# Scenario

"You're working on a CMS-backed portal that needs to expose certain pieces of content via an API. Editors will update the content in Sitecore or Umbraco, and the content must be cached and delivered quickly. You're tasked with building an API endpoint that exposes content in a user-friendly format, ensures freshness, and includes basic access control logic for different user types."

## Tasks

Choose 2 or 3

|  |
| --- |
| * Extend the provided API to retrieve and return content from a mock CMS (JSON store or in-memory DB). Add optional caching with expiry. |
| * Propose a simple architecture for handling high traffic and CMS sync. Could be a markdown diagram or short doc. |
| * Add unit tests for the logic (e.g., access control, cache fallback, error handling). |
| * Handle a scenario where the CMS is down. What should the API do? How would you test this? |
| * Suggest how this might be deployed in Azure to scale, be monitored, and be secure (document, not implement). |

## Submission Expectations

* GitHub repo (or zip)
* Readme with assumptions, trade-offs, and what you didn’t do
* Expected time: 2-3 hours