Project Report

Save-Up: GAME-DAY NOTIFICATION

VAASU GARG (E22CSEU0558)  
VIDUSHI SINGH (E22CSEU1495)

A report submitted in part fulfilment of the degree of

**B.Tech in Computer Science Supervisor:** Dr. Naween Kumar



School of Computer Science Engineering and Technology Bennett University

April 30, 2025

**Abstract**

The Game Day Notification System is an automated service that provides real-time updates on basketball games scheduled for the day. Whether games are completed, currently live, or yet to start, the system collects this data and delivers it to users via email using AWS services. It’s a lightweight, serverless solution tailored for sports fans who want quick updates without manually tracking multiple sources.

**1. Introduction**

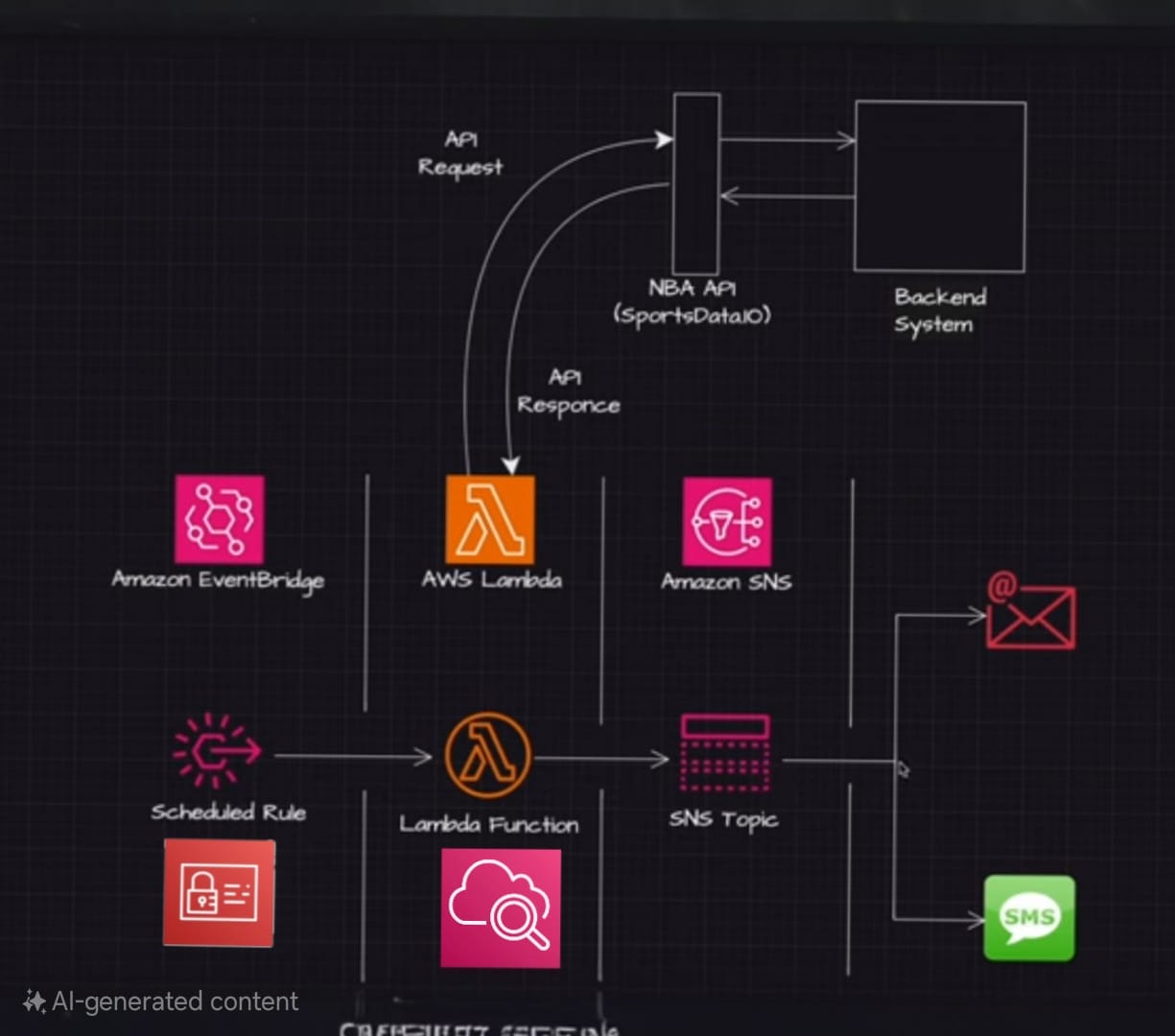
Staying informed about daily basketball games can be overwhelming due to scattered sources and varying formats. This project addresses that gap by offering a centralized, automated, and readable update service for NBA games. The system is built using a suite of AWS services to ensure reliability, scalability, and ease of maintenance.

