# Implementation Approach for Image Migration and API Development

## 1. High-Level Implementation Plan

### Objective:

Migrate existing images from `i-scanned-image` table in Oracle to AWS S3 and expose REST APIs to perform CRUD operations and image manipulation.

### Implementation Plan (Tabular Format):

|  |  |  |
| --- | --- | --- |
| Phase | Week | Tasks |
| Requirement Analysis & Design | 1-2 | Identify source database schema, Define AWS S3 structure, Design REST API endpoints, Create high-level architectural diagram |
| Development & Implementation | 3-6 | Develop Spring Boot microservice, Upload images to AWS S3, Store metadata, Implement CRUD operations, Integrate security mechanisms |
| Testing & Validation | 7-8 | Unit testing, Performance testing, Integration testing |
| Deployment & Monitoring | 9-10 | Deploy microservice to staging, Migrate test dataset, Go-live with phased migration, Monitor logs and AWS CloudWatch |

## 2. IT Release Strategy

### Release Milestones:

|  |  |
| --- | --- |
| Milestone | Week |
| Dev Environment Setup | 1-2 |
| Staging Deployment | 7-8 |
| Production Deployment | 9-10 |

### Environment Setup:

|  |  |
| --- | --- |
| Environment | Components |
| Development | Local Spring Boot service with Oracle DB |
| Staging | AWS S3 + Oracle + API Gateway |
| Production | Deployed in AWS with monitoring and logging |

## 3. Release Plan

### Phases of Migration:

|  |  |  |
| --- | --- | --- |
| Phase | Week | Tasks |
| Initial Testing | 7 | Move sample images to AWS S3 |
| Incremental Migration | 8-9 | Migrate batches of images |
| Final Cutover | 10 | Complete migration and redirect API traffic |

### API Exposure Strategy:

|  |  |
| --- | --- |
| Endpoint | Description |
| POST /images | Upload image |
| GET /images/{id} | Retrieve image |
| PUT /images/{id} | Update image |
| DELETE /images/{id} | Delete image |
| POST /images/manipulate | Image processing |

Security Measures: JWT Authentication, IAM roles for S3.

Rate Limiting: API Gateway throttling.