In this task, you need to write a simple HTTP Denial-of-Service protection system.
You may implement it using C# or Java. C++ is also fine, but might take considerably more time.
You may use any libraries that will make your code cleaner, or better in any way.
You need to create two processes (one for the client and one for the server).

Client

The user enters the number of HTTP clients to simulate.

For each HTTP client you will run a separate simultaneous thread/task which will do the following in a loop:

- Send HTTP request to a server with simulated HTTP client identifier as a query parameter (e.g.http://localhost:8080/?clientId=3).
- The client identifiers are random integers between 1 to the number of clients. Different clients can have the same identifier.
- Wait some random time and then send another request.
- The client will run until key press after which it will gracefully drain all the threads/tasks (wait for all the of them to complete) and will exit.

Server

Starts listening for incoming HTTP requests.

For each incoming HTTP request you will do the following:

- Handle the request in a separate thread/task.
- Check if this specific client reached the max number of requests per time frame threshold (no more than 5 requests per 5 secs).
- If the client hasn't reached the threshold, it will get HTTP response with status code 200 (OK) otherwise status code 503 (Service Unavailable).
- Note: The time frame starts on each client's first request and ends 5 seconds later.
- After the time frame has ended, the client's first request will open a new time frame and so forth.
- The server will run until key press after which it will gracefully drain all the threads/tasks and will exit.

General notes

Pay attention to thread safeness.