## Cloud-Based Healthcare Management System with AWS FHIR

*A Course Project Report Submitted in partial fulfillment of the course requirements for the award of grades in the subject of*

# CLOUD BASED AIML SPECIALITY (22SDCS07A)

by

**V.Thanmayee 2210030072**

### Under the esteemed guidance of

**Ms. P. Sree Lakshmi**

Assistant Professor,

Department of Computer Science and Engineering



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**K L Deemed to be UNIVERSITY**

*Aziznagar, Moinabad, Hyderabad, Telangana, Pincode: 500075*

April 2025

## K L Deemed to be UNIVERSITY



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

*Certificate*

This is Certified that the project entitled **“Cloud-Based Healthcare Management System with AWS FHIR”** which is a project work carried out by Vempati Thanmayee (2210030072), in partial fulfillment of the course requirements for the award of grades in the subject of **CLOUD BASED AIML SPECIALITY**, during the year **2024-2025**. The project has been approved as it satisfies the academic requirements.

**Ms.P.Sree Lakshmi Dr. Arpita Gupta**

**Course Coordinator Head of the Department**

**Ms. P. Sree Lakshmi Course Instructor**

**CONTENTS**

Page No.

1. [Introduction 1](#_TOC_250005)
2. AWS Services Used as part of the project 2
3. [Steps involved in solving project problem statement 3](#_TOC_250004)
4. [Stepwise Screenshots with brief description 4](#_TOC_250003)
5. [Learning Outcomes 9](#_TOC_250002)
6. [Conclusion 10](#_TOC_250001)
7. [References 11](#_TOC_250000)

# INTRODUCTION

The healthcare industry is undergoing rapid digital transformation to improve patient care, enhance data accessibility, and achieve interoperability between healthcare systems. One of the major challenges in this domain is the efficient and secure management of large volumes of healthcare data while ensuring compliance with regulatory standards.

Fast Healthcare Interoperability Resources (FHIR) is a widely adopted standard that facilitates the secure and standardized exchange of healthcare information. It ensures data interoperability and compliance with regulations such as HIPAA (Health Insurance Portability and Accountability Act)[1].

This report presents a Cloud-Based Healthcare Management System developed using AWS services—**Amazon S3, AWS Lambda, Amazon DynamoDB, and API Gateway**— designed to manage and exchange healthcare records in a FHIR-compliant format. The system features a web interface hosted on S3 for uploading patient records, which are processed through Lambda and stored securely in DynamoDB, with API Gateway handling external communications.

The adoption of these AWS services offers key benefits such as:

* **Scalability**: Efficient handling of growing patient data using DynamoDB and Lambda [2].
* **Security & Compliance**: Data protection through IAM roles and encrypted storage [3].
* **Interoperability**: FHIR-compliant architecture ensures standardized communication.
* **Cost Efficiency**: Implementation using the AWS Free Tier to minimize development costs [4].

## AWS Services Used

The following AWS services are used in the project:

* **Amazon DynamoDB**: Used for storing structured healthcare records in a NoSQL format, ensuring fast and reliable data access.
* **Amazon S3 (Simple Storage Service)**: Used for storing static assets such as documents, reports, and patient-related images.
* **AWS Lambda**: Serverless functions used to handle business logic, data processing, and integration with the FHIR standard.
* **Amazon API Gateway**: Provides RESTful endpoints for external clients to interact securely with the backend services.

## Steps Involved in Solving Project Problem Statement

**Step 1: Setting Up AWS Free-Tier Account**

* Create a new AWS account under the free-tier plan.
* Enable billing alerts to monitor usage and avoid unexpected charges.
* Configure IAM users and roles for secure access to AWS services.

**Step 2: Setting Up Amazon DynamoDB and Amazon S3**

* Create a DynamoDB table (e.g., PatientRecords) with partition key patientID.
* Create an S3 bucket to store additional healthcare files like prescriptions and scan reports.
* Apply appropriate IAM roles to control access between services.

**Step 3: Writing and Deploying AWS Lambda Functions**

* Develop Lambda functions to handle patient data (CRUD operations) and transform it to FHIR-compliant JSON.
* Use AWS SDKs (e.g., Boto3 for Python or AWS SDK for Node.js) to interact with DynamoDB and S3.
* Configure timeout, memory, and environment variables for each function.

