

# AWS State, Local, and Education Learning Days

Sacramento, CA

3:15pm – 4:15pm

**300**  
level

Designing modern applications in AWS

Unlock Serverless Potential: Reduce Costs, Boost Scalability, and Enhance Security with Cloud-Native Architectures.



# Designing modern applications in AWS

## **Ash Naik (she/her)**

Solutions Architect  
Amazon Web Services  
ashjnaik@amazon.com

## **Lakshmi Dasari (she/her)**

Sr Solutions Architect  
Amazon Web Services  
lddasar@amazon.com

# Voice of the Customer

Scan this QR Code with your phone



Or join at: <https://menti.com> with code **7595 1154**

# Architecting on AWS is different

## Legacy

Monolith (app + DB tightly coupled)

Vertical scaling (big VMs)

One language/stack

Manual deployments

Fragile updates

Owned infrastructure

## Modern (AWS)

Modular (microservices, event, SOA) architectures

Automated scaling

Polyglot, fit-for-purpose technology

Automated pipelines (CI/CD)

Independent service delivery

Managed services, focus on delivery business value

# Architecting on AWS is different

- It's not just about stringing together services, but about building scalable, elastic, resilient, secure, reliable and cost-efficient solutions using managed cloud-native services
- Leverage cloud-scale, robust infrastructure
- The cost model frees architects up to do more & more cost-efficiently; promotes innovation
- Emphasis on composable architectures of distributed, modular & reusable components; generally service-oriented
- Servers/hosts are no longer the atomic unit of architecture
- Infrastructure-as-code: infrastructure as cattle, not pets
- Hyperautomation as a strategy
- Expanded integration and orchestration pathways (asynchronous, event-driven, "everything-as-an-API" decoupling)

# Principles of Modern Cloud System Architecture

## Systems

Build flexibly for the future with loosely coupled services and component-based architecture



Require auto-scaling and load balancing



Use purpose-built services



Govern architecture across the enterprise



Ensure performance and skill alignment



Offer seamless storage functionality



Decouple infrastructure & experience



Work backwards from business needs



Leverage automation and containers

# Principles of Modern Cloud System Architecture

## Experience

Deliver holistically with user-centered design, accessibility, and reusable components



