|  |
| --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  федеральное государственное АВТОНОМНОЕ образовательное учреждение высшего образования  «Национальный исследовательский ядерный университет «МИФИ» |
| **Димитровградский инженерно-технологический институт –**  филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ»  **(ДИТИ НИЯУ МИФИ)** |

Факультет информационно-технологический и социально-экономических наук

Кафедра Информационных технологий

Направление подготовки 09.03.01 Информатика и вычислительная техника

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**К КУРСОВОМУ ПРОЕКТУ НА ТЕМУ**

Разработка модуля авторизации для серверной инфраструктуры маркетплейса автопроизводителя

Студент

Группы ВТ-41 Д.Д. Кузин

(подпись)

Руководитель

ассистент М.С. Сутягина

(подпись)

Заведующий кафедрой

доцент, к.п.н. О.А. Ракова

(подпись)

Димитровград 2025

|  |
| --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  федеральное государственное АВТОНОМНОЕ образовательное учреждение высшего образования  «Национальный исследовательский ядерный университет «МИФИ» |
| **Димитровградский инженерно-технологический институт –**  филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ»  **(ДИТИ НИЯУ МИФИ)** |

**ФАКУЛЬТЕТ** Информационно-технологический и социально-экономических наук

**КАФЕДРА** Информационных технологий

**НАПРАВЛЕНИЕ ПОДГОТОВКИ** 09.03.01 **ГРУППА** ВТ-31

**ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ**

1. Фамилия, имя, отчество Кузин Данила Денисович

2. Тема «Разработка ИС для фирмы по продаже запчастей

3. Срок сдачи студентом готовой работы

4. Руководитель курсовой работы Сутягина Мария Сергеевна

Дата выдачи задания «\_18\_»\_апреля\_\_\_2025 г.

Руководитель курсовой работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Задание принял к исполнению \_\_\_15 февраля\_2024\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата и подпись студента)

1. Исходные данные к работе:

*требования к уровню разработки программного продукта:*

*а) разработка базы данных на языке Transact-SQL;*

*а) современный пользовательский интерфейс;*

*б) язык программирования C#.*

1. Содержание работы:
   1. *теоретическая часть:*

* исследование предметной области;
* описание логики программы;
* проектирование элементов интерфейса.
  1. *практическая часть:*
* реализация интерфейса;
* реализация логического компонента программного продукта.

1. Основная литература по теме работы:
2. *Карпова, И.П. База данных. Курс лекций и материалы для практических занятий [Текст] // Учебное пособие. – М.: Питер, 2013. – 240 с.*
3. *Сергеев, С.Ф. Введение в проектирование интеллектуальных интерфейсов [Текст] / С.Ф. Сергеев, П.И. Падерно, Н.А. Назаренко. – СПб: СПбГУ ИТМО, 2011. – 108 с.*
4. Отчётный материал работы:
5. *пояснительная записка;*
6. *CD/DVD-диск с пояcнительной запиской и файлами проекта приложения.*

РЕФЕРАТ

Кузин Д.Д. «РАЗРАБОТКА МОДУЛЯ АВТОРИЗАЦИИ ДЛЯ СЕРВЕРНОЙ ИНФРАСТРУКТУРЫ МАРКЕТПЛЕЙСА АВТОПРОИЗВОДИТЕЛЯ»: КУРСОВОЙ ПРОЕКТ / ДИТИ НИЯУ МИФИ № КП.09.03.01.71.2024 – Димитровград, 2024. – стр. 32, рис. 16, табл. 4, библ. назв. 5.

Ключевые слова: База данных ……

Объект исследования –.

Цель работы –

Задачи:.