**2. Objective**

To design and deploy an automated notification system that:

* Retrieves basketball game data for the current day.
* Classifies games as **Completed**, **Live**, or **Upcoming**.
* Formats the data for human readability.
* Sends updates to subscribers via email using AWS SNS.

**3. System Architecture**

****

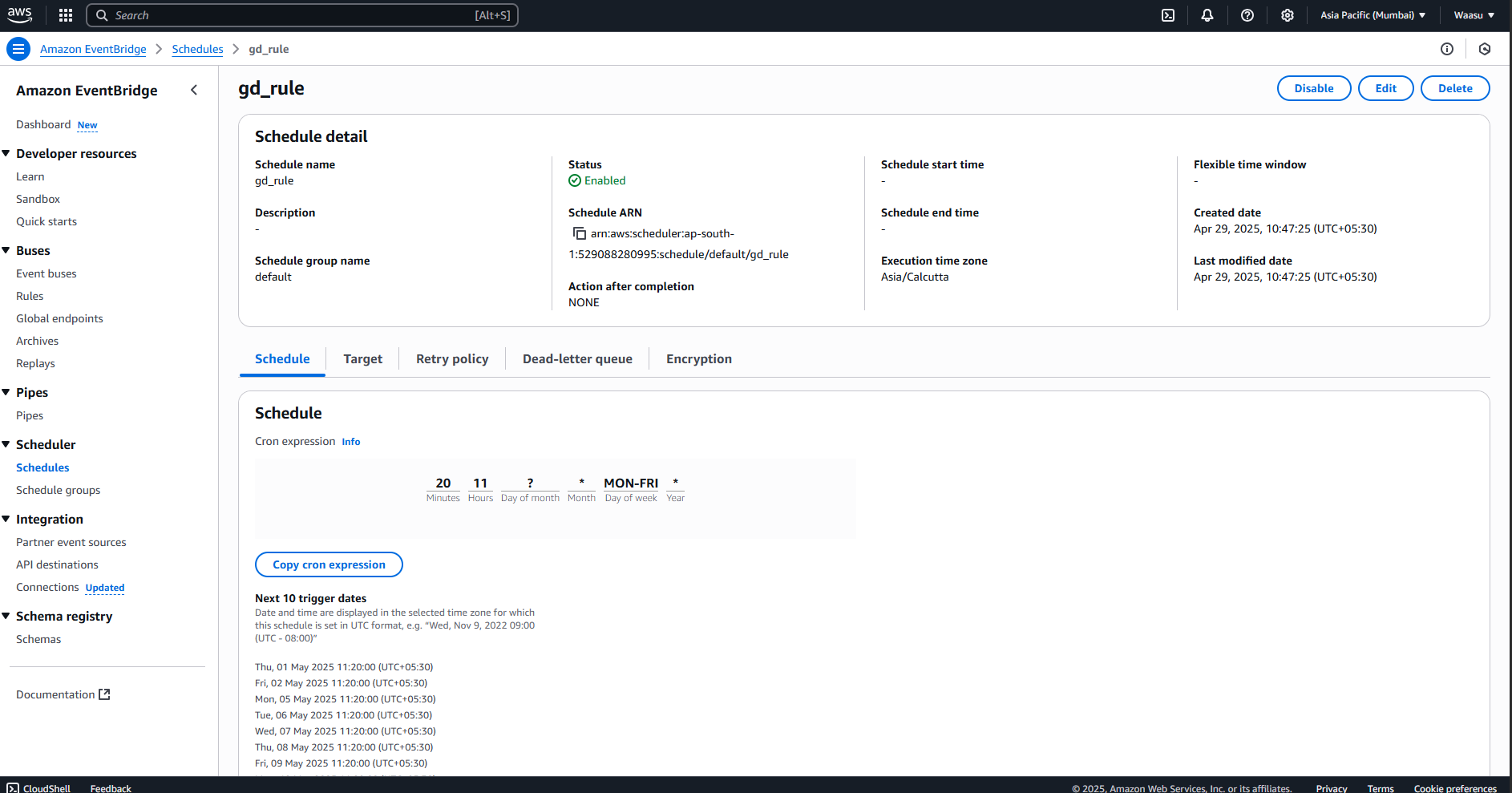
**Architecture**

**Services Used:**

* **AWS Lambda** – Executes the core logic to fetch and process NBA game data.
* **AWS EventBridge** – Schedules the Lambda according to the given schedule.
* **AWS SNS** – Sends formatted notifications via email.
* **AWS CloudWatch** – Handles monitoring, logging, and debugging.
* **AWS IAM** – Manages secure access via roles and policies for Lambda, EventBridge, and SNS.

**4. Workflow**

 Triggering: AWS EventBridge triggers the Lambda function daily at a specified time.





EventBridge rule

 DataRetrieval: The Lambda function calls an external NBA game API, which returns data for all basketball games scheduled for the current day in JSON format.



JSON DATA

 Data Extraction: The Lambda function parses the JSON response and filters out only the relevant information such as:

* Game status (e.g., Final, Live, Scheduled)
* Team names
* Final or current score
* Start time
* Channel
* Quarter-wise scores (if available)

 Notification: The extracted information is compiled into a structured text format and published to an AWS SNS topic, which sends it via email to subscribers.

**5. Data Handling**

 API Response: The external API provides a comprehensive JSON payload containing all scheduled, live, and completed games for the day.

 Lambda Logic:

* Filters the games based on the current date.
* Selectively extracts necessary fields such as game status, team names, scores, timing, channel, and quarters.
* Ignores unnecessary metadata to ensure the final output is concise and relevant.