**Step 4: Exposing the FHIR API with AWS API Gateway**

* Create an HTTP REST API using API Gateway.
* Link API endpoints (e.g., /patients, /records) to corresponding Lambda functions.
* Deploy the API and enable CORS for frontend integration.
* Test endpoints using Postman or curl with sample FHIR-compliant payloads.

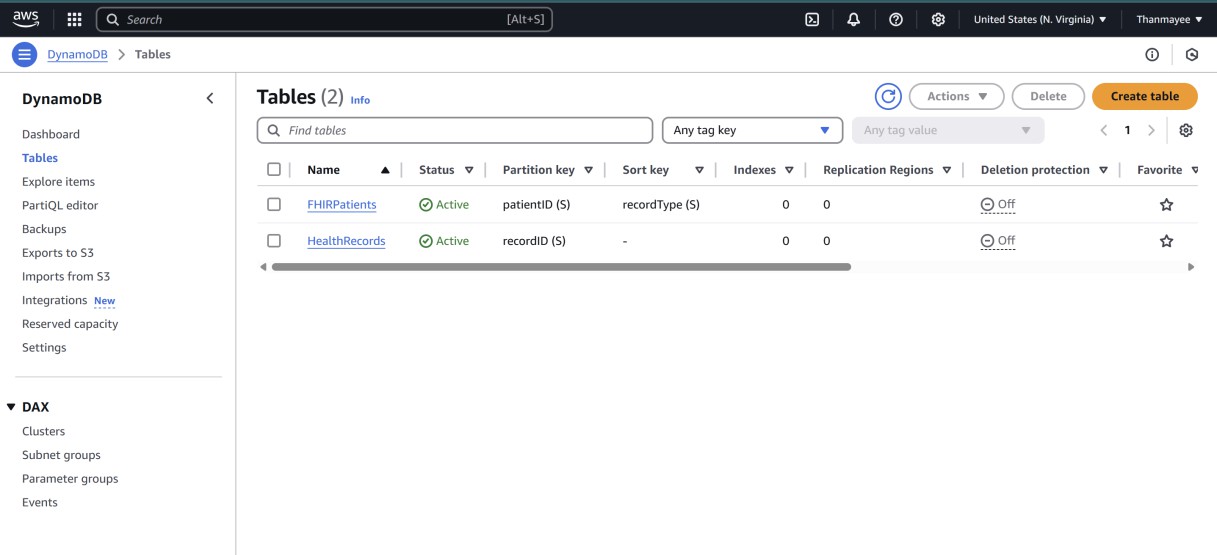
**Step 5: Frontend Development**

* Use React.js to build a simple dashboard for patients and doctors.
* Create forms and tables to visualize and submit healthcare records.
* Connect the frontend to API Gateway endpoints using Axios or Fetch API.

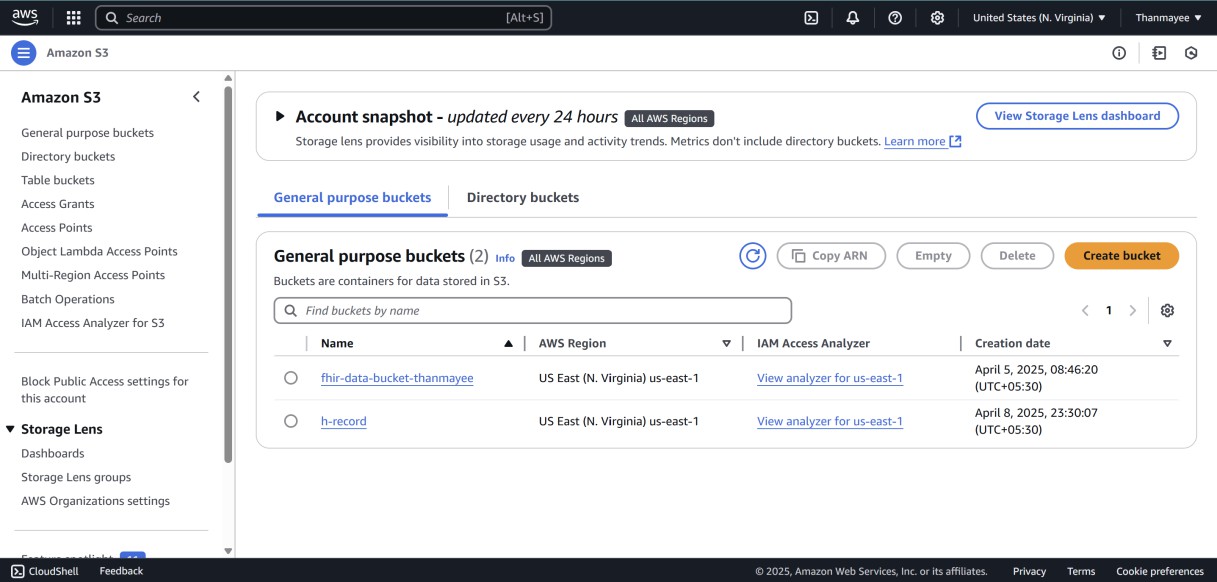
**Step 6: Monitoring and Security Enhancements**

