

Browse the Book

This sample chapter discusses how various SAP Cloud Platform services can help you and your organization create and define the best user experience for your end users. Specifically, it covers how you can provide a consistent look and feel across your enterprise using the SAP Cloud Platform Launchpad and SAP Work Zone.



“User Experience and Mobile Consumption”



Contents



Index



The Authors

Paresh Mishra, Vipin Varappurath

Customizing SAP S/4HANA with SAP Cloud Platform: Designing a Future-Ready Enterprise Architecture

383 Pages, 2021, \$79.95

ISBN 978-1-4932-2008-3



www.sap-press.com/5157

Chapter 7

User Experience and Mobile Consumption

This chapter further builds on the UX principles mentioned in Chapter 3. The best end user experience ensures business efficiency for your business and adoption of your enterprise system by end users. Thus, it's essential to consider the best UX strategy: one that's the least disruptive to end users, is intuitive, provides a consumer-grade experience, and can be accessed from anywhere and on any device for a modern and agile landscape. In this chapter, we'll cover various SAP Cloud Platform services that help create the best end user experience.

In this chapter, we'll look at an important aspect in your future-ready transformation journey: defining the experience for your end users using and interacting with your system. End users could be employees in your organization, your vendors or suppliers, or storefront users.

The user experience you provide determines how engaged your end users are and defines their level of satisfaction with your systems, which could be vital to your project's success. The challenges in this area are very demanding, and transformation may be ongoing because the user experience requirements and domain remains disruptive. For example, as mentioned in Chapter 1, the global COVID-19 pandemic in 2020 forced many organizations to extend remote-working possibilities and still help their employees remain productive in the best possible way. This demands that data and applications are available for employees to support their daily activities in a seamless and mobile-friendly way, which in turn demands that IT teams provide such features quickly and agilely while still meeting enterprise security requirements and needs.

In this chapter, we'll discuss how various SAP Cloud Platform services can help you and your organization create and define the best user experience for your end users in an agile and nimble way and still be future ready to adapt to changes quickly. Specifically, we'll cover how you can provide a consistent look and feel across your enterprise using the SAP Cloud Platform Launchpad, implementing adaptive content, and use SAP Cloud Platform mobile services to design and create a mobile user experience for end users in a quick, agile, and flexible way.

7.1 SAP Cloud Platform Launchpad

Let’s consider the following use cases and scenarios to plan for in your organization:

- Your organization has multiple applications in its landscape, which includes both cloud and on-premise systems.
- Your organization is moving to SAP S/4HANA. You decide to move over by each application area. In other words depending on your project plan and schedule, there could be times when you have both SAP S/4HANA and SAP Business Suite running together.
- Your organization could have multiple installations of SAP S/4HANA and other on-premise systems segregated based on usage in different geographies. To further add to the challenge, different SAP S/4HANA systems might have different upgrade schedules.
- Your organization operates a core SAP S/4HANA system. Your organization has a subsidiary that operates another installation of SAP S/4HANA Cloud. There is data exchange and user interaction between these systems.

All these use cases and scenarios pose challenges for how you can provide a harmonized, seamless, and easy user experience to your end users, spanning different systems in your landscape and abstracting the underlying system transformation complexities from them.

Adding to these landscape and system challenges, frequently changing business requirements and needs in the world mean that IT and business must find ways to make these systems accessible from anywhere over the internet, while at same time ensuring their proper security, including authentication and authorization. All this must happen while making sure that these changes take place quickly and can adapt to the needs of business end users.

The *SAP Fiori Deployment Options and System Landscape Recommendations* guide (<http://s-prs.co/v515749>) outlines the solution to this challenge, using SAP Cloud Platform Launchpad as explained in Figure 7.1.

These requirements and needs to adapt exist not only for employees, but also for suppliers, vendors, and end customers who interact with your organization.

The SAP Cloud Platform Launchpad in the Cloud Foundry environment acts as a central entry point for your different SAP S/4HANA systems and custom applications. At the same time, SAP Cloud Platform services provide features to expose your on-premise systems and landscape in a secure and fast way without having to go through the process of changing ports and firewalls in your network, thus helping to adapt to changing business requirements at a quick pace. At the time of writing this book, the roadmap for this service is evolving fast, and more system and service integrations as out-of-the-box functionalities are planned.