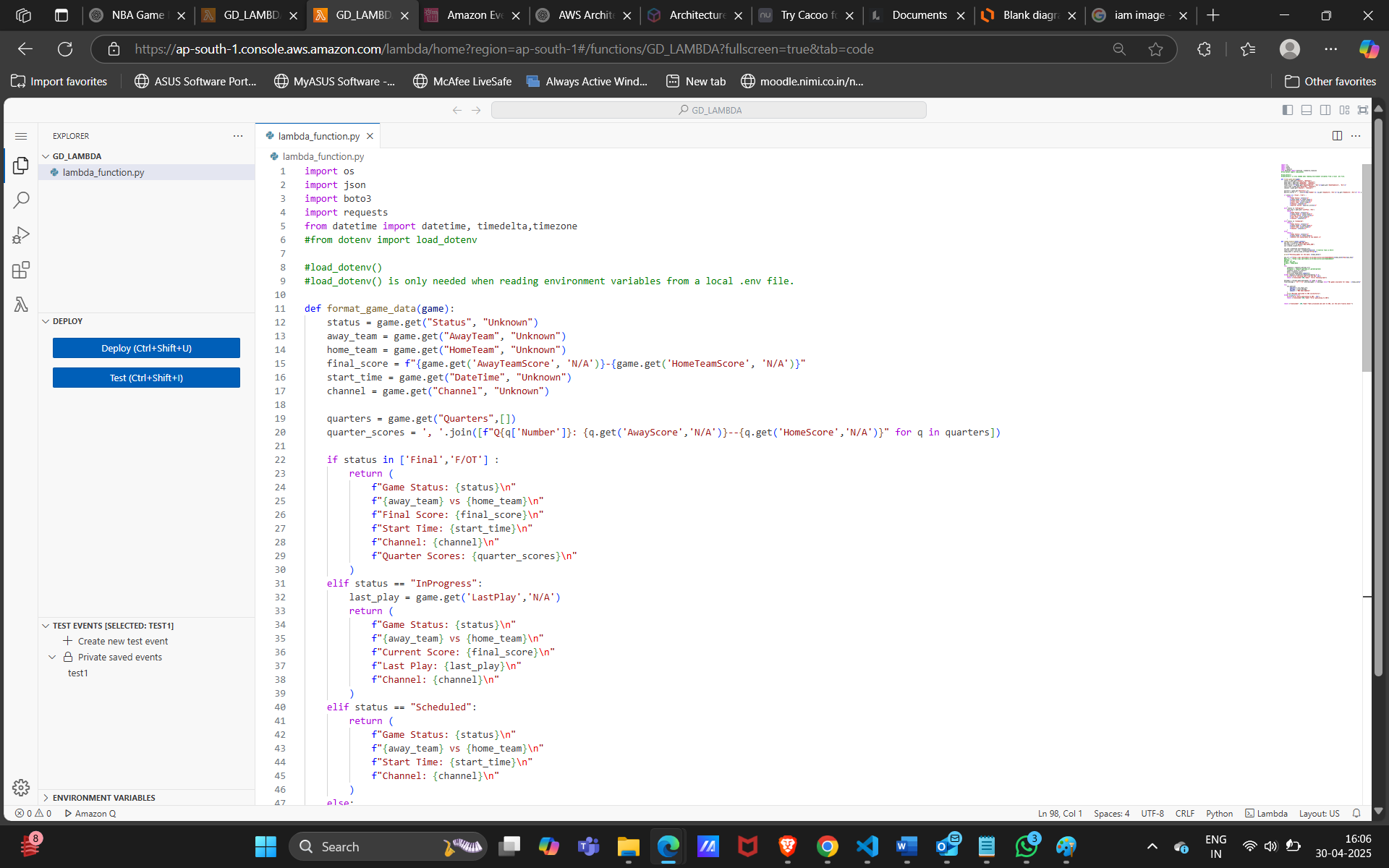


Figure 