Выводы: в ходе курсовой работы была изучена предметная область «Фирма по продаже запчастей», определили главные сущности и их свойства, создали базу данных в MS SQL Server. На основе исследований, а также создания базы данных разработана информационная система на языке программирования C# для платформы Windows.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 5](#_Toc166279974)

[1.1 Понятие и определение базы данных 6](#_Toc166279975)

[1.2 Понятие модели данных 6](#_Toc166279976)

[1.3 Исследование методов проектирования 6](#_Toc166279977)

[1.4 Сценарий пользовательского интерфейса 6](#_Toc166279978)

[1.5 Исследование и обоснование выбора инструментальных средств разработки программного продукта 6](#_Toc166279979)

[2 ПРАКТИЧЕСКАЯ ЧАСТЬ 7](#_Toc166279980)

[2.1 Разработка базы данных 7](#_Toc166279981)

[2.2 Технология программирования интерфейса программы 7](#_Toc166279982)

[2.3 Описание инструментов программирования. 7](#_Toc166279983)

[2.4 Тестирование программного продукта. 7](#_Toc166279984)

[Заключение 8](#_Toc166279985)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 9](#_Toc166279986)

ВВЕДЕНИЕ

В современном цифровом мире информационные технологии играют ключевую роль в обеспечении безопасности и эффективного функционирования бизнес-процессов. Для крупных компаний, таких как автопроизводители, надежная серверная инфраструктура является основой стабильной работы всех сервисов, включая маркетплейсы, которые обеспечивают взаимодействие с клиентами, дилерами и партнерами. Одним из критически важных компонентов такой инфраструктуры является система авторизации пользователей, позволяющая управлять доступом к различным сервисам и защищать конфиденциальные данные.

В данном курсовом проекте будет рассмотрен процесс разработки модуля авторизации пользователей для серверной инфраструктуры маркетплейса автопроизводителя. Будут проанализированы основные требования к системе авторизации, изучены современные методы аутентификации и защиты данных, а также реализованы механизмы управления правами пользователей.

Актуальность темы: Разработка безопасного и масштабируемого модуля авторизации позволит повысить уровень защиты пользовательских данных и предотвратить несанкционированный доступ к информационным ресурсам маркетплейса. Это особенно важно в условиях роста киберугроз и увеличения объемов персональных данных, обрабатываемых в цифровых экосистемах.

Цель проекта: Создание надежной и гибкой системы авторизации, обеспечивающей безопасный доступ пользователей к ресурсам маркетплейса, а также управление их ролями и правами.

Задачи проекта:

Анализ предметной области и изучение существующих решений в области авторизации пользователей.

* Разработка требований к модулю авторизации с учетом специфики маркетплейса автопроизводителя.
* Выбор технологий и инструментов для реализации системы аутентификации и управления доступом.
* Проектирование архитектуры модуля, разработка логической и физической моделей данных.
* Реализация механизма аутентификации пользователей с поддержкой многофакторной авторизации.
* Обеспечение безопасности хранения и передачи данных, внедрение современных криптографических методов.
* Тестирование и оптимизация модуля, интеграция с существующей серверной инфраструктурой маркетплейса.

1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

## Основные понятия и принципы аутентификации и авторизации

Аутентификация — это процедура проверки подлинности пользователя, то есть подтверждение, что он действительно является тем, за кого себя выдает. Чаще всего это осуществляется с помощью пароля, пин-кода, биометрических данных или других методов [1].

​Авторизация — это процесс предоставления пользователю или группе пользователей определенных разрешений, прав доступа и привилегий в компьютерной системе [2].

​В сфере информационной безопасности аутентификация и авторизация являются ключевыми процессами, обеспечивающими контроль доступа к системам и данным. Существует несколько видов аутентификации и авторизации, применяемых в зависимости от требований безопасности и специфики системы. Далее рассмотрены некоторые виды аутентификации:

* + - 1. По количеству факторов:
* **Однофакторная аутентификация**: использует один метод проверки, например, ввод пароля [3]. ​
* **Многофакторная аутентификация (MFA)**: предполагает использование нескольких независимых факторов для подтверждения личности, таких как знание: то, что знает пользователь (пароль, PIN-код);владение: то, что есть у пользователя (смартфон, смарт-карта);присутствие: биометрические данные пользователя (отпечаток пальца, распознавание лица).
  + - 1. По методам реализации:
* **Парольная аутентификация**: самый распространенный метод, при котором пользователь вводит статический или одноразовый пароль [4]. ​
* **Биометрическая аутентификация**: использует уникальные физиологические характеристики пользователя, такие как отпечатки пальцев или голос.
* **Аутентификация с использованием токенов**: предполагает применение физических или программных устройств, генерирующих одноразовые коды доступа. ​
* **Аутентификация по сертификатам**: основана на использовании цифровых сертификатов для подтверждения личности пользователя [5]. ​

