Resmed - Take home exercise – AWS Developer

Guidelines

Estimated Completion Time :

- The exercise is designed to take approximately 4 hours.
- If you wish to explore the task in greater depth, feel free to do so, we always appreciate extra effort!

Submission Timeline :

- You will have 5 calendar days to complete the assignment, typically including a weekend.
- If you anticipate any scheduling conflicts or constraints that may affect your ability to complete the task within this timeframe, please reach out to the Talent Acquisition Partner as soon as possible. We are happy to discuss a revised deadline if needed.

Use of AI Tools :

- The use of AI tools (e.g., ChatGPT, GitHub Copilot, etc.) is permitted.
- However, please document the prompts or queries you used and be prepared to discuss them during the interview. Transparency is key : we are interested in your thought process as much as the final result.

Use of Paid Services :

- You should not incur any costs to complete this assignment.
- Please use free-tier services or open-source tools where applicable.

Goal:

As a university professor, I want a simple way to share resource files with my students. Yourtask is to build a lightweight file-sharing API that supports uploads and downloads, storesfiles on disk, and makes it easy to retrieve shared files via a simple HTTP endpoint.

Requirements:

Build a minimal file sharing service with the following functionality:

Endpoints

- POST : Accepts a file upload (multipart/form-data). Returns a unique file ID or name usedto retrieve it later.
- GET : Downloads the file associated with the given ID.
- GET : Lists all uploaded files (metadata only - no content).

File Metadata

- File name
- Size (in bytes)
- Upload timestamp
- Unique ID

Storage

- Store file content
- Store metadata

Constraints

- No authentication required.
- Max file size: 20MB.

Tasks:

- What happens if something goes wrong during a request? How does the API communicate this to a client?
- How can you confirm the code works?
- How can someone else run and test the API quickly?

Submission:

- All source code
- Any notes on assumptions or design decisions