**Best Practices for Digital and CoreSuite Integration Flow**

Table of Contents

[Introduction 1](#_Toc169521271)

[Best Practices for Integration Flow 2](#_Toc169521272)

[**1.** Define Clear Ownership 2](#_Toc169521273)

[1.1 Digital Integrations (Non-CoreSuite) 2](#_Toc169521274)

[1.2 Digital to CoreSuite 2](#_Toc169521275)

[**2.** Segregation of Responsibilities 2](#_Toc169521276)

[2.1 Digital Team 2](#_Toc169521277)

[2.2 CoreSuite Team 2](#_Toc169521278)

[**3.** Collaborative Development Process 2](#_Toc169521279)

[User Stories: 2](#_Toc169521280)

[Joint Review Sessions: 2](#_Toc169521281)

[**4.** Consistent Data Transformation 3](#_Toc169521282)

[Transformation Logic: 3](#_Toc169521283)

[Standardization: 3](#_Toc169521284)

[**5.** Robust API Documentation 3](#_Toc169521285)

[API Specifications 3](#_Toc169521286)

[Version Control 3](#_Toc169521287)

[**6.** Security and Compliance 3](#_Toc169521288)

[Authentication: 3](#_Toc169521289)

# Introduction

Effective integration between Digital and CoreSuite (CS) systems provides smooth operations, robust data exchange, and efficient project implementation and improve performance.

Here are the best practices and guidelines for successful Digital and CoreSuite (CS) integration flow:

# Best Practices for Integration Flow

**1.** Define Clear Ownership

## 1.1 Digital Integrations (Non-CoreSuite)

* + Managed by the Digital team.
  + These are digital APIs integrated with CMS, KeyCloak, Mailjet, and other systems.

## 1.2 Digital to CoreSuite

* These APIs call Coresuite ACE APIs, such as create-quote and create-policy.
* Ownership and responsibility for these APIs are typically shared between the Digital Team and the CS Team.

Important Notes

1. **Digital and** CoreSuite **Business Logic:**
   * The Digital Team will **not** write or duplicate any CS-specific business logic.
   * If there are modifications needed or a new Coresuite ACE API needs to be created, these requests should be communicated to the Coresuite ACE team through user stories.
2. **CS Team and API Creation/Enhancement:**
   * The CS team is responsible for creating or enhancing APIs to implement specific requirements.
3. **Data Transformation:**
   * The Digital may create APIs or flows for data transformation as needed to provide support.

Here are two important points

1. Digital team will not write/duplicate any CS specific business logic.
2. In case if there is any modification required or need to create a new core suite ace API, then such request should be passed to core suite ACE team through user stories.
3. CS team will create or enhance APIs to implement such requirement(s)

ii. Digital team may create api/flow for any data transformation as required to support.

**2.** Segregation of Responsibilities

## 2.1 Digital Team

* + Focus on user experience and front-end integration.
  + Manage non-CoreSuite API interactions.

## 2.2 CoreSuite Team

* + Handle core business logic and backend operations.
  + Create and enhance APIs a smentione dabove based on user stories from the Digital team.

For more information, refer [Journey and Component APIs](https://sapiens365-my.sharepoint.com/:x:/r/personal/vishish_jangde_sapiens_com/_layouts/15/Doc.aspx?sourcedoc=%7BC8781F58-0A70-4458-9743-BD944B72C2EB%7D&file=Journey%20&%20Component%20APIs.xlsx=&wdLOR=c93179DA5-2B91-4EA3-BDF5-3CDC8ADB4B3C&fromShare=true&action=default&mobileredirect=true)

**3.** Collaborative Development Process

## User Stories:

* + Digital team documents requirements as user stories.
  + These stories are passed to the CS team for implementation.

## Joint Review Sessions:

* + Regular meetings to review user stories and API requirements.
  + Ensure alignment and address any integration challenges.

**5.** Robust API Documentation

## API Specifications

* + Clearly document API endpoints in open API 3.0 specification and Swager.

For more information, refer [OpenAPI Specification](https://swagger.io/specification/)

Adhering to these best practices and guidelines, the integration between Digital and CoreSuite can be managed efficiently, ensuring a seamless flow of data and functionality across systems.