



Armen Melkumyan • 1st
Technical / Solutions Architect
1yr • Edited •

...

🔥 C# .NET Interview Puzzle: Implement a Rate Limiter 🔥

Why It Matters:

Real-World Application: Rate limiting is crucial for maintaining the reliability and security of web services.

Skill Test: Evaluates your proficiency with C# and .NET, focusing on collections, concurrency, and system design.

Your Mission:

Design a thread-safe rate limiter using native .NET libraries.

Ensure efficiency, handling high volumes of requests with minimal overhead.

Hints Towards the Solution:

Data Structures: Consider using a Dictionary to map users to their request timestamps.

Concurrency: Utilize lock statements or concurrent collections for thread safety.

Algorithm Efficiency: Implement an algorithm that efficiently checks and cleans up old request data to prevent memory bloat.

Github link: <https://lnkd.in/dDcD8GxZ>

[#CSharp](#) [#DotNET](#) [#SoftwareEngineering](#) [#WebDevelopment](#) [#ChallengeAccepted](#)

```
RateLimiter.cs

public class RateLimiterThreadSafe
{
    private readonly Dictionary<string, Queue<DateTime>> userRequests = new Dictionary<string, Queue<DateTime>>();
    private readonly int requestLimit;
    private readonly TimeSpan timeSpan;
    private readonly object lockObj = new object();

    public RateLimiterThreadSafe(int limit, TimeSpan period)
    {
        this.requestLimit = limit;
        this.timeSpan = period;
    }

    public bool RequestShouldBeAllowed(string userId)
    {
        return false; // Request denied
    }
}
```

codetoint.com