* Enable AWS CloudWatch for logs, monitoring Lambda executions and errors.
* Use AWS WAF (Web Application Firewall) to protect against common web exploits.
* Define strict IAM policies to control access to DynamoDB, Lambda, and S3 resources.

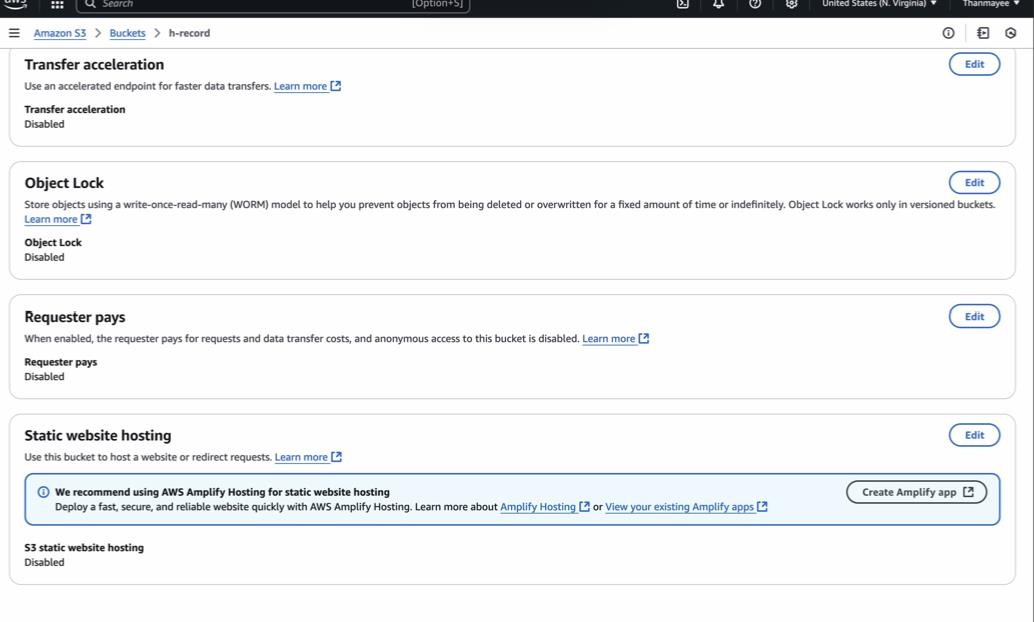
## Stepwise Screenshots with Brief Description