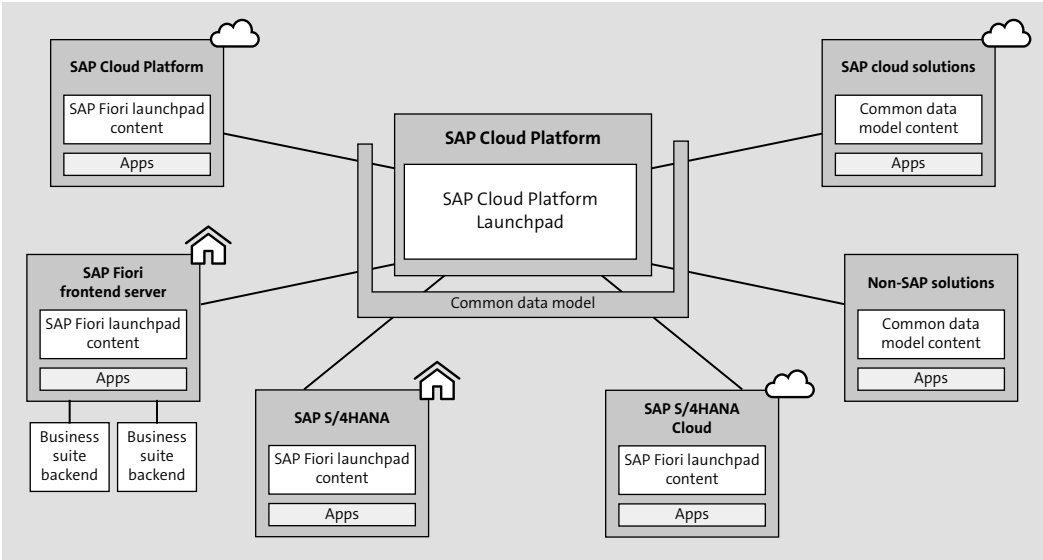


Figure 7.1 SAP Cloud Platform Launchpad Future Direction

Before you begin to set up the SAP Cloud Platform Launchpad, you must have a Cloud Foundry environment subaccount. You must also have set up entitlement for the SAP Cloud Platform Launchpad for this subaccount from the **Entitlements** area.

To begin your setup, follow these steps:

1. Navigate inside your subaccount to **Subscriptions** and choose the **Portal** service tile.
2. Click **Subscribe**.
3. Navigate to **Security • Role Collections**. The SAP Cloud Platform Launchpad has two roles, listed in Table 7.1.

Role Collection	Description
Portal_Admin	Administrator role that provides authorization to perform all administrator roles for the portal
Portal_External_User	Authorizations for external users for portals to enable access to vendors, suppliers, and so on

Table 7.1 SAP Cloud Platform Launchpad Role Collections

4. Now assign these role collections to users. For this, navigate to **Security • Trust Configuration**.
5. Select the identity provider, enter a user name, and choose **Show Assignments** (for the SAP ID service, the user name is your email address).
6. Click **Assign Role Collection** and assign the role collection to a user.

- 7. Navigate back to **Subscriptions** and choose **Portal**, then choose **Go to Application**. This will open the **Site Manager** view as shown in Figure 7.2, from where you can create sites.
- 8. Within a site, you can create apps for SAPUI5, SAP GUI for HTML, and Web Dynpro for ABAP, and integrate with web content to provide a single user experience for your end users.