Require scalable public and worker interfaces



Segment and personalize



Develop a unified design approach



Ensure performance and skill alignment



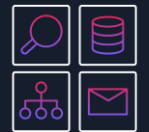
Use seamless data to make decisions



Decouple data and channels



Work backwards from user needs



Leverage reusable components

# What is the AWS Well-Architected Framework?



Pillars & Lenses



Design principles



Questions



Best Practices

# Pillars of the AWS Well-Architected Framework

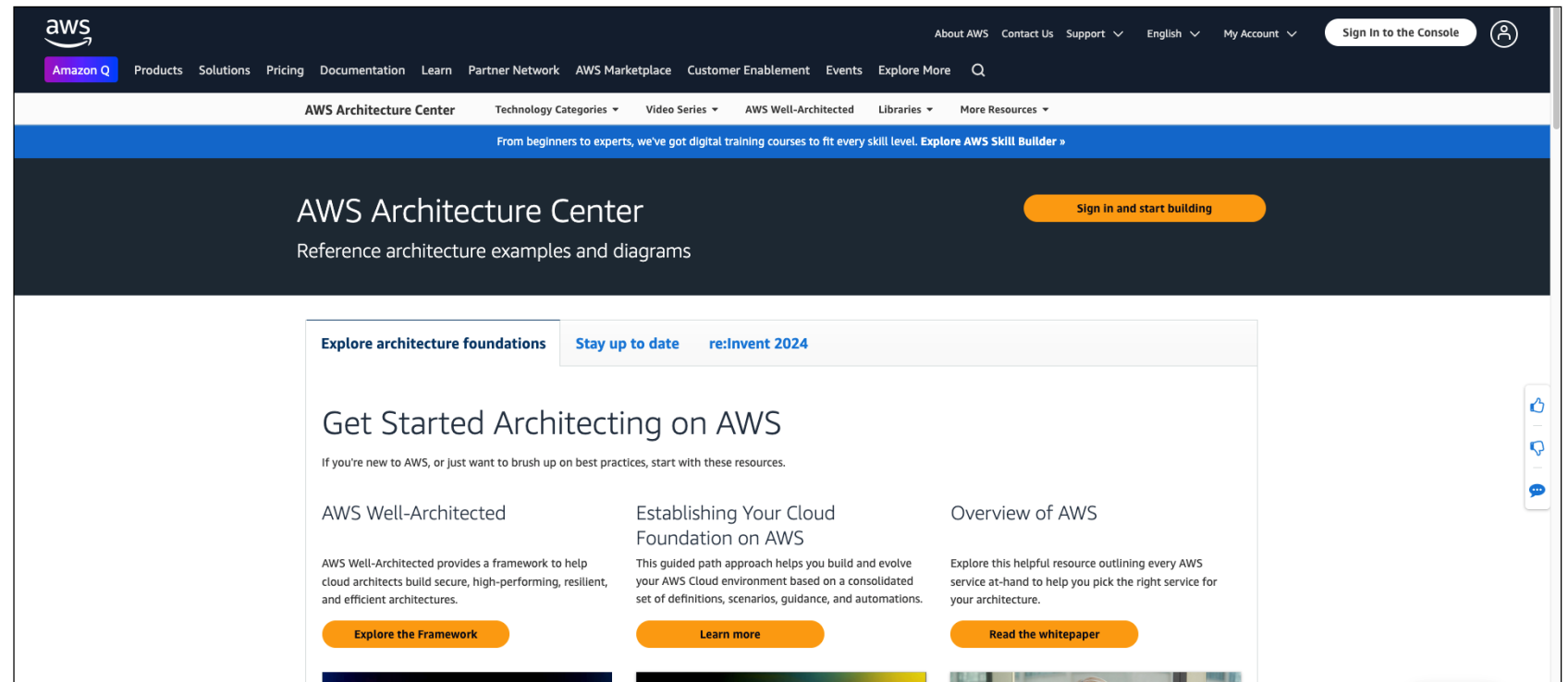


# AWS Architecture Center

- Library of content including
  - Patterns
  - Reference Architectures
  - Guidance
  - Solutions, and more
- Links and other resources for architecting on AWS
- Video Series like 'This is My Architecture' and 'How to Build This'
- Architecture Best Practices



<https://aws.amazon.com/architecture>



The screenshot shows the AWS Architecture Center homepage. At the top, there's a navigation bar with the AWS logo, 'Amazon Q', and various menu items like 'Products', 'Solutions', 'Pricing', 'Documentation', 'Learn', 'Partner Network', 'AWS Marketplace', 'Customer Enablement', 'Events', and 'Explore More'. A search icon is also present. On the right side of the navigation bar, there are links for 'About AWS', 'Contact Us', 'Support', 'English', 'My Account', and a 'Sign In to the Console' button with a user icon.

Below the navigation bar, the main heading reads 'AWS Architecture Center' with a subtitle 'Reference architecture examples and diagrams'. A prominent orange button says 'Sign in and start building'. A blue banner below the heading states 'From beginners to experts, we've got digital training courses to fit every skill level. Explore AWS Skill Builder »'.