1:- Lambda logic-1

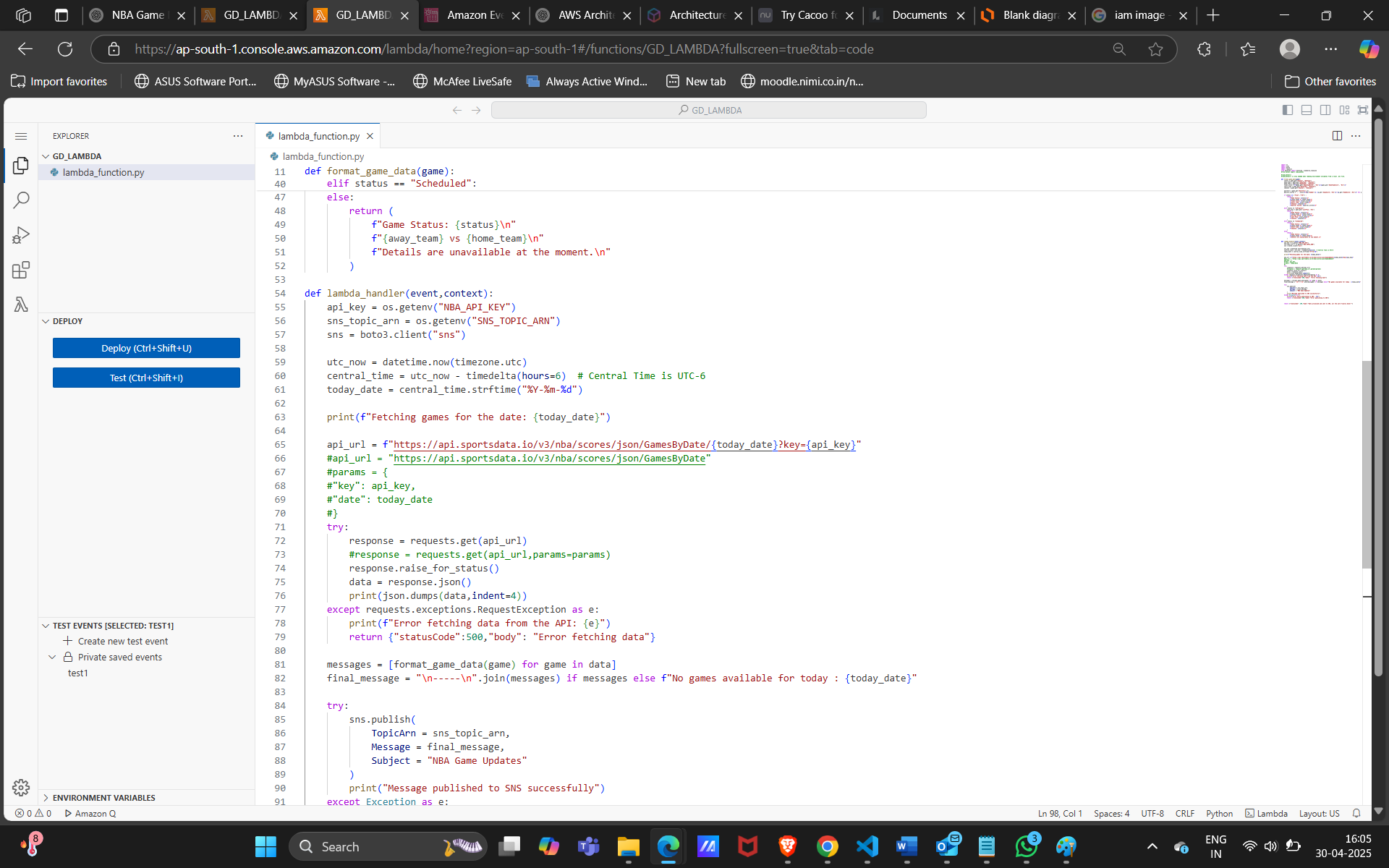
