This roadmap is broken into **6 key phases** (aligned with Agile Program Increments) and each phase has **multiple sprints (2 weeks)**. This structure allows incremental delivery, continuous feedback, and ongoing validation of the architectural integrity and business value.

**Phase 1: Environment & Tooling Foundation (Sprint 1–2)**

**Objectives:**

* Provision Amazon Workspaces (Windows 11) for each dev
* Install baseline tooling (Java 17, IntelliJ, Docker, Git, AWS CLI, etc.)
* Set up GitHub org/repo structure
* Configure GitHub Actions skeleton (CI only)
* Initialize AWS CDK repo (TypeScript)

**Outcomes:**

* Dev teams can work in isolation
* CDK supports IaaC for future phases
* CI pipeline runs tests and checks on commits

**Phase 2: Core API & Local Dev Architecture (Sprint 3–4)**

**Objectives:**

* Define domain model (Movie, Rating)
* Build local Spring Boot project structure
* Add DTOs, Controllers, Services
* Integrate SQLite with Spring Data JPA
* Implement List All Movies + Movie Details endpoints
* Unit testing (JUnit5 + Mockito)

**Outcomes:**

* Functional local API
* Fully testable, layered architecture
* Swagger docs locally visible

**Phase 3: Cloud Integration & CI/CD (Sprint 5–6)**

**Objectives:**

* Containerize Spring Boot App (Docker)
* Set up Amazon ECR + GitHub Actions integration
* Provision ECS Fargate service via CDK
* Define and deploy VPC, subnets, ALB, NAT, etc.
* Migrate SQLite to Amazon RDS (PostgreSQL)

**Outcomes:**

* Deployed dev environment (isolated VPC)
* CI/CD builds + deploys container to ECS
* Shared RDS database with secure access

**Phase 4: Authentication, Security & Monitoring (Sprint 7–8)**

**Objectives:**

* Integrate Amazon Cognito (AuthN/AuthZ)
* Secure API Gateway with Cognito authorizer
* Enable HTTPS via ALB
* Add Micrometer → CloudWatch metrics
* Configure SNS for alarms & email alerts

**Outcomes:**

* Secure, monitored API
* Audit-ready traceability
* Alerting pipeline in place

**Phase 5: Feature Expansion & API Maturity (Sprint 9–10)**

**Objectives:**

* Implement Movies By Year + Movies By Genre endpoints
* Add pagination, sorting, filtering
* Refactor into versioned API paths (e.g., /api/v1/movies)
* Add global exception handling and RFC 7807 error schema
* Add Postman collection for all endpoints

**Outcomes:**

* Full MVP delivered
* Clean REST contract
* Ready for UAT & external testing

**Phase 6: Staging, Prod & Rollout Strategy (Sprint 11–12)**

**Objectives:**

* Define blue/green deployment pattern for ECS
* Provision staging & prod environments via CDK
* Implement usage throttling on API Gateway
* Finalize GitHub branching & tagging strategy
* Conduct load testing (Artillery or k6)

**Outcomes:**

* Resilient deployment strategy
* Dev → Staging → Prod flow enabled
* Rollback/rollback plan implemented