The main content area features a section titled 'Explore architecture foundations' with sub-sections for 'Stay up to date' and 're:Invent 2024'. The primary focus is on 'Get Started Architecting on AWS', which includes a sub-heading 'If you're new to AWS, or just want to brush up on best practices, start with these resources.' Below this, there are three featured resources:

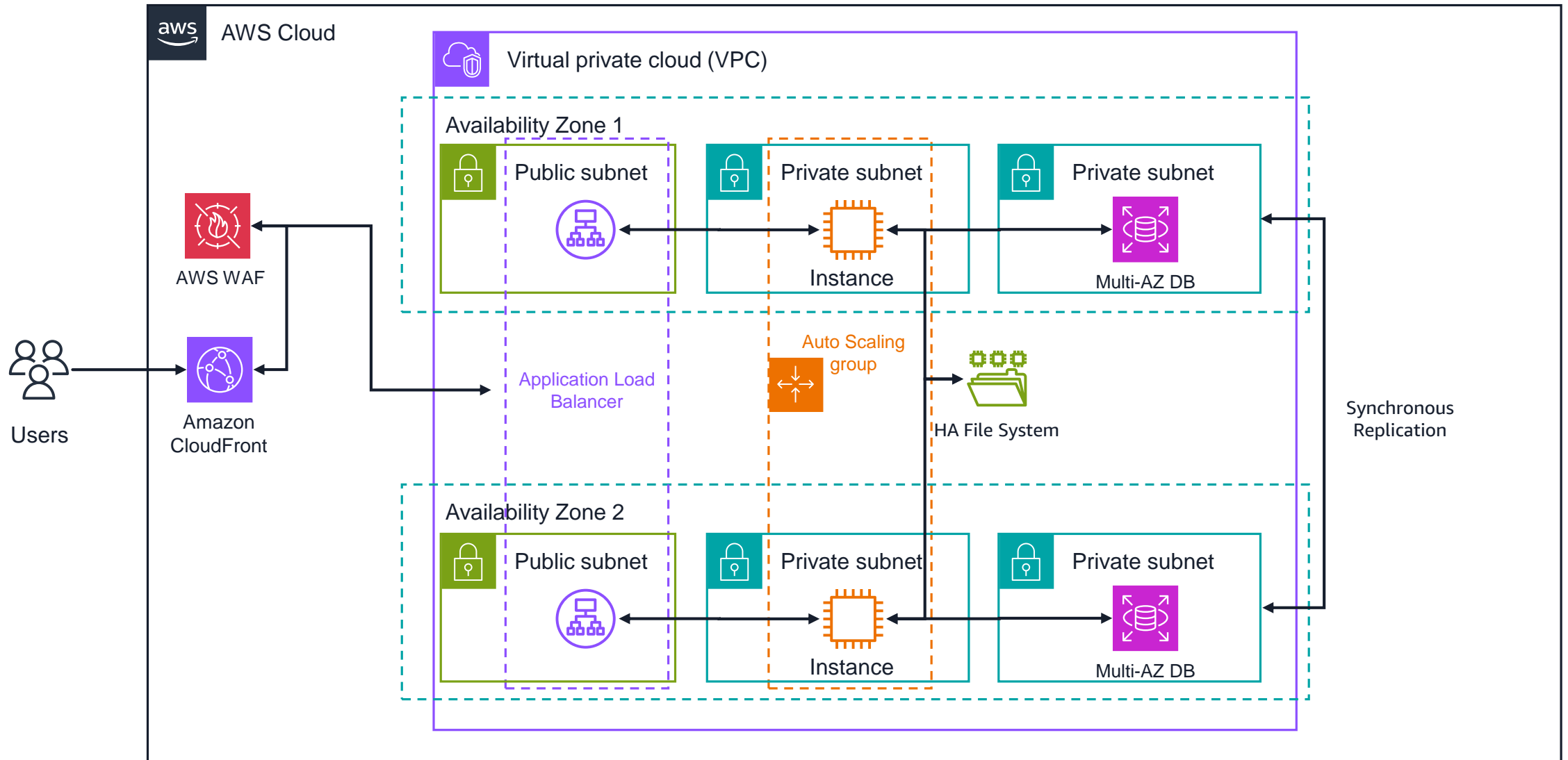
- AWS Well-Architected**: 'AWS Well-Architected provides a framework to help cloud architects build secure, high-performing, resilient, and efficient architectures.' with a button 'Explore the Framework'.
- Establishing Your Cloud Foundation on AWS**: 'This guided path approach helps you build and evolve your AWS Cloud environment based on a consolidated set of definitions, scenarios, guidance, and automations.' with a button 'Learn more'.
- Overview of AWS**: 'Explore this helpful resource outlining every AWS service at-hand to help you pick the right service for your architecture.' with a button 'Read the whitepaper'.