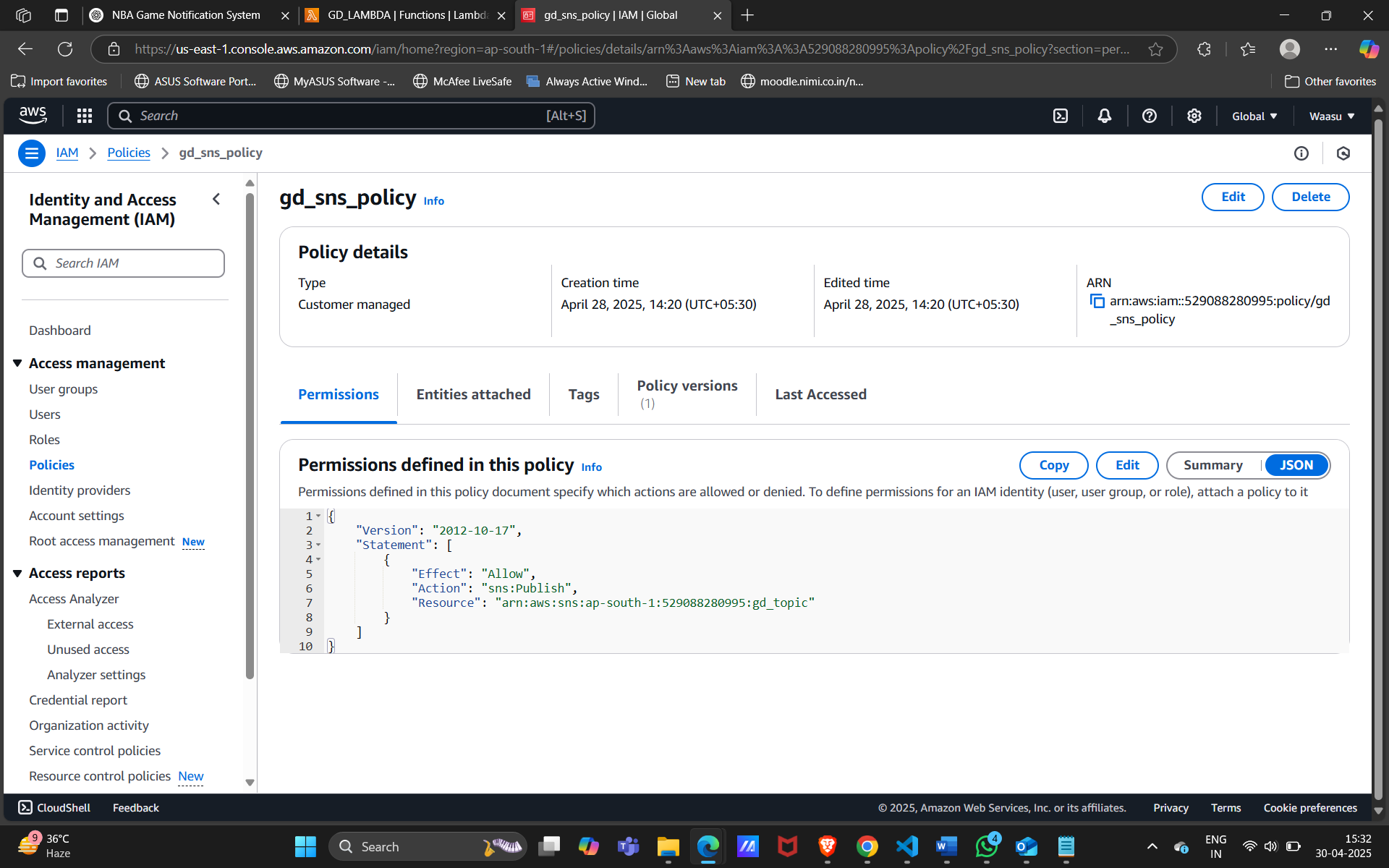


