Final Project Narrative: AWS Serverless URL Shortener

Executive Summary

This project implements a fully serverless URL shortening service on AWS, demonstrating cloud-native architecture principles and operational best practices. The solution leverages managed AWS services for scalability, cost-effectiveness, and production readiness.

Technical Implementation

Architecture Philosophy

The application follows a serverless-first approach using managed AWS services to minimize operational overhead and maximize scalability.

Key Design Decisions

- **Serverless Over EC2:** Lambda functions provide automatic scaling, pay-per-use pricing, and reduced operational overhead.
- **DynamoDB:** Chosen for low-latency lookups and automatic scaling. TTL enables automatic URL expiration.
- **Static Frontend on S3:** Cost-effective, highly available hosting with optional CloudFront CDN.
- **Security:** Least-privilege IAM roles and environment variables for configuration; HTTPS enforced via API Gateway.

Challenges & Solutions

- **CORS:** Enabled CORS in API Gateway and Lambda headers to allow frontend communication.
- **Database Schema:** Single-table design with short_code as primary key and GSI for analytics.
- **Error Handling:** Standardized error responses and comprehensive logging for easier debugging.

Cloud Learning Outcomes

- **Modules 1–5, 6, 8–11:**
  Selected managed services, implemented IAM best practices, configured API Gateway, S3 hosting, Lambda compute, DynamoDB storage, and CloudWatch monitoring.
- **Monitoring:** SLOs defined with p95 latency < 300ms, error rate <1%, availability >99%.

Team Contributions

- **Konduri Vamsi Reddy:** AWS infrastructure, IAM policies, deployment automation

- **Vineeth Gongati:** Lambda functions, API Gateway configuration, backend testing
- **Sudarshan Challa:** Frontend development, end-to-end testing, documentation

AI Usage & Human Review

- **AI Assistance:** Generated Lambda templates, IAM policy snippets, troubleshooting commands, and CloudFormation configurations.
- **Human Review:** Security hardening, input validation, monitoring setup, production readiness, operational documentation.

Future Enhancements

- CloudFront distribution for global low-latency access
- Custom short domains (branded URLs)
- Analytics dashboard for click tracking
- Rate limiting through API Gateway usage plans
- Multi-region deployment for high availability
- Automated URL expiration with DynamoDB TTL

Conclusion

The project demonstrates a production-ready serverless URL shortener on AWS, highlighting best practices in security, scalability, cost optimization, and operational excellence. This solution can serve as a foundation for commercial applications.