On the right side of the page, there is a vertical sidebar with social media sharing icons (Facebook, Twitter, LinkedIn, Email, Print).

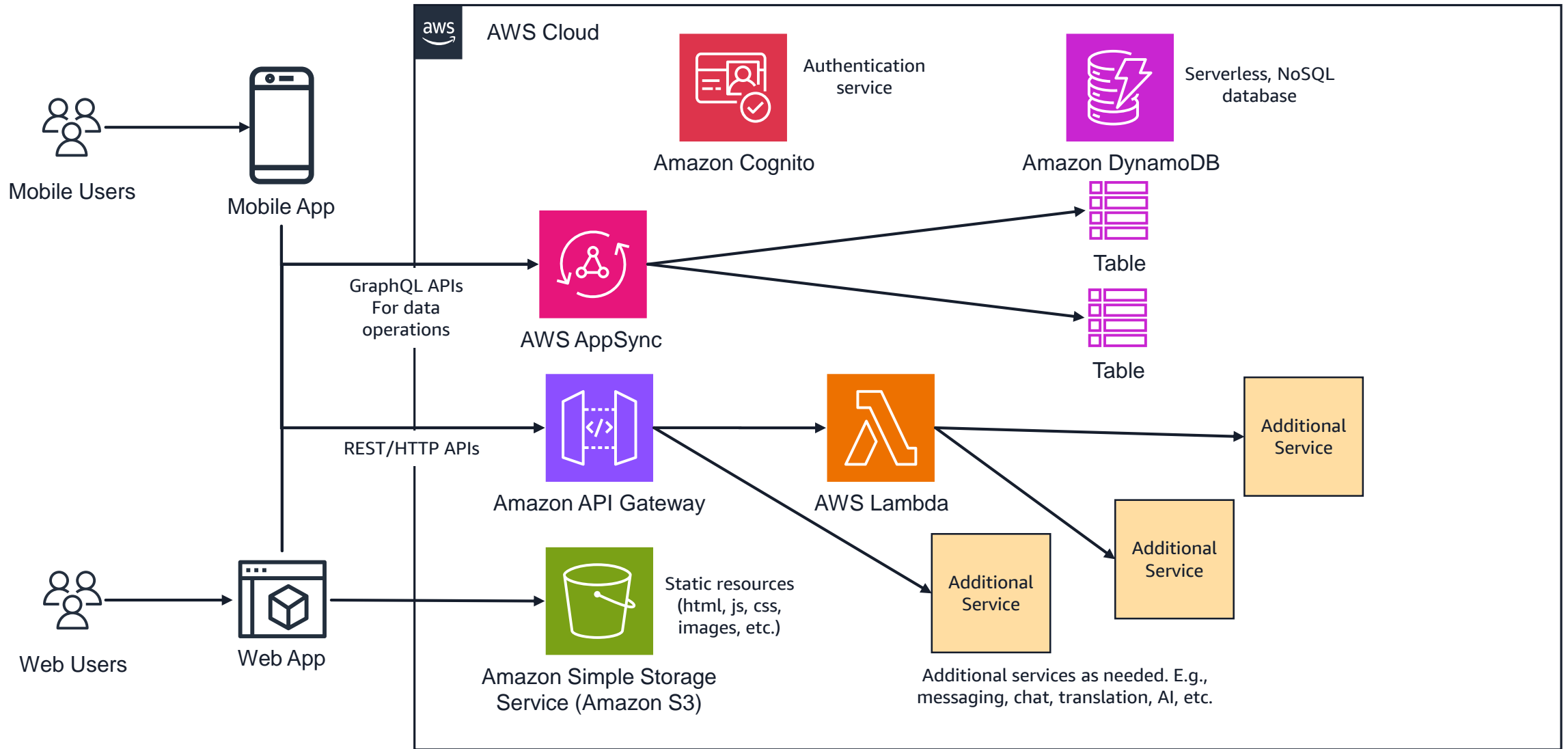


# Let's get out the chalk

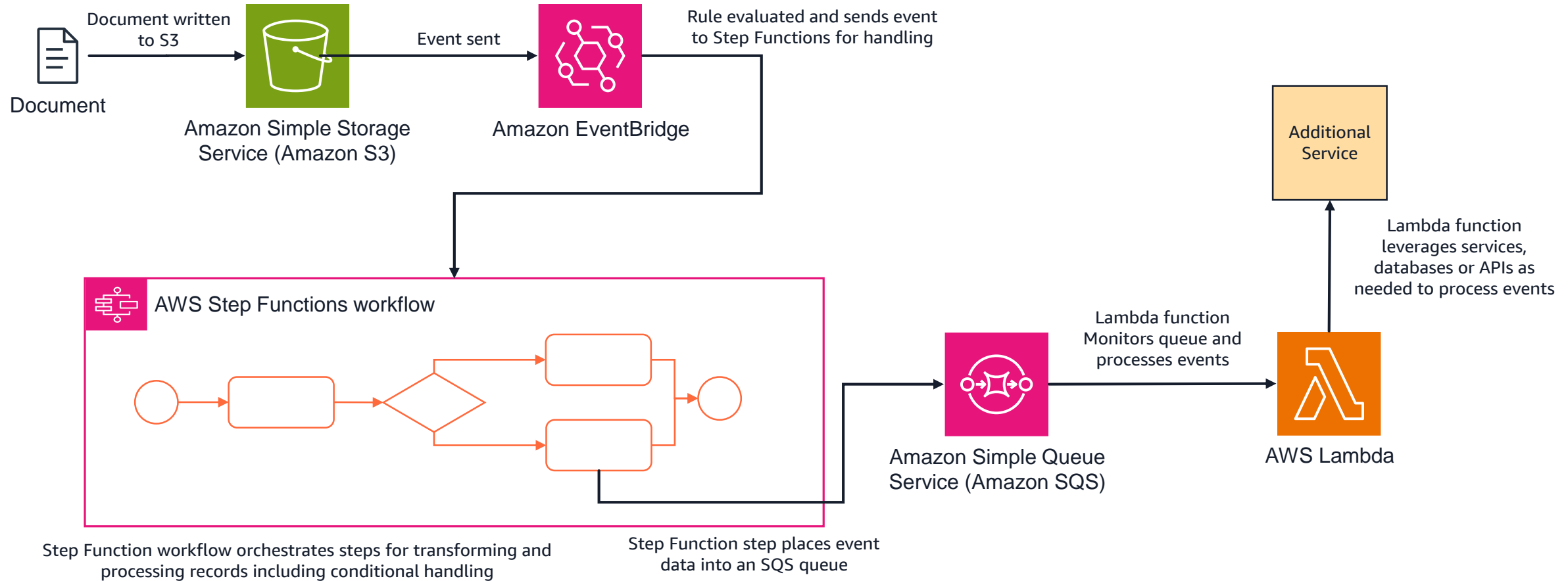