****

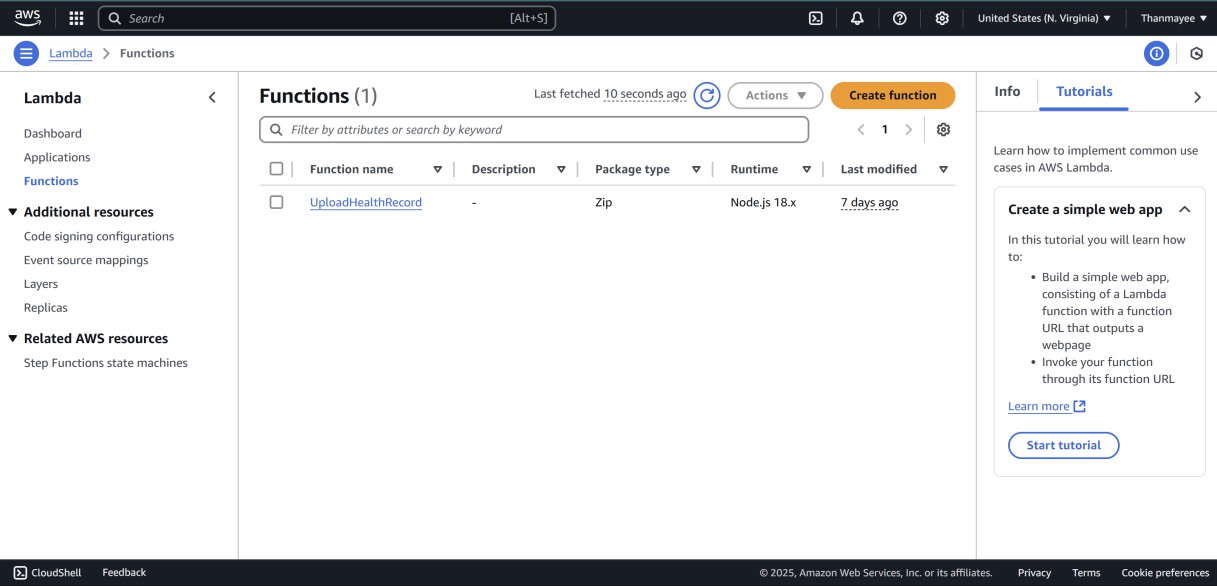
**Fig 4.1** HealthRecords table created



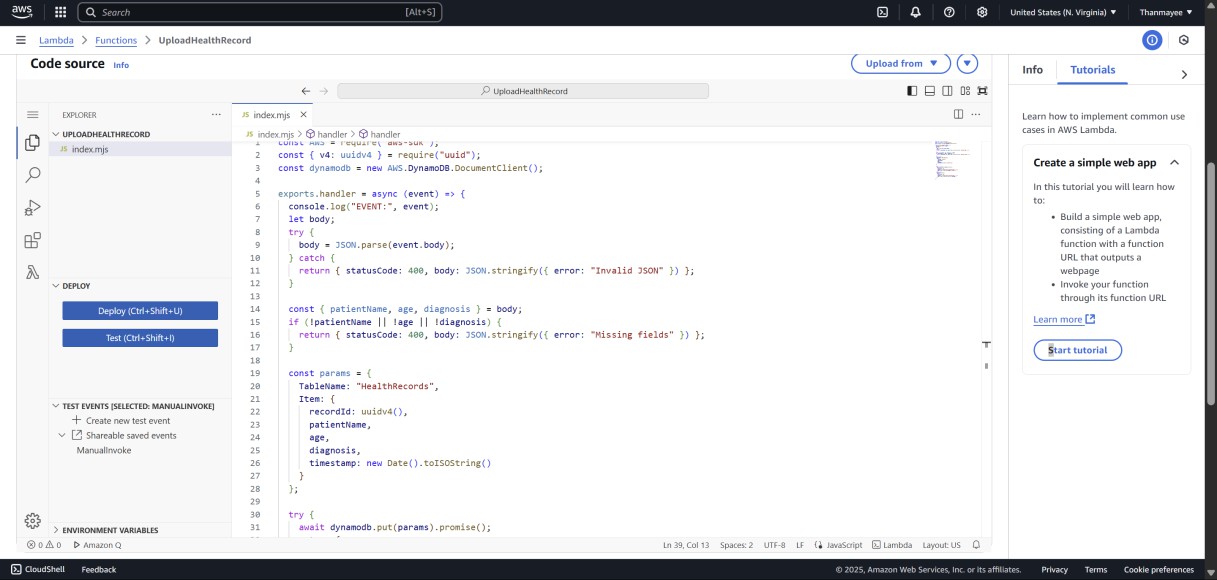
**Fig 4.2** S3 bucket(h-record) configured



