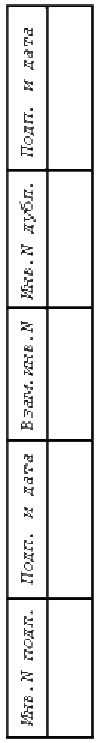
Разработка системы оценки сложности планов SQL запросов

Техническое задание

643.1.0001-01 90 01-1

Листов 12



СОДЕРЖАНИЕ

[1. ВВЕДЕНИЕ 3](#__RefHeading___Toc416_2031643108)

[1.1 Наименование темы 3](#__RefHeading___Toc432_2031643108)

[1.2 Актуальность разработки 3](#__RefHeading___Toc456_20316431081)

[1.3 Область применения 3](#__RefHeading___Toc456_2031643108)

[2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ 4](#__RefHeading___Toc430_2031643108)

[3. НАЗНАЧЕНИЕ РАЗРАБОТКИ 5](#__RefHeading___Toc428_2031643108)

[3.1 Цель проекта 5](#__RefHeading___Toc454_2031643108)

[3.2 Задачи проекта 5](#__RefHeading___Toc452_2031643108)

[4. СОДЕРЖАНИЕ И ОСНОВНЫЕ ТРЕБОВАНИЯ К ПРОВЕДЕНИЮ ДИПЛОМНОЙ РАБОТЫ 7](#__RefHeading___Toc426_2031643108)

[4.1 Требования к функциональным характеристикам 7](#__RefHeading___Toc1126_1229431424)

[4.1.1 Требования к составу выполняемых функций 7](#__RefHeading___Toc458_2031643108)

[4.1.2 Требования к организации входных и выходных данных 7](#__RefHeading___Toc470_2031643108)

[4.1.3 Требования к временным характеристикам 7](#__RefHeading___Toc468_2031643108)

[4.2 Требования к надежности 7](#__RefHeading___Toc448_2031643108)

[4.3 Требования к эксплуатации 8](#__RefHeading___Toc446_2031643108)

[4.3.1 Климатические условия эксплуатации 8](#__RefHeading___Toc466_2031643108)

[4.4 Требования к информационной и программной совместимости 8](#__RefHeading___Toc444_2031643108)

[4.4.1 Требования к информационным структурам и методам решения 8](#__RefHeading___Toc464_2031643108)

[4.4.2 Требования к исходным кодам и языкам программирования 8](#__RefHeading___Toc462_2031643108)

[4.4.3 Требования к программным средствам, используемым программой 8](#__RefHeading___Toc460_2031643108)

[4.5 Требования к маркировке и упаковке 8](#__RefHeading___Toc442_2031643108)

[4.6 Требования к программной документации 8](#__RefHeading___Toc440_2031643108)

[5. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ 9](#__RefHeading___Toc420_2031643108)

[5.1 Разработка архитектуры системы 9](#__RefHeading___Toc440_20316431081)

[5.2 Создание базы данных и настройка ее структуры в соответствии с требованиями 9](#__RefHeading___Toc440_203164310811)

[5.3 Реализация серверной и клиентской сторон системы 9](#__RefHeading___Toc440_2031643108111)

[5.4 Разработка модулей для оценки сложности планов SQL запросов 10](#__RefHeading___Toc440_2031643108112)

[5.5 Реализация пользовательского интерфейса 10](#__RefHeading___Toc440_2031643108113)

[5.6 Unit тестирование 10](#__RefHeading___Toc440_2031643108114)

[5.7 Интеграционное тестирование и отладка 10](#__RefHeading___Toc440_2031643108115)

[6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 11](#__RefHeading___Toc418_20316431081)

[7. ПОРЯДОК КОНТРОЛЯ ПРИЕМКИ 12](#__RefHeading___Toc418_2031643108)

### ВВЕДЕНИЕ

#### 1.1 Наименование темы

Тема разработки ПО — “Разработка учебной системы оценки сложности планов SQL запросов”.

#### 1.2 Актуальность разработки

SQL (Structured Query Language) является одним из основных инструментов работы с базами данных и широко применяется в различных сферах, включая программирование, аналитику данных, бизнес-разработку и т. д. Разработка системы, которая помогает оценивать сложность SQL запросов, может быть востребована среди студентов, обучающихся и профессиональных специалистов. Эффективные SQL запросы играют важную роль в производительности баз данных. Оценка сложности планов SQL запросов помогает разработчикам и аналитикам улучшить производительность и оптимизировать эффективность запросов, что может привести к снижению времени выполнения запросов и улучшению общей производительности системы

#### 1.3 Область применения

Полученная система может быть использована в учебных заведениях, где изучаются базы данных и SQL, таких как: университеты, колледжи и учебные курсы. Система может быть полезной для компаний, занимающихся разработкой программного обеспечения, баз данных или аналитикой данных. Разработчики и аналитики смогут использовать систему для анализа и оптимизации SQL запросов, а также для оценки производительности баз данных. Система может стать основой для дальнейших исследований и разработки новых методов оценки сложности запросов и оптимизации баз данных.

### ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

Основанием для разработки является договор от 01.09.2023 номером \_. Договор утвержден ректором НГУ Лаврентьевым М.М., именуемым в дальнейшем «Заказчиком», и Савченко Егором Владимировичем, именуемым в дальнейшем «Исполнителем».

Согласно Договору, Исполнитель обязан провести исследование методов оценивания сложности SQL запросов, разработать учебную систему по оцениванию сложности планов SQL запросов, предоставить Заказчику результаты исследований и программный код системы не позднее 06.06.2024.

### НАЗНАЧЕНИЕ РАЗРАБОТКИ

Назначение разработки состоит в создании программного инструмента, предназначенного для обучения, изучения и практического применения навыков анализа сложности планов SQL запросов. Главной преследуемой целью разработки является помощь студентам и учащимся в освоении концепций, методов и практических аспектов определения сложности SQL запросов.

#### 3.1 Цель проекта

Разработать систему оценки сложности планов SQL запросов.

#### 3.2 Задачи проекта

1. Проектирование системы: Разработать архитектуру и дизайн учебной системы, определить основные модули, функции и интерфейсы системы.
2. Разработка обучающего материала: Создать обучающий материал, который поможет студентам понять основы SQL запросов, структуру, синтаксис и основные концепции.
3. Разработка компоненты оценки сложности: Реализовать компоненту системы, которая позволит оценивать сложность планов SQL запросов на основе различных метрик, таких как количество операторов, уровень вложенности, количество условий и т. д.
4. Разработка компоненты измерения производительности: Создать компоненту системы, которая позволит измерять производительность SQL запросов, включая время выполнения запроса, объем передаваемых данных и другие показатели производительности.
5. Разработка анализатора запросов: Реализовать анализатор, который будет анализировать структуру и синтаксис SQL запросов, выявлять возможные проблемы и улучшения для оптимизации запросов.
6. Разработка среды выполнения и практических заданий: Создать среду выполнения SQL запросов, где студенты смогут практиковаться, выполнять запросы и получать обратную связь о сложности и производительности.
7. Тестирование и отладка: Провести тестирование системы, выявить и исправить возможные ошибки и проблемы.
8. Оценка эффективности: Провести оценку эффективности разработанной системы, сравнить результаты оценки сложности и измерения производительности с реальными показателями.
9. Документирование: Подготовить документацию по системе: инструкции для пользователей, разработчиков и администраторов.
10. Внедрение и демонстрация: Подготовить систему к внедрению, провести демонстрацию и обучение пользователей.
11. Поддержка и сопровождение: Обеспечить поддержку и сопровождение системы после внедрения, реагировать на обратную связь от пользователей, исправлять возникающие проблемы и предлагать улучшения.

### СОДЕРЖАНИЕ И ОСНОВНЫЕ ТРЕБОВАНИЯ К ПРОВЕДЕНИЮ ДИПЛОМНОЙ РАБОТЫ

#### 4.1 Требования к функциональным характеристикам

Система должна предоставлять DSL (DSL должен быть написан на языке Haskell), который предоставляет «fluent interface» по вводу языка напоминающего SQL. С помощью DSL пользователи могут выполнять запросы и получать результаты в виде реряционных данных, а также информацию о сложности запроса к виде текста. Система должна также автоматически проверять правильность выполнения запросов и выдавать обратную связь пользователю.

##### 4.1.1 Требования к составу выполняемых функций

К системе можно сконструировать запрос, на который она даст ответ.

##### 4.1.2 Требования к организации входных и выходных данных

Входные данные программы состоят из пользовательского запроса и конфигурации отображения на реальную базу данных. Пользовательский запрос - формируются пользователем посредствам синтаксиса DSL-клиента. Конфигурация отображения базы данных задается в формате json (содержит перечисление названий необходимых таблиц и необходимые данные для подключения к существующей базе данных).

Выходные данные состоят из 2х частей: json-объект с описанием сложности запроса и json-объект с результатами выполнения запроса.

##### 4.1.3 Требования к временным характеристикам

Время исполнения запроса не должно превышать timeout указанный для этого запроса (timeout указывается пользователем, является частью запроса)

#### 4.2 Требования к надежности

Требования к надежности не предъявляются.

#### 4.3 Требования к эксплуатации

##### 4.3.1 Климатические условия эксплуатации

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к ноутбукам и стационарным компьютерам в части условий их эксплуатации.