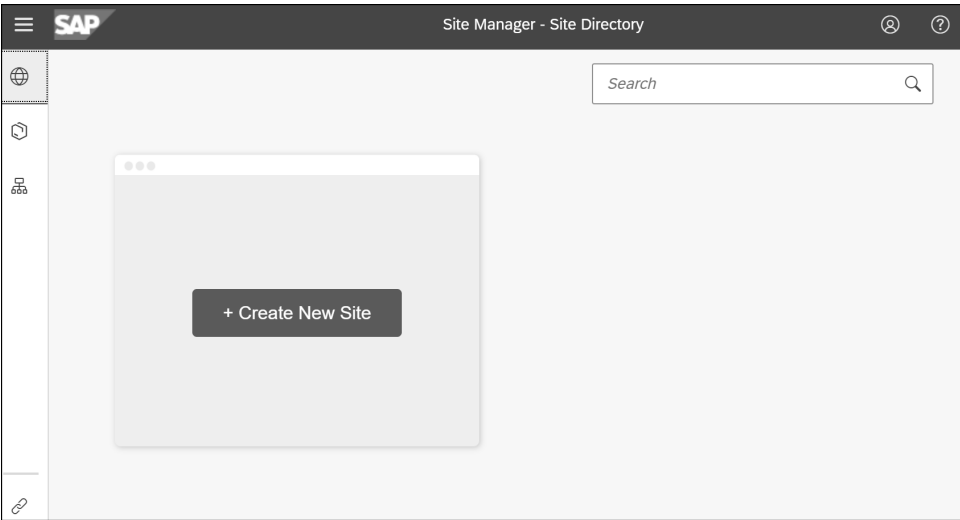


Figure 7.2 SAP Cloud Platform Launchpad Site Manager

7.2 SAP Work Zone

The need for a harmonized user experience is being redefined quite rapidly. Customers and employees of organizations are expecting a consumer-like experience with business systems as well. Users not only expect to have relevant business information and data from the systems presented in one place, but also expect the ability to further engage with that information through contextual recommendations and collaborate with other users through groups and chats around that information. This results in driving more end user productivity, efficiency, and satisfaction.

SAP Work Zone is a set of tools to provide such a user experience to your end users. SAP Work Zone aims to provide a simple, modern, and consistent user experience to your end users with data from enterprise systems.

From a technical standpoint, SAP Work Zone integrates a number of different business services available on SAP Cloud Platform, as follows:

- SAP Cloud Platform Portal
- SAP Cloud Platform Mobile Services
- SAP Cloud Platform Workflow Management

- SAP One Inbox
- SAP Conversational AI

It provides a harmonized user experience and design, all while making sure that your data is secure and follows your enterprise’s security practices, as shown in Figure 7.3.



Figure 7.3 SAP Work Zone Overview

Broadly speaking, there are three different user personas who will use SAP Work Zone:

- **End users**
End users are those to whom data is presented. These could be your employees, who are presented with modern user interface and collaboration capabilities, or your external users, like vendors, partners, and suppliers, who are provided with a modern and efficient way to integrate with your business system. End users can also be presented with mobile apps for a more engaging experience.
End users have the ability to personalize and customize the experience to their unique needs and are presented data based on role-based authorizations.
- **Administrators**
Administrators in an organization are responsible for design and manage SAP Work Zone for their organizations to make the user experience more seamless and harmonized for end users.
An area administrator will have access to only those pages and workspaces for their particular area. A company administrator will have access to individual area administrator profiles. As an administrator, you can see information on user activities within SAP Work Zone.
- **Developers**
Developers are responsible for creating content and deploying the content for use in SAP Work Zone’s pages and workspaces. Developers use SAP Cloud Platform Business Application Studio to develop their contents.

At the time of writing this book, SAP Work Zone for HR is available and is a more employee-centered product. It brings different HR-related apps and services together in one place and provides the ability to collaborate on top of them, mobilize the data, and drive workflows, all in the same place.

UI5 Flexibility for Key Users

Another UX tool worth mentioning is UI5 flexibility for key users, which allows business users to adapt the UI to, for example, only show the fields most used by them. Using this tool, end users can change the UI without having to modify the underlying code. For more information, see <http://s-prs.co/v515755>.

7.3 SAP Cloud Platform Mobile Services

In this section, we'll dive into SAP Cloud Platform Mobile Services, which can help you create enterprise-grade business applications quickly. We'll discuss how various capabilities within this service can be used to quickly develop and deploy mobile applications for your end users, providing rich user experience. We'll first provide an overview of SAP Cloud Platform Mobile Services as a whole and its initial configuration. We'll then dive into some of the individual services: the mobile card kit (and the corresponding app, SAP Mobile Cards), the mobile development kit, SAP Cloud Platform SDK for iOS, and SAP Cloud Platform SDK for Android.