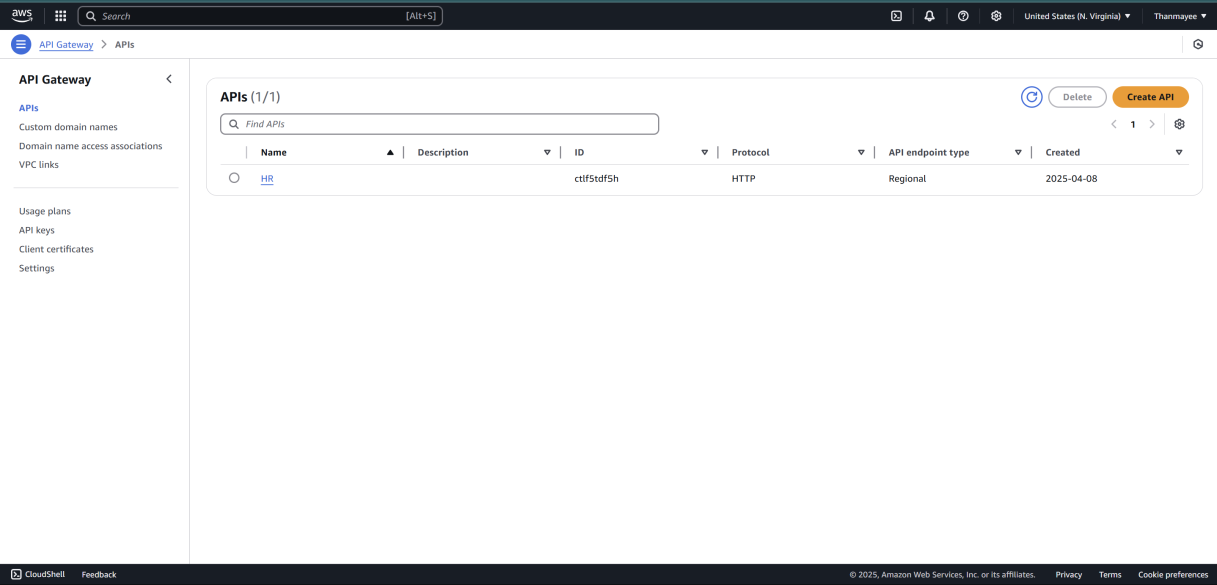
**Fig 4.3** Enable the Static Website hosting which is disabled