Виды авторизации:

* + - 1. По моделям управления доступом:
* **Ролевая (RBAC)**: пользователям назначаются определенные роли, каждая из которых имеет набор разрешений. ​
* **Мандатная (MAC)**: доступ к ресурсам определяется на основе установленных меток безопасности и правил, заданных администратором. ​
* **Дискреционная (DAC)**: владелец ресурса самостоятельно определяет, кто и какие действия может выполнять с его ресурсами. ​
  + - 1. По способу предоставления доступа:
* **Онлайн-авторизация**: требует постоянного соединения с сервером для проверки прав доступа. ​
* **Офлайн-авторизация**: позволяет пользователю получать доступ к ресурсам без постоянного подключения к сети, основываясь на ранее полученных данных. ​

1. По количеству проверок:

* **Одноступенчатая авторизация**: предполагает единовременную проверку прав доступа. ​
* **Многоступенчатая авторизация**: требует прохождения нескольких этапов проверки перед предоставлением доступа [6].

## Понятие модели данных

Модель данных – множество допустимых типов данных, а также отношений и операций, определённых на данных этих типов. С математической точки зрения модель данных может трактоваться как алгебраическая система. В информационных технологиях модель данных служит инструментом проектирования, создания и использования баз данных и некоторых других информационных ресурсов [3].

Модели данных играют решающую роль в системах управления базами данных, поскольку они:

Руководство по проектированию базы данных. Модели данных помогают разрабатывать логические и физические конструкции баз данных, описывая, как данные будут организованы, тем самым устанавливая сущности, атрибуты и отношения.

Обеспечьте согласованность данных: определяя структуры и соглашения данных, модели данных обеспечивают единообразие во всех системах баз данных, тем самым поддерживая согласованность данных.

Обеспечьте эффективное хранение и поиск данных. Хорошо разработанная модель данных обеспечивает оптимальное хранение данных, более быстрые запросы и эффективный доступ к необходимой информации.

Улучшение коммуникации: модели данных служат визуальным и концептуальным справочником, который помогает устранить разрывы в общении между разработчиками, администраторами баз данных, участниками проекта и конечными пользователями. Они обеспечивают полное понимание предполагаемой структуры и использования базы данных.

Упрощение управления данными. Модели данных необходимы для управления данными, поскольку они помогают определить правила, ограничения и отношения, которые управляют данными, и помогают поддерживать качество и целостность данных [4].

За прошедшие годы было разработано несколько типов моделей данных. Каждый тип имеет свой набор преимуществ и недостатков, и их пригодность зависит от конкретного варианта использования. Основными типами моделей данных являются:

* Иерархическая модель данных - данные организованы в виде иерархии, где каждый элемент имеет одного родителя и может иметь несколько детей.
* Сетевая модель данных - данные организованы в виде графа, где каждый элемент может иметь несколько родителей и детей.
* Реляционная модель данных - данные хранятся в виде таблиц, где каждая строка соответствует записи, а каждый столбец - атрибуту.
* Модель «сущность-связь» - данные организованы в виде сущностей и связей между ними.
* Объектно-ориентированная модель данных - данные организованы в виде объектов, которые могут иметь свойства и методы.

## Исследование методов проектирования

БЛа бла бла

## Сценарий пользовательского интерфейса

Блу блу блу

## Исследование и обоснование выбора инструментальных средств разработки программного продукта

Инструментальные средства разработки программного продукта — это программы, приложения и другие средства, предназначенные для облегчения и ускорения процесса создания программного обеспечения. Эти средства включают в себя разнообразные инструменты, которые поддерживают разработчиков на различных этапах жизненного цикла программного продукта.