7.3.1 Product Overview and Configuration

SAP Cloud Platform Mobile Services provides various features and capabilities to empower your organization to quickly develop and deploy enterprise-grade secure mobile applications, thus enabling your business to offer the latest digital experience to customers and employees faster. SAP Cloud Platform Mobile Services supports development of different app types: native apps, hybrid apps, and web apps. Let's take a quick look at the various features that are available with SAP Cloud Platform Mobile Services:

■ Mobile cards

This is a capability within SAP Cloud Platform Mobile Services that enables developers to deploy a consumer-grade wallet or passbook-style app with little to no coding. Mobile cards provide a lot of easy-to-access and predefined templates that can be used to create simple apps.

Mobile cards also enable end users to mobilize SAP Fiori app web content, which allows them to view business content in the form of cards with the most up-to-date business content and view the same while offline. End users can perform actions on this content, such as approve or reject. End users can subscribe and unsubscribe from business cards to personalize their list.

The SAP Mobile Cards app needs to be downloaded from the Apple App Store or Google Play Store. Using this feature, you can quickly enable your end users to perform tasks like approve or view a sales order or purchase order, view product information, view pay slips for your employees, approve leave requests, and many more.

■ Mobile development kit

This capability within SAP Cloud Platform Mobile Services provides a metadata-based application development platform. This is a rapid mobile app development tool that can help your organization quickly create native mobile apps by minimizing the coding needed.

Applications can be developed rapidly with offline capabilities and robust business logic with native mobile support, and they provide an SAP Fiori user experience. Application development is done using web editor tools available in SAP Cloud Platform via the Mobile Development Kit editor plug-in for the SAP Web IDE full-stack service or SAP Cloud Platform Business Application Studio. The plug-in/template provides app templates, drag-and-drop features, additional wizards, and building blocks to create applications.

This feature will enable your organization to simplify app lifecycle management with a lower learning curve and higher developer productivity, letting you create native apps across platforms (iOS and Android) cost effectively.

■ SAP Cloud Platform SDK for iOS

This capability in SAP Cloud Platform Mobile Services lets organizations develop feature-rich native iOS enterprise mobile applications using iOS development features. The SDK is based on Apple's Swift programming language and supplements the Swift SDK. Apps are developed using the Xcode IDE. This simplifies developing iOS-native enterprise mobile apps that can take full advantage of iPhone and iPad features with easy access to core SAP S/4HANA data and business processes with mobile services.

■ SAP Cloud Platform SDK for Android

This capability in SAP Cloud Platform Mobile Services provides tools and capabilities to create native Android applications. This SDK is built on top of Google's Android SDK and provides an easy way to develop native Android enterprise applications. The SDK comes with a plug-in for Android Studio (a native Android project development IDE) that simplifies the development of Android projects by generating source code templates.

■ Mobile transaction bridge

The mobile transaction bridge in SAP Cloud Platform Mobile Services provides the capability to expose parts of SAP transactions by recording user interaction flows running in the web GUI as OData services. This can enable you to quickly build, develop, and mobilize your SAP transactions in the web GUI, bringing immense value to your end users, both employees and customers.

Thus, an exposed OData service can be consumed for building any applications, web or mobile, using the capabilities noted previously. This service is available only in the Cloud Foundry runtime.

Table 7.2 outlines additional features provided by SAP Cloud Platform Mobile Services.

Feature	Description
Mobile app catalog	This feature helps to manage apps during development, test apps for purposes like early beta release, and push apps to enterprise mobility management solutions to bring them to end users.
Mobile app update	This feature helps to manage and roll out updates for each application through its lifecycle.
Mobile client log update	This feature allows applications to upload application log files for analysis on the server.
Mobile client resources	This feature allows you to manage resources that can be accessed from mobile applications.
Mobile client usage and user feedback	This feature allows mobile applications to gather feedback from client usage to analyze further on the server.
Mobile cloud build	This feature allows you to easily build clients using the latest SAP SDKs in the cloud. This also allows you to create customized versions of standard SAP mobile applications like SAP Mobile Cards and implement your enterprise brand.
Mobile connectivity	This feature provides the functionality to securely access backend data and systems.
Mobile network trace	This feature enables applications to collect network trace information for debugging based on user name or content type.
Mobile offline access	This feature enables applications to run in offline mode by switching to a local data source on a device.
Mobile push notification	This feature enables you to notify your end users proactively of important events by registering devices to receive native push notifications. Mobile application developers don't need to implement specific code for the Apple Push Notification Service (APNS) or Firebase Cloud Messaging (FCM).
Mobile sample OData ESPM	This feature provides a sample OData service, which simplifies the application development.
Mobile settings exchange	This feature handles device registrations and provides an exchange of general settings between the mobile client and server.
Mobile transaction bridge OData	As explained earlier, this feature provides the capability to expose parts of SAP transactions by recording user interaction flows running in the web GUI as OData services.