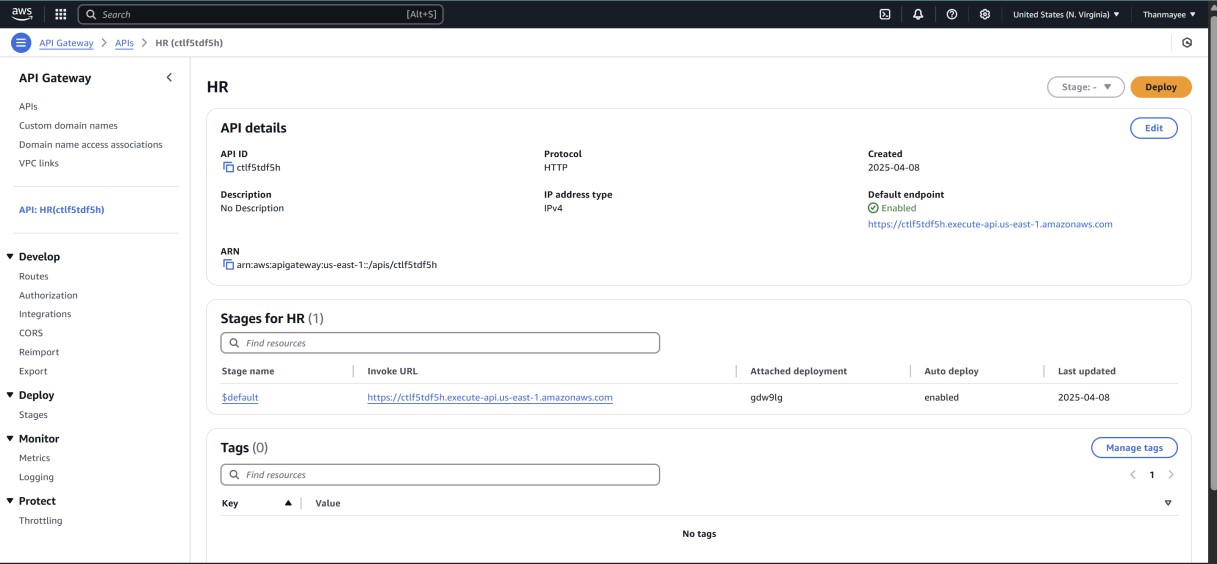
**Fig 4.4** Create a Lambda function



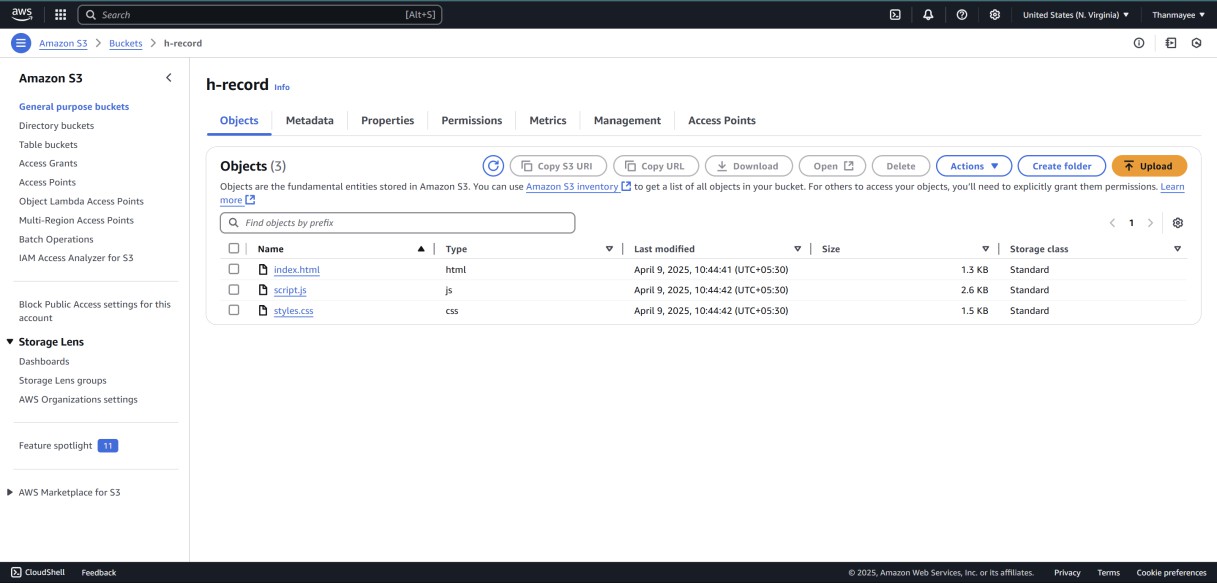
**Fig 4.5** Code Source in Lambda



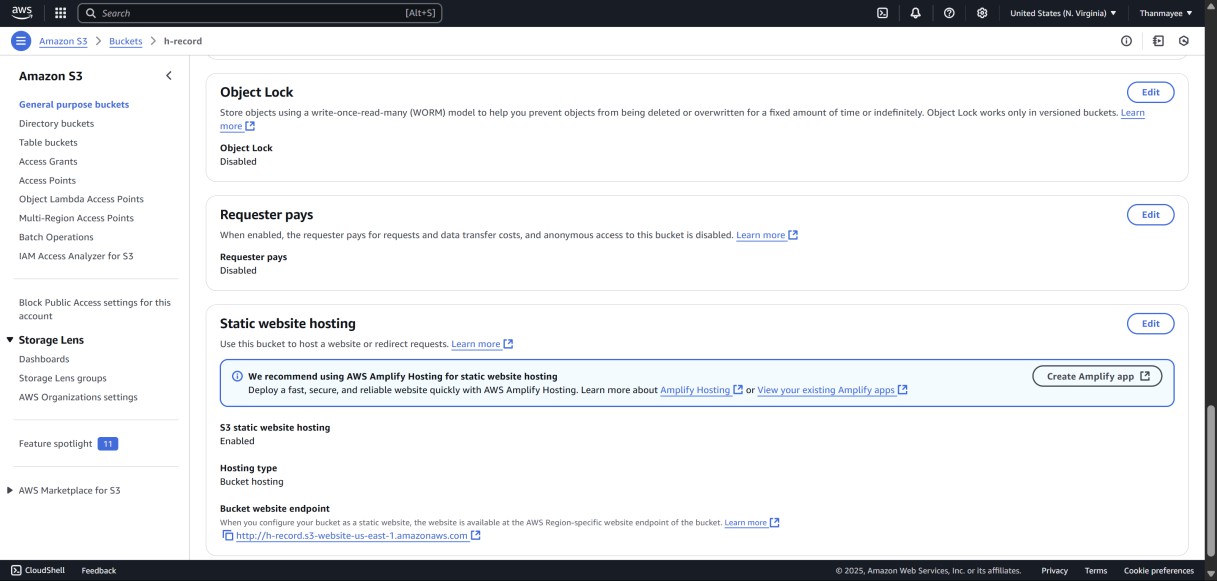
**Fig 4.6** Create an API Gateway (HR)



