|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет  имени Н. Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н. Э. Баумана)** |

|  |  |
| --- | --- |
| ФАКУЛЬТЕТ | «Информатика, искусственный интеллект и системы управления» (ИУ) |

|  |  |
| --- | --- |
| КАФЕДРА | «Информационная безопасность» (ИУ8) |

**Курсовая работа**

**по дисциплине** «**Технологии и методы программирования**»

«Разработка консольного приложения для поиска города по IP адресу»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ8-34 |  |  |  | О. Ю. Спиридонов |
|  | (Группа) |  |  |  | (И. О. Фамилия) |
|  |  |  |  |  |  |
| Преподаватель: |  |  |  |  | А. В. Колесников |
|  |  |  |  |  | (И.О. Фамилия) |

Москва 2025г

**Содержание**

Введение 3

Выявление требований к среде разработки4

Ход работы 6

Тестирование приложения8

Заключение9

Список использованной литературы10

**Введение**

В современном мире, где информация становится все более доступной, а технологии развиваются с неимоверной скоростью, важность геолокации и возможности определения местоположения пользователя по IP-адресу становятся все более актуальными. Геолокация по IP-адресу позволяет не только улучшить пользовательский опыт, но и предоставляет множество возможностей для бизнеса, таких как таргетированная реклама, анализ аудитории и оптимизация контента.

**Выявление требований к среде разработки**

**Актуальность** выбранной темы обусловлена следующими факторами:

* **Рост интернет-пользователей**: с каждым годом количество пользователей интернета продолжает расти, что делает важным понимание их географического положения. Это позволяет компаниям адаптировать свои услуги и контент под конкретные регионы, улучшая пользовательский опыт
* **Персонализация контента**: геолокация по IP-адресу позволяет предоставлять пользователям более релевантный контент. Например, новостные сайты могут показывать местные новости, а интернет-магазины — предлагать товары, популярные в определенном регионе. Это способствует повышению вовлеченности пользователей и увеличению конверсии.
* **Безопасность и предотвращение мошенничества**: геолокация может использоваться для повышения безопасности онлайн-транзакций. Например, если пользователь пытается войти в систему из необычного местоположения, это может быть признаком мошеннической активности, что позволяет предпринять дополнительные меры безопасности.
* **Анализ аудитории**: определение местоположения пользователей по IP-адресу предоставляет ценную информацию для анализа аудитории. Это позволяет компаниям лучше понимать свою целевую аудиторию, выявлять тренды и адаптировать свои стратегии.

**Целью** данного курсового проекта является разработка консольного приложения на языке программирования C#, которое будет осуществлять поиск города по предоставленному IP-адресу с использованием API DaData. DaData — это сервис, предоставляющий различные инструменты для работы с данными, включая возможность определения местоположения по IP-адресу. Использование данного API позволит получить актуальную информацию о географическом местоположении пользователя, что может быть полезно в различных сферах, таких как маркетинг, аналитика и разработка программного обеспечения.

В рамках проекта будет разработано консольное приложение, которое будет запрашивать у пользователя ввод IP-адреса, отправлять запрос к API DaData и обрабатывать полученный ответ. Приложение будет использовать библиотеку HttpClient для выполнения HTTP-запросов и библиотеку Newtonsoft.Json для работы с JSON-форматом, что обеспечит простоту и удобство в обработке данных.

**Ход работы**

1. Изучение API DaData: ознакомление с документацией API, его возможностями и ограничениями, а также изучение формата запросов и ответов.
2. Создание структуры консольного приложения: определение архитектуры приложения, включая основные классы и методы, и реализация логики взаимодействия с пользователем, включая ввод IP-адреса и вывод результатов.

using System;

using System.Net.Http;

using System.Threading.Tasks;

using Newtonsoft.Json.Linq;

class Program

{

private static readonly HttpClient client = new HttpClient();

private const string apiKey = "0bbef86f96702509c74ef6e45f607d0598ac1066";

private const string apiUrl = "http://suggestions.dadata.ru/suggestions/api/4\_1/rs/iplocate/address?ip=";

static async Task Main(string[] args)