Table 7.2 SAP Cloud Platform Mobile Services Features

Let's now look into how you can use and set up these various capabilities from SAP Cloud Platform Mobile Services to mobilize your business applications for your employees and customers, thus bridging the gap between IT and business.

To set up SAP Cloud Platform Mobile Services, you must meet the following prerequisites:

- You have a Cloud Foundry environment subaccount.
- You have assigned the SAP Cloud Platform Mobile Services plan to your subaccount from the **Entitlements** area.
- You have enough SAP Cloud Platform Application Runtime assigned to your space.
- You have an SAP Cloud Platform Destination service created for your SAP S/4HANA system already.

Once these prerequisites are met, you can begin your setup, as follows:

1. Inside your Cloud Foundry subaccount, navigate inside your space.
2. Navigate to **Services • Service Marketplace** and select the **Mobile Services** tile.
3. Click the **Support** link, enter **Organization** and **Space** information, and choose **Open**. This will open the SAP Cloud Platform Mobile Services cockpit page, as shown in Figure 7.4. During this process, SAP Cloud Platform Mobile Services performs a health check of your organization and will inform you of any issues it finds.

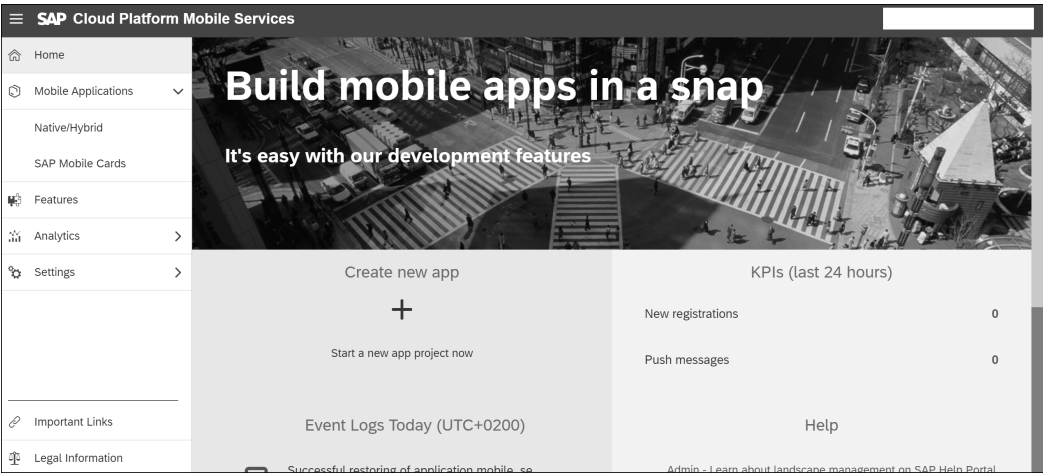


Figure 7.4 SAP Cloud Platform Mobile Services Home Screen

Note
Alternatively, you can also click the **Mobile Services** tile, navigate to **Instances**, and choose **New Instance**. This creates a new mobile application; every new mobile application created from within the mobile services cockpit will appear as instances in the future.

Navigate through the steps in the pop-up window, enter an **Instance Name**, and choose **Finish**. A new service instance of the mobile service is created in the service instance. Click **Actions • Open Dashboard**. This opens the SAP Cloud Platform Mobile Services cockpit as shown in Figure 7.4.

