**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**

**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

***Факультет Информационных технологий***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**КУРСОВОЙ ПРОЕКТ**

**Дисциплина:** \_\_\_\_\_\_\_\_Объектно-ориентированное программирование (С#)\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Тема:** \_\_\_\_Информационная система управления товаров на складе\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Выполнил(а): студент(ка) группы \_\_221-371\_\_**

\_\_\_\_\_\_\_\_\_\_Куренков Алексей Олегович\_\_\_\_\_\_\_\_\_\_

(Фамилия И.О.)

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил: \_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

(Фамилия И.О., степень, звание)

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва**

**2022**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**

**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

***Факультет Информационных технологий***

***Кафедра Информатики и информационных технологий***

**\_\_\_\_\_\_\_\_\_\_\_\_Утверждаю**

**Зав. кафедрой Е.В. Булатников**

**« \_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.**

**ЗАДАНИЕ**

**на курсовое проектирование**

**Направление 09.03.02 «Информационные системы и технологии»**

**Группа: 221-371**

**Дисциплина:**\_\_\_\_\_\_ Объектно-ориентированное программирование (С#)\_\_\_\_\_\_

**Студент: \_\_\_**Куренков Алексей Олегович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1. Тема проекта: \_\_\_\_\_** Информационная система управления товаров на складе\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**2. Срок сдачи студентом законченного проекта: \_\_22** Декабря 2023 года**\_\_\_\_\_\_**

**3. Исходные данные к проекту:\_** Язык программирования: C# , Microsoft Visual Studio,Фреймворк: .NET Framework \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**4. Содержание расчетно-пояснительной записки (перечень подлежащих**

**разработке вопросов): Разработка модели данных, Описание функциональности и основных требований к приложению \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**5. Перечень графического материала (с точным указанием обязательных**

**чертежей):\_\_\_\_\_**Листинг программы, скриншоты этапов программирования**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**6. Литература и прочие материалы, рекомендуемые студенту для изучения:**

**Джозев, Ал. and Aл. Бен, 2021. C# 9.0. Карманный справочник. Диалектика, стр: 256; Марк, Пр., 2023. C# 10 и.NET 6. Современная кросс-платформенная разработка. Питер, стр: 848; Рихтер, Дж., 2023. CLR via C#. Программирование на платформе Microsoft. NET Framework 4.5 на языке C#. 4-е изд.. Питер, стр: 896; Болье, А., 2021. Изучаем SQL. Генерация, выборка и обработка данных. Диалектика, стр: 400; Маркин, А. В. Программирование на sql в 2 ч. Часть 2 : учебник и практикум для бакалавриата и магистратуры / А. В. Маркин. — М. : Издательство Юрайт, 2019. — 292 с; Литвиненко, Н.А., 2022. Декларативное программирование на языке XAML + C#. WPF проекты. Горячая линия - Телеком, стр: 320.**

**7. Промежуточные сроки исполнения \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**8. Дополнительные указания \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**9. Дата выдачи задания \_\_\_7\_Сентября 2023\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Руководитель проекта: \_\_* *Полубояринова Анастасия Сергеевна \_\_\_\_\_\_\_\_\_\_\_\_*

*Задание принял студент:\_Куренков Алексей Олегович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 5](#_Toc153487332)

[**1** **АНАЛИТИЧЕСКАЯ ЧАСТЬ** 6](#_Toc153487333)

[**1.1** **Разработка модели данных** 6](#_Toc153487334)

[**1.2** **Описание функциональности и основных требований к приложению** 6](#_Toc153487335)

[**2** **ТЕОРЕТИЧЕСКАЯ ЧАСТЬ** 7](#_Toc153487336)

[**2.1** **Язык программирования C#** 7](#_Toc153487338)

[**2.2** **Microsoft Visual Studio** 8](#_Toc153487339)

[**2.3** **Фреймворк: .NET Framework** 8](#_Toc153487340)

[**3** **ПРАКТИЧЕСКАЯ ЧАСТЬ** 9](#_Toc153487341)

[**3.1** **Разработка баз данных** 9](#_Toc153487343)

[**3.2** **Создание окна Авторизации** 9](#_Toc153487344)