# Highly Available 3-Tier Application



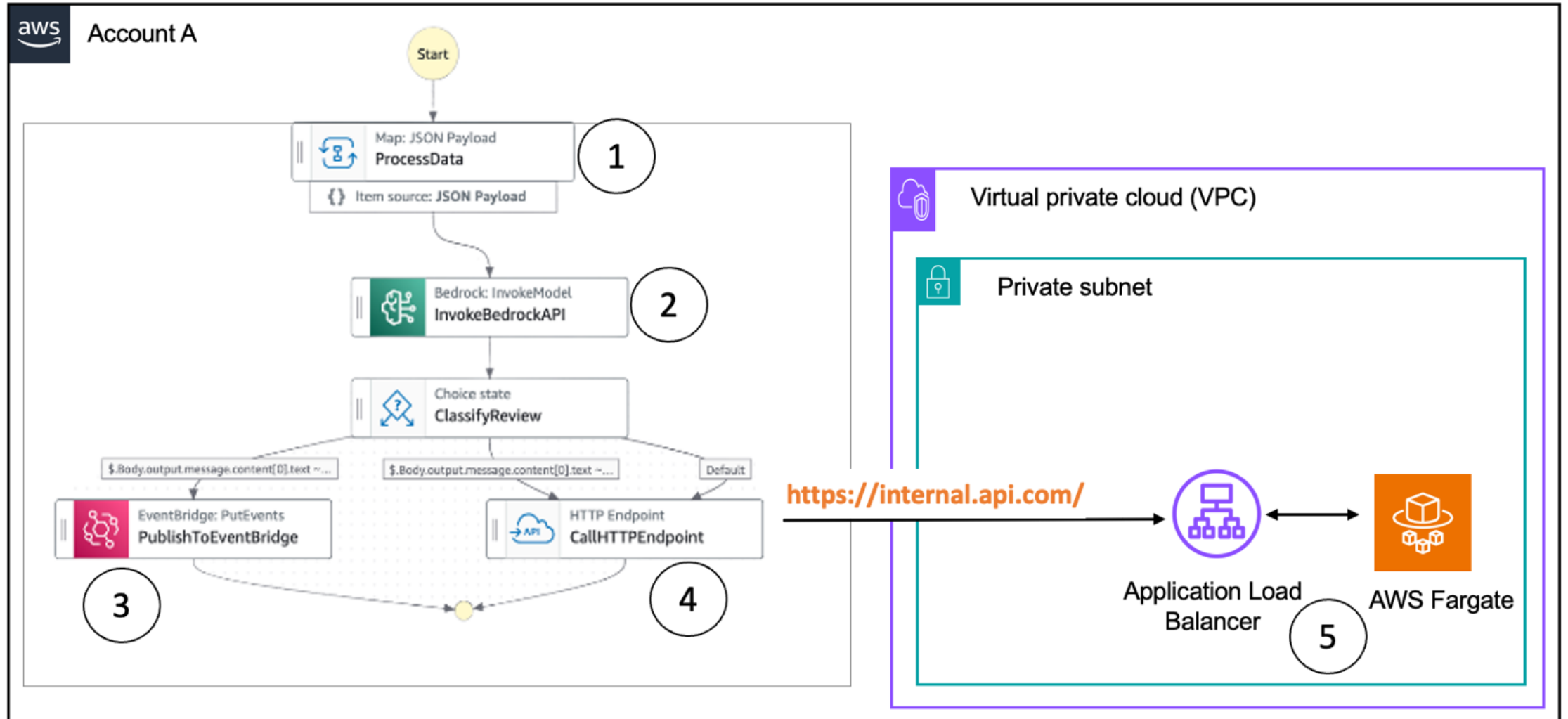
# Serverless Web/Mobile Application with APIs



