One ACE Deployment Guide: Integrating DigitalSuite ACE and Core System ACE

Introduction

## Purpose:

This document provides step-by-step instructions to deploy the One ACE solution by integrating DigitalSuite ACE and Core System ACE (Ex-CoreSuite). It outlines the procedures for packaging, Docker image creation, and environment promotion (Development → QA → UAT → Production).

## Target Audience:

* DevOps Engineers,
* System Administrators,
* Release/Deployment Teams.

## Scope:

* Covers packaging,
* Docker image creation,
* Validation,
* Environment promotion (Development → QA → UAT → Production).

# Prerequisites

* Access to CS and Digital Git repositories.
* Docker installed and configured.
* Valid credentials for CI/CD pipeline.
* Environment-specific configurations applied (QA, UAT, Production).

Pipeline Integration

#### **3.1 Pull Code from Git Repositories**

* **Step 1:** Clone the Core System Git repository:

bash

CopyEdit

git clone https://git.example.com/core-system.git

* **Step 2:** Clone the DigitalSuite Git repository:

bash

CopyEdit

git clone https://git.example.com/digital-ace.git

#### ✅ **3.2 Merge Code from Both Repositories**

* **Step 3:** Merge CS and DigitalSuite code:

bash

CopyEdit

git merge origin/digital-branch

* **Step 4:** Resolve any naming conflicts as per the naming convention.

Package and Docker Image Creation

#### **4.1 Create a Package from Merged Code**

* **Step 1:** Zip the merged code for deployment:

bash

CopyEdit

zip -r ace\_package.zip ./merged\_code/

#### ✅ **4.2 Build Docker Image**

* **Step 2:** Build the Docker image:

bash

CopyEdit

docker build -t one-ace-image:latest .

#### ✅ **4.3 Push Docker Image to Registry**

* **Step 3:** Push the image to the Docker registry:

bash

CopyEdit

docker push registry.example.com/one-ace-image:latest

Verification and Testing (Quality Checks)

#### **5.1 Run Integrity and Functional Tests**

* **Step 1:** Verify package integrity:

bash

CopyEdit

pytest tests/

* **Step 2:** Validate Docker image functionality:

bash

CopyEdit

docker run -d -p 8080:8080 one-ace-image:latest

Deployment Stages

* **Step 1:** Deploy to Development Environment.
* **Step 2:** Deploy to QA Environment for functional and non-functional testing.
* **Step 3:** Deploy to UAT for user acceptance testing.
* **Step 4:** Final deployment to Production.

Rollback Strategy

**In case of failures, redeploy the last successful Docker image:**

* **Step 1:** Identify and revert to the last successful version.

bash

CopyEdit

docker tag registry.example.com/one-ace-image:previous

* **Step 2:** Redeploy the previous version.

bash

CopyEdit

docker run -d -p 8080:8080 registry.example.com/one-ace-image:previous