Homework 3 – Application Telemetry with Azure Application Insights

🎯 Objective

Attach Azure Application Insights to a web application and use it to collect, analyze, and visualize runtime telemetry. The focus of this homework is observability, not application complexity.

Students who completed Homework 2 may reuse their existing application. Students who did not complete Homework 2 may deploy a simplified web app with mock or empty endpoints. No database is required.

🧩 Assignment Requirements

1. Application Insights Integration

Your application must:

- Have Azure Application Insights enabled and connected to the web app.
- Use the Application Insights SDK suitable for your technology stack.
- Successfully send telemetry that is visible in the Azure Portal.

2. Request Performance Telemetry

Your application must emit performance telemetry for every HTTP request.

Each request must include:

- Request timestamp
- HTTP method (GET, POST, etc.)
- Request path / endpoint
- Response status code
- Request duration (in milliseconds)

This telemetry must be visible in Application Insights Logs.

3. Business logging

You should manually log whenever an item is successfully added (something simple, e.g. “Item successfully added”).

Your application must include explicit error handling and error logging.

You must provide an easy way to trigger an error, for example:

- Adding a duplicate item
- Invalid input (you can invent a rule and specify it in README)

Errors must be recorded in Application Insights and return an appropriate HTTP status code.

4. Frontend Custom Event

Your application must emit a custom Application Insights event from the frontend when the user clicks the "Add" button.

The event must:
- Be sent from client-side code
- Appear as a custom event in Application Insights Logs
- Include at least one custom property (e.g., button name or action type)

5. Health Endpoint

Your application must expose a health endpoint (for example: /health or /ping).

Requirements:
- The endpoint must return HTTP 200 when the app is healthy
- Requests to this endpoint must appear in Application Insights
- Request duration for this endpoint must be observable in telemetry

6. Portal Verification

Using Application Insights Logs, you must demonstrate that you can:
- View successful and failed requests
- Observe request durations
- Identify requests to the health endpoint
- See frontend custom events

📦 Deliverables

You must upload on your GitHub repository and submit a ZIP file containing:

1. Source Code
- Application source code
- Application Insights configuration

2. README.md
- Public URL of the application
- Description of collected telemetry
- Explanation of how to trigger an error
- At least two KQL queries used to analyze telemetry
- Screenshot(s) of Application Insights Logs and dashboard

📨 Submission Instructions

Please send me the link to your repo and the zip file on Teams.
Deadline: 7 January 2026, 23:59