{

Console.WriteLine("Введите IP адрес:");

string ipAddress = Console.ReadLine();

var city = await GetCityByIp(ipAddress);

Console.WriteLine($"Город для IP адреса {ipAddress}: {city}");

}

private static async Task<string> GetCityByIp(string ipAddress){}

}

1. Выполнение HTTP-запросов: использование библиотеки HttpClient для отправки запросов к API DaData и формирование корректного URL для запроса, включая IP-адрес и токен доступа.

var response = await client.GetStringAsync($"{apiUrl}{ipAddress}&token={apiKey}");

1. Обработка JSON-ответов: использование библиотеки Newtonsoft.Json для парсинга JSON-ответов от API и извлечение необходимой информации, такой как название города, из полученных данных.

var json = JObject.Parse(response);

return json["location"]["data"]["city"].ToString();

1. Обработка ошибок: реализация механизма обработки возможных ошибок, проблемы с подключением к API или неверный формат IP адреса. Вывод информативных сообщений пользователю в случае возникновения ошибок, чтобы улучшить пользовательский опыт.

try

{

var response = await client.GetStringAsync($"{apiUrl}{ipAddress}&token={apiKey}");

var json = JObject.Parse(response);

// Проверяем найден ли город

try

{

return json["location"]["data"]["city"].ToString();

}

catch

{

Console.WriteLine("Ошибка: город не найден");

return "null";

}

}

catch

{

Console.WriteLine("Ошибка: не удалось получить ответ");

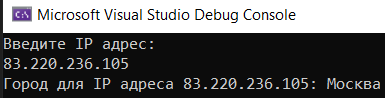
return "null";

}

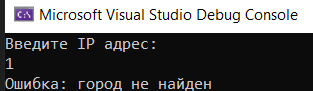
**Тестирование приложения**

На этапе тестирования программы проявилась проверка работоспособности программы в различных условиях.

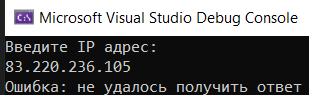
1. Верные входные данные.



1. Неверные входные данные.



1. Ошибка при выполнении запроса.



**Заключение**

В ходе выполнения курсового проекта на тему "Разработка консольного приложения для поиска города по IP адресу" была достигнута основная цель — создание функционального инструмента, позволяющего пользователям определять географическое местоположение по IP-адресу. Проект продемонстрировал важность геолокации в современном цифровом мире, где понимание местоположения пользователей становится ключевым фактором для бизнеса и улучшения пользовательского опыта.

В процессе разработки были рассмотрены и реализованы ключевые этапы, включая изучение API DaData, создание структуры приложения, выполнение HTTP-запросов, обработку JSON-ответов и реализацию механизма обработки ошибок. Эти этапы не только способствовали созданию работоспособного приложения, но и позволили углубить знания в области программирования, работы с API и обработки данных.

Тестирование приложения подтвердило его корректность и надежность, что является важным аспектом для любого программного продукта. Кроме того, проект предоставил возможность развить навыки работы в команде, управления временем и решения проблем, что является неотъемлемой частью процесса разработки программного обеспечения.

В заключение, разработанное консольное приложение не только отвечает современным требованиям пользователей и бизнеса, но и служит отличной основой для дальнейшего изучения и развития в области программирования и информационных технологий. В будущем возможно расширение функционала приложения, включая интеграцию с другими сервисами и улучшение пользовательского интерфейса, что сделает его еще более полезным и востребованным.

**Список использованной литературы**

1. Руководство по использованию API: город по IP-адресу

<https://dadata.ru/api/iplocate/>

1. Документация C#

<https://learn.microsoft.com/ru-ru/dotnet/csharp>/

1. Документация .NET

<https://learn.microsoft.com/en-us/dotnet>

1. Официальный сайт JSON

<https://www.json.org/json-en.html>

**Приложение**

Исходный код проекта размещён на GitHub:

<https://github.com/AlanOleg/AL-Sem-3-Course-Project>