7.3.2 Mobile Card Kit

In this section, we'll look at the details of how you can create a mobile card kit application. Before you begin, you must have downloaded the SAP Mobile Cards app from the App Store or Google Play on your device. You can now follow these steps:

1. In the SAP Cloud Platform Mobile Services cockpit, navigate to **Mobile Applications • SAP Mobile Cards**.
2. This will open the home page for the mobile card kit, where you'll find details about various **Features** assigned; **Security, Configuration**, and **Card Templates** already created; a **Template Manager** in which predefined templates from SAP are provided; and other settings for SAP Mobile Cards.
3. Under the **Features** tab, update the SAP S/4HANA service destination under **Mobile Connectivity**.
4. Choose the **APIs** tab and scan the QR code for iOS or Android devices to onboard the SAP Mobile Cards client application. Go through the steps in your mobile device to onboard the client.
5. Navigate to the **Template Manager** tab, select **Sample Sales Orders Template**, and choose the **Create Card Template** action from the icon on the right side.
6. This opens a screen to create new mobile cards, as shown in Figure 7.5. Here you should provide a **Name** and **Version** for the application. Make sure the **Destination**, **Template Source**, and **HTML Template** are correctly set.

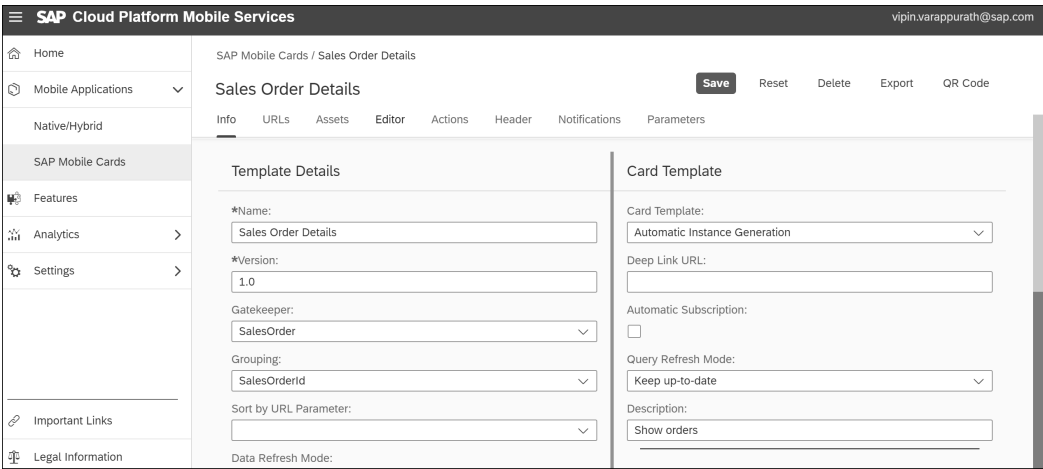


Figure 7.5 Mobile Card Kit Creation

7. You can choose to group your mobile cards in **Grouping**.
8. In **Card Template**, choose **Automatic Instance Generation**. This field provides the following options:
 - **Default**
These are card templates created via an explicit call made to create them.
 - **Welcome Card**
These are custom cards that are automatically downloaded after registration. Welcome cards are autosubscribed.
 - **Server Managed Card**
These are similar to welcome cards, but end users need to explicitly subscribe to these cards.
 - **Web Page Matching**
Allows the mobile cards kit plug-in installed in the SAP Fiori launchpad to automatically detect applications for which cards can be created.
 - **Automatic Instance Generation**
These are card sets that are automatically created based on a query entity set. This option helps the user to download a few specified instances of the object automatically.
9. In the **URLs** tab, you'll find the query URL, which points to the data endpoint definitions for the element you want to display in a card inside your connectivity service destination.

Note

If the user doesn't have access to this query URL, they can't subscribe to the card. For this, you can define gatekeeper settings under **SAP Mobile Cards • Configurations**, as follows:

1. Click **+**, enter a **Name**, select the **Destination**, and provide the **URL** data endpoint definition of the card template.
2. Select this defined gatekeeper setting for your card template under the **Info** tab in the **Gatekeeper** field.

This way, you can create gatekeeper settings to limit card subscriptions to specific roles associated with users.

