме вариантов использования ПО:

• по каждой из задач программа сможет запросить соответствующие входные данные и выдать результирующую информацию;

• код будет оптимизирован;

• информационные объекты будут понятны пользователям.

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

Изображение выглядит как текст, диаграмма, круг, Шрифт

Автоматически созданное описание

Рисунок 1 – UML-пользователь

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

**5.1. Требования к информационному обеспечению**

**•** требуется организация данных, которая должна сохраняться в ПС;

• должна отсутствовать дублированная информация, данные должны быть актуальны и достоверны;

• 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

• инструментальные средства программной инженерии будут представлять все графические элементы проектируемой ПО;

• все инструментальные средства разработки ПО должны обеспечивать возможность реализации ПС согласно макетам проектирования;

• требования к использованию программных пакетов: ПП не должны нагружать систему.

* 1. **Нефункциональные требования**

1. *Защищенность* (Безопасность), то есть способность обеспечения защиты данных от разрушения, искажения или преднамеренных фальсификаций злоумышленником. Характеризует возможное отсутствие риска, связанного с нанесением некоторого ущерба. Имеют особую важность, так как компания имеет много конкурентов.
2. *Надёжность* – свойство системы сохранять во времени в установленных пределах значения всех параметров, характеризующих способность системы выполнять требуемые функции в заданных режимах и эксплуатации. Важно, чтобы сбои в работе системы не должны приводить к опасным последствиям.
3. *Производительность* – характеристика системы, отражающая ее способность производить определённый̆ объем работ в единицу времени, например, время ответа, доступность, число продуктов, полученная прибыль.
4. *Наглядность интерфейса* – должен быть удобным, интуитивно понятным и продуманным, что обеспечит наглядность и простоту использования ПО, благодаря чему дизайн не будет пугать клиентов излишней сложностью и ненужной изощрённостью.
5. *Наличие технической поддержки*. Введенная в эксплуатацию готовая система требует определённой̆ технической̆ поддержки, что особо важно для интернет-магазина, ведь в ведении подобного процесса всего идёт диалог клиента с администратором сайта или бухгалтером. Это обусловлено, прежде всего, динамичностью информационных процессов: совершенствованием документооборота, появлением дополнительных структур данных и автоматизированных функций, что является обычным явлением для любых развивающихся систем.

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