#### 4.4 Требования к информационной и программной совместимости

Необходимо любое окружение удовлетворяющее минимальным системным требованиям программы Docker Desktop.

##### 4.4.1 Требования к информационным структурам и методам решения

База данных работает под управлением Microsoft SQL Server или Postgre SQL. Используется многопоточный доступ к базе данных. Необходимо обеспечить одновременную работу всех компонентов системы.

##### 4.4.2 Требования к исходным кодам и языкам программирования

Исходный код сервера системы должен быть реализован на языке Java. Исходный код клиента системы должн быть реализован на языке Haskell. В качестве интегрированной среды разработки программы может быть использована среда «Intellij Idea».

##### 4.4.3 Требования к программным средствам, используемым программой

Java 11, Parboiled 1.4.1. Haskell GHC version 9.2.1. Docker Desktop 4.25.0

#### 4.5 Требования к маркировке и упаковке

Программное изделие передается по сети Internet в формате .tar структура которого задается форматом модулей docker image. Специальных требований к маркировке не предъявляется.

#### 4.6 Требования к программной документации

В программную документацию должны входить следующие документы: техническое задание (включает описание применения), спецификация, текст (код) и описание программы, ТЗ, пояснительная записка, руководство по техническому обслуживанию, описание применения.

### СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

Разработка должна быть проведена в несколько стадий.

1. Разработка архитектуры системы;
2. Создание базы данных и настройка ее структуры в соответствии с требованиями;
3. Реализация серверной и клиентской сторон системы;
4. Разработка модулей для оценки сложности планов SQL запросов;
5. Реализация пользовательского интерфейса;
6. Unit тестирование;
7. Интеграционное тестирование и отладка;

Далее подробно описывается каждый из этапов, конкретизируются проводимые действия.

#### 5.1 Разработка архитектуры системы

Определение основных компонентов системы, включая сервер, клиентское приложение и базу данных. Разработка диаграммы классов и диаграммы последовательностей для иллюстрации взаимодействия компонентов. Определения алгоритмов и методов оценки сложности планов SQL запросов, которые будут использоваться в системе. Разработка архитектурного дизайна и определение технологий, которые будут применяться.

#### 5.2 Создание базы данных и настройка ее структуры в соответствии с требованиями

Создание таблиц, определение связей между ними и настраивание ограничений целостности данных. Определение необходимых индексов для обеспечения эффективности выполнения запросов. Реализация механизмов обновления базы данных при изменении требований или добавлении новых функций.

#### 5.3 Реализация серверной и клиентской сторон системы

Разработка серверной часть системы, которая будет обрабатывать запросы от клиентского приложенияю. Создание API (Application Programming Interface) для взаимодействия между сервером и клиентом. Реализация функции обработки запросов от клиентского приложения, включая оценку сложности планов SQL запросов. Обеспечение безопасности (аутентификацию) при обмене данными между сервером и клиентом.

#### 5.4 Разработка модулей для оценки сложности планов SQL запросов

Разработайтка модулей, которыми будут использоваться для оценки сложности планов SQL запросов. Определение и реализация алгоритмов оценки сложности, включая учет различных факторов, таких как количество таблиц, условий, соединений и других. Интеграция этих модулей в систему.

#### 5.5 Реализация пользовательского интерфейса

Разработка интерфейса взаимодействия пользователя с системой для ввода и отображения запросов. Создание формы и элементов управления для ввода SQL запросов. Определение и реализация функций отображения результатов оценки сложности запросов и дополнительной информации.

#### 5.6 Unit тестирование

Написание модульных тестов для каждого компонента системы. Выполнение автоматическое тестирование при помощи фреймворка для тестирования.

#### 5.7 Интеграционное тестирование и отладка

Проведение тестирования взаимодействия компонентов системы и обработки запросов от клиентского приложения. Проверка правильности передачи данных между сервером и клиентом. Отлаживание и исправление ошибкок, которые были обнаружены в процессе тестирования. Проверка эффективности системы и обеспечение стабильной работы.

### ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

При разработке учебной системы оценки сложности планов SQL запросов, основным акцентом будет являться приведение алгоритмов оценки сложности и разрешение типичных задач, связанных с оценкой планов запросов. Система является учебной и не предполагает коммерческую эксплуатацию, необходимость в представлении детальных расчетов финансовых показателей отпадает.

### ПОРЯДОК КОНТРОЛЯ ПРИЕМКИ

По окончании разработки продукта программа поставляется заказчику, где она проходит окончательную проверку на соответствие всем требованиям заказчика, и правильность выполнения всех функций (комплексное тестирование со стороны Заказчика). В случае обнаружения ошибок и/или недочетов, програмное средство дорабатывается Исполнителем в течение срока не превышающего две недели.