Существует множество языков программирования, которые могут быть использованы для написания приложений. Далее рассмотрены несколько из них, а также их преимущества и недостатки:

1. Java. Преимуществами данного языка программирования являются переносимость, многозадачность, богатая библиотека, широкое использование в корпоративной среде. Недостатком можно выделить не такую эффективность по производительности, как у некоторых нативных альтернатив [6].

2. C# используется в среде .NET, которая богата возможностями для разработки пользовательских интерфейсов (Windows Forms, WPF), интеграция с другими технологиями Microsoft. Недостатком является привязанность к платформе Windows, что ограничивает переносимость программного обеспечения [7].

3. C++. Преимуществами являются высокая производительность, низкоуровневое программирование, широкое использование в разработке игр и производительных приложениях. Требование более внимательного управления памятью, менее высокоуровневый синтаксис по сравнению с некоторыми языками являются недостатками данного языка [8].

4. Python. К преимуществам относится простота синтаксиса, обширные библиотеки (Qt, Tkinter), кроссплатформенность, быстрая разработка. В недостатки стоит отнести меньшую производительность по сравнению с компилируемыми языками [9].

По окончанию исследования достоинств и недостатков языков программирования для разработки программного обеспечения сделан выбор в пользу языка программирования Java и его фреймворка Spring Boot.

2 ПРАКТИЧЕСКАЯ ЧАСТЬ

* 1. Разработка базы данных

Для разработки базы данных

В таблице «Details» будет храниться информация обо всех деталях. Структура таблицы «Details», состоит из следующих полей, как показано в табл. 2.1.

Таблица 2.1 – Структура таблицы «Details»

|  |  |  |
| --- | --- | --- |
| **Наименование столбца** | **Тип данных** | **Ограничение** |
| Id | Числовой | Ключевое поле  Формат поля – 2 символа |
| Name | Текстовый | 256 символов |
| Articul | Текстовый | 256 символов |
| Note | Текстовый | 256 символов |

В таблице «Customers» будет храниться информация обо всех деталях. Структура таблицы «Customers», состоит из следующих полей, как показано в табл. 2.2.

Таблица 2.2 – Структура таблицы «Customers»

|  |  |  |
| --- | --- | --- |
| **Наименование столбца** | **Тип данных** | **Ограничение** |
| Id | Числовой | Ключевое поле  Формат поля – 2 символа |
| Name | Текстовый | 256 символов |
| Address | Текстовый | 256 символов |
| Phone | Текстовый | 256 символов |

В таблице «Suppliers» будет храниться информация обо всех деталях. Структура таблицы «Suppliers», состоит из следующих полей, как показано в табл. 2.3.

Таблица 2.3 – Структура таблицы «Suppliers»

|  |  |  |
| --- | --- | --- |
| **Наименование столбца** | **Тип данных** | **Ограничение** |
| Id | Числовой | Ключевое поле  Формат поля – 2 символа |
| Name | Текстовый | 256 символов |
| Address | Текстовый | 256 символов |
| Phone | Текстовый | 256 символов |

Синтаксис создания предоставления представлен ниже.

CREATE VIEW название\_представления [(столбец\_1, столбец\_2, ....)]

AS выражение\_SELECT

* 1. Технология программирования интерфейса программы
  2. Описание инструментов программирования.
  3. Тестирование программного продукта.

## Заключение

В рамках выполнения

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. <https://developers.sber.ru/help/sber-id/what-is-authentication>
2. <https://encyclopedia.kaspersky.ru/glossary/authorization/>
3. <https://rt-solar.ru/products/solar_inrights/blog/4518/>
4. <https://sendpulse.com/ru/support/glossary/authentication>
5. <https://ru.wikipedia.org/wiki/Аутентификация_в_Интернете>
6. <https://ekassir.com/blog/avtorizacziya-opredelenie-i-metody/?utm_source=chatgpt.com>