# Event-driven processing

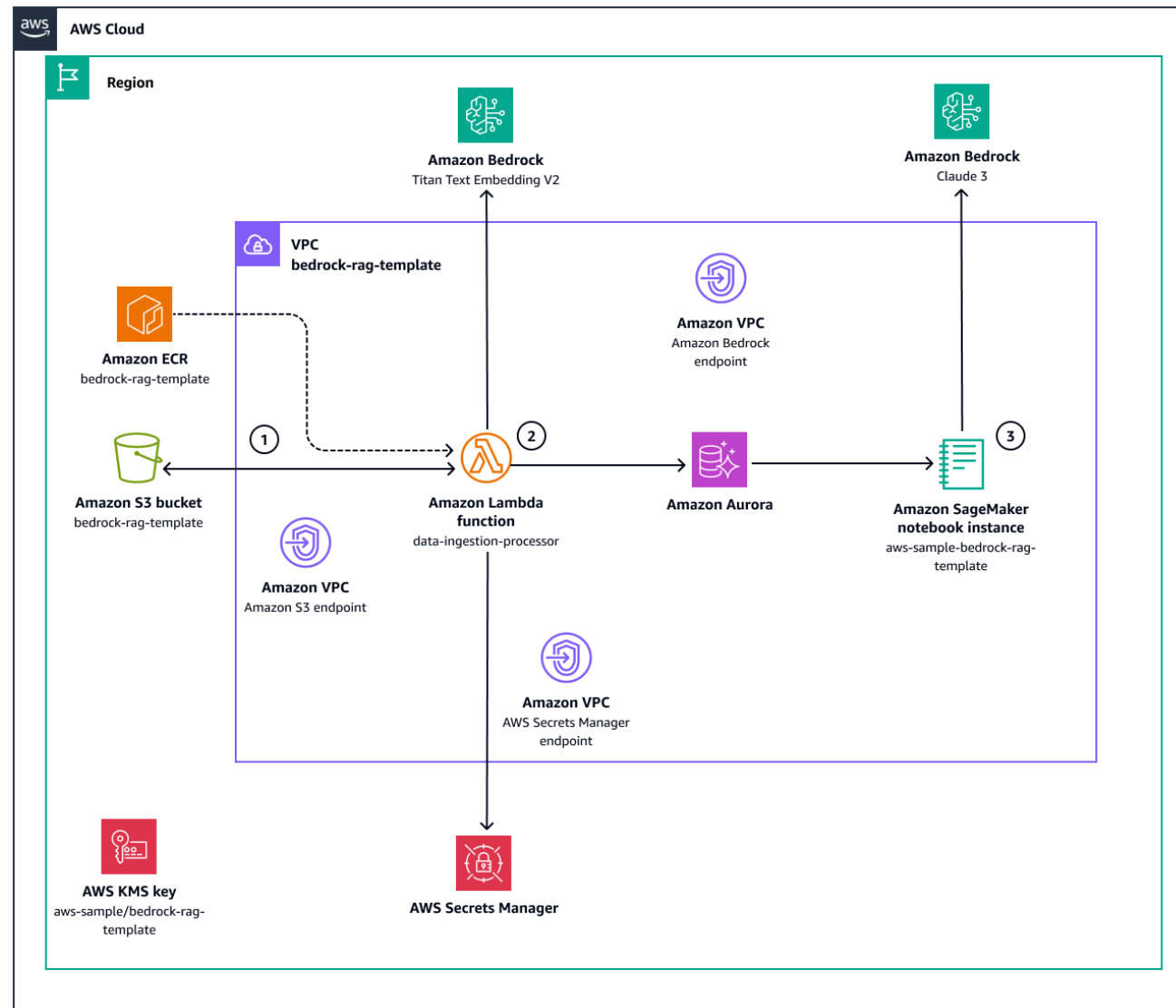


# Orchestrated Asynchronous Processing



Source: <https://aws.amazon.com/blogs/compute/simplifying-private-api-integrations-with-amazon-eventbridge-and-aws-step-functions-2/>

# Generative AI RAG Architecture



Source: <https://docs.aws.amazon.com/prescriptive-guidance/latest/patterns/deploy-rag-use-case-on-aws.html>



# Thank you!

**Ash Naik (she/her)**

Solutions Architect  
Amazon Web Services  
ashjnaik@amazon.com

**Lakshmi Dasari (she/her)**

Sr Solutions Architect  
Amazon Web Services  
lddasar@amazon.com

Please complete the survey  
for this session



**Track: Cloud Fundamentals**

Session: Designing modern applications in AWS