1. **WRITE A SHORT NOTE ON THE EVOLUTION OF.NET FRAMEWORK AND C# (100 WORDS)**

In late 1990s, Microsoft sought a modern .NET platform and its own object-oriented language. C#, born from C++, Java, and Delphi, simplified Windows development. With rich libraries and tools, .NET empowered secure, efficient apps.

C# evolved with features like generics, LINQ, and async, becoming more expressive and efficient. .NET embraced cross-platform with .NET Core, then unified into a single platform with .NET 5 and 6.

From simple beginnings to versatile tools, C# and .NET's journey continues, empowering developers to build diverse applications.

1. **EXPLAIN THE FOLLOWING TERMS:**

MONO

XAMARIN

COM

.NET CORE

UNITY C#

REST

1. **Mono:** Mono is an open-source implementation of the .NET Framework. It allows developers to build and run .NET applications on various platforms, including Linux, macOS, and Windows.
2. **Xamarin:** Xamarin is a platform for building cross-platform mobile applications using C#. It allows developers to write code once and deploy it across multiple platforms, including iOS, Android, and Windows.
3. **COM:** COM stands for Component Object Model. It is a binary-interface standard for software components introduced by Microsoft. COM allows software components to communicate with each other and provides a way for applications to interact with system services.
4. **.NET Core:** .NET Core is an open-source, cross-platform framework for building modern, cloud-based applications. It is a modular framework that allows developers to use only the components they need for their applications and runs on Windows, macOS, and Linux.
5. **Unity C#:** Unity C# refers to the programming language C# used in the Unity game engine. Unity allows developers to create 2D and 3D games and interactive experiences, and C# is the primary language used for scripting in Unity.
6. **REST:** REST stands for Representational State Transfer. It is an architectural style for designing networked applications. RESTful systems use standard HTTP methods like GET, POST, PUT, and DELETE to perform CRUD (Create, Read, Update, Delete) operations on resources. It emphasizes scalability, reliability, and simplicity in distributed systems.
7. **CRITICALLY EXPLAIN ANY THREE KEY FUNCTIONS OF THE CLR.**

1. **Memory Management:** The CLR manages memory allocation and deallocation, ensuring efficient memory usage by automatically reclaiming unused memory through garbage collection.

2. **Exception Handling:** CLR provides robust exception handling mechanisms, enabling developers to gracefully handle runtime errors and exceptions, ensuring the stability and reliability of applications.

3. **Security:** CLR enforces code access security policies, verifying the permissions required for code execution, preventing unauthorized access and protecting against malicious code execution.

**REFERENCES**

Mohammadnezhad, A. (2023, June 26). Unveiling the untold story: The evolution of C# and .NET. LinkedIn. <https://www.linkedin.com/pulse/unveiling-untold-story-evolution-c-net-ali-mohammadnezhad#:~:text=NET-,C%23%20and%20.,potential%20of%20the%20Windows%20platform>.

According to OpenAI's ChatGPT (OpenAI, 2024, February 1) [Explanation of Mono, Xamarin, COM, .Net Core, Unity C#, REST].

ChatGPT, personal communication, (February 1, 2024). Three key functions of the CLR