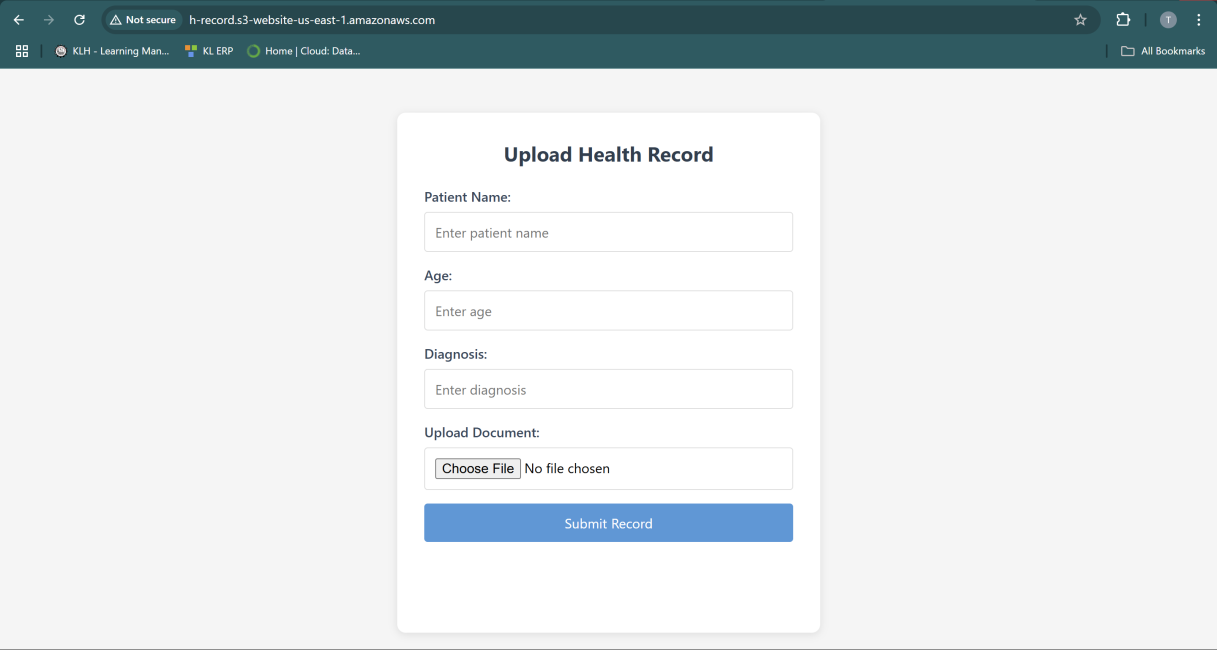
**Fig 4.7** Created API (HR)



**Fig 4.8** In S3 Bucket the Object are Uploaded



**Fig 4.9** In the S3 select the bucket and go for properties, will find a URL for static website



**Fig 4.10** The final Output

## Learning Outcomes

* Gained practical knowledge of AWS serverless services including Lambda, API Gateway, DynamoDB, and S3.
* Understood how to build FHIR-compliant APIs on cloud infrastructure.
* Learned how to integrate frontend applications with backend cloud services.
* Enhanced understanding of cloud security, monitoring, and IAM policy configurations.

## Conclusion

This project demonstrates how to build a FHIR-compliant Cloud-Based Healthcare Management System using AWS services like DynamoDB, S3, Lambda, and API Gateway. The system enables secure and scalable storage, retrieval, and exchange of healthcare data. A user-friendly S3-hosted frontend allows uploading patient records, which are processed via Lambda and stored in DynamoDB. API Gateway provides secure endpoints for seamless integration. Future enhancements may include analytics, HealthLake integration, and advanced access controls.

## References

1. HL7. (n.d.). *FHIR Overview*. Retrieved from https://[www.hl7.org/fhir/over-](http://www.hl7.org/fhir/over-) view.html
2. Amazon Web Services. (n.d.). *Serverless Computing*. Retrieved from <https://aws.amazon.com/serverless/>
3. Amazon Web Services. (n.d.). *AWS Security Best Practices*. Retrieved from <https://docs.aws.amazon.com/general/latest/gr/aws-security-best-practices.html>
4. Amazon Web Services. (n.d.). *AWS Free Tier*. Retrieved from [https://aws.ama-](https://aws.amazon.com/free/) [zon.com/free/](https://aws.amazon.com/free/)