Backend Assessment

We'd like you to complete our Backend Coding Challenge. *It's intentionally broad to give you freedom to approach it in your own way. It's much more important for us to be able to follow your decision making process, rather than the perfect finished result.*

Once the challenge has been completed, we'll arrange a 90 minute interview to talk through the challenge and assess our other key BE skills - AWS, Scalability, JS, Resilience, DevOps, Performance, Security, and Software Engineering Practices.

<Functional Requirements/>

Build a service in Node.js that exposes an API which can be consumed from any client. This service must check how many video streams a given user is watching and prevent a user from watching more than 3 video streams concurrently. Using services like API Gateway, ECS, ALB etc will be a huge bonus although this is not a requirement.

<Output Requirements/>

The solution must be provided as a Git repository including full commit history. Please commit **frequently** (**DO NOT DO ONE SINGLE COMMIT INTO THE REPO)** so that we can follow your implementation and see the milestones of your implementation. You are more than welcome to share your repository on a free service such as GitHub or BitBucket.

The repository must contain:

- Source code
- **Dockerfile**
- Any installation and deployment instructions
- README file with
    - Steps to use/test the service (we must be able to run the service locally)
    - A brief explanation on the scalability strategy

<Description/>

We're looking to see how you approach a problem with a broad spec which could have a number of different solutions. Keep the implementation simple, but make sure you have automated tests, logging, and include information in the README about how you'll scale the solution to millions of users, how you'd approach logging & monitoring at scale so that you can actually debug the system as it increases in complexity.

We are not expecting the solution to be deployed, but we expect you to understand the process and best practices around the deployment process. It's enough if you could provide to our engineers a clear, easy instruction step to make your node application run locally.

We'll talk through the challenge with you in the final interview, exploring possible areas of improvement.

<FAQ/>

Any client - what are the clients?
A client is a consumer of the API (e.g. web app, another backend service, a mobile app, etc).
In this case "Any client" means the API should be implemented independently of who/what is going to consume it.

Will this service stream any video?
No, another service would be responsible for streaming video. This service only checks whether there are too many streams active. In a real implementation, it's likely that the streaming service would call this API to check whether the stream can begin.

How is authentication performed?
You do not need to implement any authentication or authorisation.

How will the service be able to find out about video streams currently being watched? Is there a service that provides this information?
This API will be involved every time a new user wants to watch new content, so it should be able to know exactly how many videos a user is watching.

Should the service be on a public network?
Ideally (less work for us), but not required. We're happy to start a Docker image ourselves.

Should I build any deployment pipelines?
If you want to deploy your service, do it manually - it's a one-time thing.

Can I use npm packages and libraries?
Yes, but please use as few frameworks as possible.

Can we implement this using AWS?
Sure! Feel free to show off your AWS skills.

How long should I spend on the challenge?
Around 3 hours is usually enough if you take longer it’s not a problem. We're not looking for a perfect solution here, so don't worry if some areas aren't finished.

If you don't finish, please note what work you'd do to finish it in the README.