[**3.3** **Создание основного функционала приложения** 9](#_Toc153487345)

[**ЗАКЛЮЧЕНИЕ** 10](#_Toc153487346)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 11](#_Toc153487347)

[**ПРИЛОЖЕНИЕ** 12](#_Toc153487348)

# **ВВЕДЕНИЕ**

**Система управления складом** – это мощный программный инструментарий, предназначенный для автоматизации управления процессами склада. Система управления складом помогает уменьшить трудозатраты, сократить ошибки при отборе и отгрузке заказов, упростить проведение инвентаризации и повысить ее точность, улучшить качество обслуживания клиентов.

Актуальность данной темы заключается в разработке простой, но структурированной системой управления товаров на складе. В ней должно четко прописана логическая структура данных, поскольку такая модель позволяет легко и быстро получать, обрабатывать и выдавать нужные нам результаты запросов из склада к сотруднику предприятия.

Для этого необходимо выявить следующие задачи:

1. Разработать логику модели данных под товары, склад и сотрудников
2. Управление данных склада предприятия, а конкретно: добавление, обновление и удаление.
3. Добавление роли. Пользователи разных должностей могут работать с разными типами данных, поэтому должно быть четкое распределение доступа к данным.
4. Безопасность. Любое приложение ограничивается доступом к данным предприятия при помощи входа в систему через зарегистрированные аккаунты. Это делается в целях защиты от кражи данных, которые могут повлиять на работу предприятия в целом.

# **АНАЛИТИЧЕСКАЯ ЧАСТЬ**

## **Разработка модели данных**

Перед тем, как начать разрабатывать приложение и базу данных, потребуется продумать логическую модель данных, которую будет использовать в последующем. Это достаточно важный момент, поскольку в случае возникновения ошибок на поздних этапах разработки, которые будут связаны с фундаментом, придется всю разработку свернуть и вернуться к начальному этапу, что сильно скажется на предприятии в худшую сторону. Для этого приложения потребуется продумать несколько небольших моделей данных под каждый конкретный объект, а именно:

* Склад. В него будут храниться все наши товары на складе. Причем они могут храниться в нескольких секторах. В каждом секторе может быть некоторое кол-во товаров. Ко всему этому, за каждый склад отвечает конкретный сотрудник предприятия.
* Сотрудники. В любом предприятии всегда имеется сотрудники, который отвечает за ту или иную задачу. В его параметры входят: ФИО, дата рождения, зарплата, должность. Кроме этого, сотрудники имеют персональные аккаунты соответствующими со своими должностями управлением складом.
* Товары. На текущий момент товары имеют два основных типов: Книги и Техника. У каждого типа есть свои особенности и представления данных.

Теперь следует остановиться на каждом из них. Для начала стоит разобраться с сотрудниками. Выше уже были описаны их основные свойства, однако их надо стоит разбить до атомарных данных, где каждый отдельный вид информации описывает конкретное назначение. В итоги мы будем иметь:

* Фамилия
* Имя
* Отчество
* Дата рождения,
* Зарплата

Должность (или роль) будет выноситься под другую таблицу. Это из-за того, что у нас данные должны быть не только атомарными и простыми, но и уникальными. Сотрудников в предприятии могут быть несколько, имеющий одну и ту же должность, в следствии чего, данные будут повторяться. Хоть это не сильно будет влиять на модель, однако без разбиения, наша база данных будет занимать больше места. Поэтому необходимо один раз прописать все должности сотрудников отдельно и привязать их к нашим сотрудникам. В итоге у нас получается две модели: сотрудники и их должности. Не стоит забывать, что у них есть аккаунты, нацелены на безопасность от чужого проникновения в систему. Поэтому третья модель будет нацелена на описание аккаунтов сотрудников, состоящий из двух основных полей: Логин и Пароль.

## **Описание функциональности и основных требований к приложению**

# **ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**



## **Язык программирования C#**

**С# (“Си Шарп”)** - один из наиболее быстро растущих и востребованных, а также в какой-то степени “удобный” язык программирования. Это модификация фундаментального языка С/C++ от компании Microsoft, позволяющее создать наиболее универсальное средство для разработки программного обеспечения на большого количества устройств и операционных систем.