Figure 1:- Lambda logic-2

**6. IAM Roles & Policies**

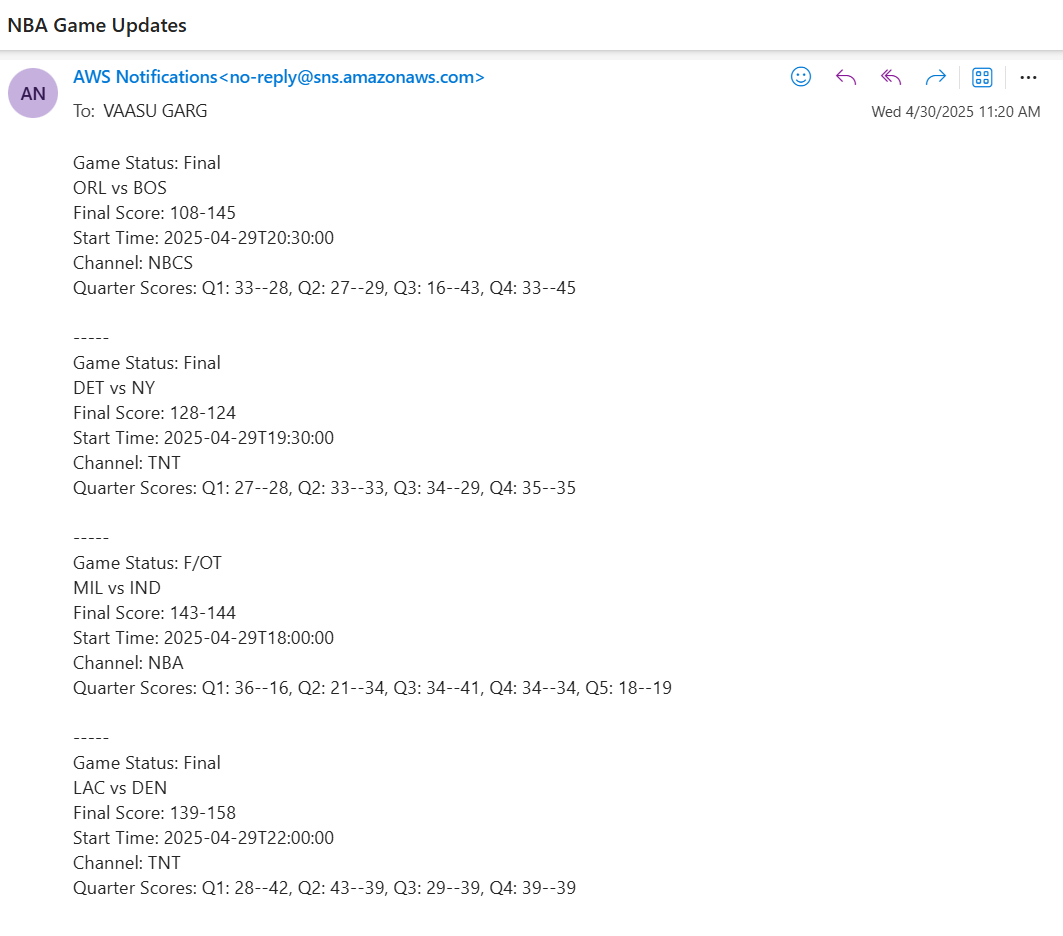
* A custom IAM Role was created for the Lambda function with:
  + Permissions to execute Lambda
  + Access SNS to publish messages
  + Write logs to CloudWatch
* EventBridge and CloudWatch also had scoped IAM policies to maintain least privilege.





**7. Sample Output (Email Format)**

**Game Status: Final  
ORL vs BOS  
Final Score: 108-145  
Start Time: 2025-04-29T20:30:00  
Channel: NBCS  
Quarter Scores: Q1: 33--28, Q2: 27--29, Q3: 16--43, Q4: 33—45**

****

**8. CloudWatch & Monitoring**

* **Logging**: Lambda logs execution flow, errors, and runtime metrics.
* **Alerts**: Can be integrated for failures (future enhancement).
* **Debugging**: Used log groups to troubleshoot JSON parsing and format issues.

**9. Challenges Faced**

* **Dynamic Game Status**: Ensuring correct classification when games are ongoing or delayed.
* **Timezone Adjustments**: Start times had to be validated or converted for accurate filtering.
* **Email Clarity**: Designed a simple readable format to avoid information overload.

**10. Future Enhancements**

* User-specific subscriptions (e.g., updates for favorite teams only)
* Push notifications or SMS via SNS/Firebase
* Web-based interface for preference management

**11. Conclusion**

The Game Day Notification System is a robust, serverless solution that brings together multiple AWS services to deliver timely basketball game updates. By automating the end-to-end pipeline from data fetch to human-readable notification, it serves as an efficient tool for sports fans and demonstrates the power of cloud-native architecture.