Below is the instruction for the offline coding try-out.

Long-running job

- Bare minimum requirements:
    - Provide us with your example of the simple SPA with .Net 6 on the backend, and Angular/ReactJS on the front-end. On the front-end side user should be able to enter the text into the text field, press the "Convert" button, and get this text encoded into the base64 format.
    Encoding should be performed on the backend side. Encoded string should be returned to the client one character at a time, one by one, and for each returned character there should be random pause on the server 1-5 seconds. All received characters should form a string in a UI textbox, hence it will be updated in real-time by adding new incoming characters. User cannot start another encoding process while the current one is in progress, but user can press the "cancel" button and thus cancel the currently running process.
- Bonus (recommended) requirements:
    - Web page should look neat; use Bootstrap or its derivatives
    - Use default IoC, package managers and other tools to build & run the app
    - Use the latest released .Net & C# with all possible new features they provide
    - Server-side app should be hosted in Linux Docker container
        - Host API & UI backend in different containers
        - Support basic authentication using nginx in another container
    - Business logic should be implemented as a service, with unit tests for each respectively
    - No compiled/build stuff should be provided, only source code is required

Example of the app functioning:

- input = "**Hello, World!**". Generated base64="**SGVsbG8sIFdvcmxkIQ==**"
    - What web client receives from the server:
        - Random pause… "**S**"
        - Random pause… "**G**"
        - Random pause… "**V**"
        - Random pause… "**s**"
        - Etc.
    - What does user see in the result text field on web UI:
        - Random pause… "**S**"
        - Random pause… "**SG**"
        - Random pause… "**SGV**"
        - Random pause… "**SGVs**"
        - Etc.