С#‎ разрабатывался как язык программирования прикладного уровня для CLR и, зависит, прежде всего, от возможностей самой CLR(Common Language Runtime). Common Language Runtime из себя представляет исполняющую среду для байт-кода CIL(Common Intermediate Language или высокоуровневый ассемблер), в который компилируются программы, созданные на .NET-совместимых языках программирования. CLR считается одним из основных компонентов пакета Microsoft .NET Framework. Прежде всего, системы типов С#‎, которая отражает BCL(Base Class Library) диктуется тем, может ли конкретная языковая особенность быть транслирована в соответствующие стилю CLR. Так, с развитием CLR от версии 1.1 к 2.0 значительно расширился и сам C#; подобного следует ожидать и в дальнейшем. CLR предоставляет С#‎, как и всем другим .NET-ориентированным языкам, многие возможности, которых лишены “классические” языки программирования. Например, сборка мусора не реализована в самом C#‎, а производится в CLR для программ, написанных на данном языке, точно так же, как это делается для программ на VB.NET, J# и др.

*Основные плюсы C#:*

Независимость от аппаратного функционала. Программу не нужно адаптировать или внедрять проект под многочисленные платформы.

Поддержка. Данный язык до сих пор поддерживается и обслуживается компанией Microsoft.

Управление памятью. Если требуется, чтобы программа работала как можно стабильнее, необходимо часто выполнять очистку от различного мусора. Язык программирования C# выполняет эту задачу в автоматическом режиме. В итоге, разработчику не нужно контролировать расход памяти, устранять её утечки или удалять ненужные части кода.

Строгость типизации. Если вы объявляете переменную в C#, необходимо сначала указать, что в ней находится (например – строка, число или массив). Разработка таким методом выполняется немного дольше, но написанный код становится более предсказуемым и понятным.

*Теперь основные недостатки C#:*

Невысокая скорость. При открытии программы на C#, код сначала адаптируется под конкретное техническое оборудование, а уже после исполняется.

Безопасность. Специалисты считают, что код, написанный на C#, очень просто декомпилировать (перевести из машинного языка в человеческий). Следовательно, программу без особых усилий может прочитать и взломать хакер, чтобы написать вредоносный софт.

## **Microsoft Visual Studio**

## **Фреймворк: .NET Framework**

# **ПРАКТИЧЕСКАЯ ЧАСТЬ**



## **Разработка баз данных**

## **Создание окна Авторизации**

## **Создание основного функционала приложения**

# **ЗАКЛЮЧЕНИЕ**

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Джозев, Ал. and Aл. Бен, 2021. C# 9.0. Карманный справочник. Диалектика, стр: 256.
2. Марк, Пр., 2023. C# 10 и.NET 6. Современная кросс-платформенная разработка. Питер, стр: 848.
3. Рихтер, Дж., 2023. CLR via C#. Программирование на платформе Microsoft. NET Framework 4.5 на языке C#. 4-е изд.. Питер, стр: 896.
4. Болье, А., 2021. Изучаем SQL. Генерация, выборка и обработка данных. Диалектика, стр: 400.
5. Учебник по языку SQL (DDL, DML) на примере диалекта MS SQL Server. Date Views 2023 год www.habr.com.
6. Нормализация отношений. Шесть нормальных форм. Date Views 2023 год www.habr.com.
7. Максименко, Д., 2023. Разработка индекса для системы управления базами данных MySQL на основе B-link деревьев для интенсивной многопоточной вставки. Разработка индекса для системы управления базами данных MySQL на основе B-link деревьев для интенсивной многопоточной вставки, Санкт-Петербургский политехнический университет Петра Великого Date Views 2023 год www.elib.spbstu.ru.
8. Маркин, А. В. Программирование на sql в 2 ч. Часть 2 : учебник и практикум для бакалавриата и магистратуры / А. В. Маркин. — М. : Издательство Юрайт, 2019. — 292 с
9. Литвиненко, Н.А., 2022. Декларативное программирование на языке XAML + C#. WPF проекты. Горячая линия - Телеком, стр: 320.

# **ПРИЛОЖЕНИЕ**

Листинг работы