10. The following are other functionalities available on the **Mobile Card App Creation** screen:
 - **Assets**
Add images for the card and define a separate resource Javascript file
 - **Editor**
Make changes to HTML5 or CSS code for the card, or create custom handlebar function definitions

- **Data Mapping**
Define handlebar bindings to an actual JSON path reference
 - **Actions**
Define an action to be performed from a mobile card
 - **Headers**
Create custom headers to be sent with the card
 - **Notifications**
Enable and define push notification rules for the card
 - **Parameters**
Define global parameters, URL parameters, or subscription parameters for the card
11. Click **Save**.
12. You can set the card into production from **Info • Versions**; select **Actions** to publish the card template to productive status. The new **State** is **Productive**.

You can test the card from the **Emulator** tab inside the mobile card kit application by choosing **Launch Emulator**, or subscribe to the card directly from your SAP Mobile Cards application on your mobile device as shown in Figure 7.6.

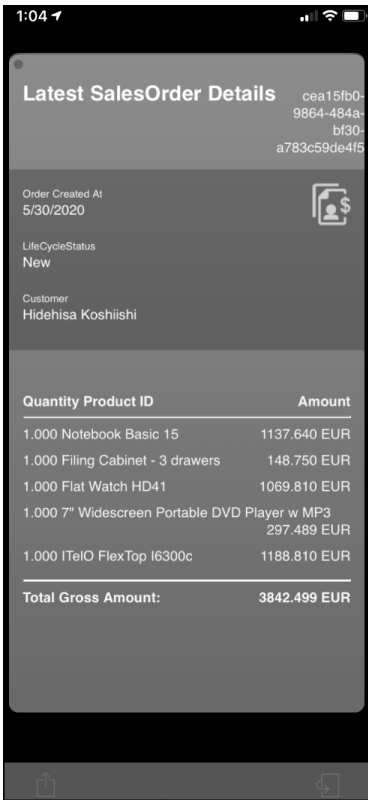


Figure 7.6 SAP Mobile Cards

Note

As mentioned earlier, you can enable mobile card kit configuration for your SAP Fiori launchpad to enable sharing content from the SAP Fiori launchpad directly to your device. You can learn more about this configuration at <http://s-prs.co/v515747>.

7.3.3 Mobile Development Kit

As mentioned, the mobile development kit is a rapid mobile application development platform for creating native mobile applications. In this section, we'll look at how to create your first native application using the mobile development kit to help you bridge the gap between your business expectations and IT backlog. The mobile development kit has following logical components:

- Development plug-in for SAP Web IDE full-stack service and SAP Cloud Platform Business Application Studio
- Mobile development kit client build
- SAP Cloud Platform application metadata definitions and runtime

Before you begin, you must have downloaded the SAP Cloud Platform Mobile Services client from your chosen app store and have SAP Cloud Platform Business Application Studio assigned to your subaccount. To create your first application, follow these steps:

1. Inside the SAP Cloud Platform Mobile Services cockpit, navigate to **Mobile Applications • Native/Hybrid**.
2. Click **New** and provide the details indicated in Table 7.3.

Field	Description
ID	<p>This is the unique identifier for the application, in reverse domain notation.</p> <p>The administrator uses this ID to register the application with SAP Cloud Platform Mobile Services, and client applications use the application ID when sending requests to the server.</p> <p>The recommendation is to provide an ID that contains a minimum of two periods—for example, <i>com.<org_name>.mobile.<app_name></i>.</p> <p>If it's an Android application, the ID must follow Google's defined rules: it must have one or more periods, each segment should start with a letter, and all characters must be alphanumeric or underscores.</p>
Name	An application name that can contain only alphanumeric characters, spaces, underscores, and periods and is up to 80 characters long.
Description (optional)	This is a description of the app.

Table 7.3 SAP Cloud Platform Native/Hybrid Application Definition

Field	Description
Vendor (optional)	The vendor that developed the application.
XSUAA Service	The XSUAA authentication and authorization service you want to use for the application. The default instance value would create a new instance, or you can select an existing service from the list.

Table 7.3 SAP Cloud Platform Native/Hybrid Application Definition (Cont.)

3. Click **Next**.
4. On the next screen, in the **Assign Features For** dropdown, select **Mobile Development Kit Application**, as shown in Figure 7.7.

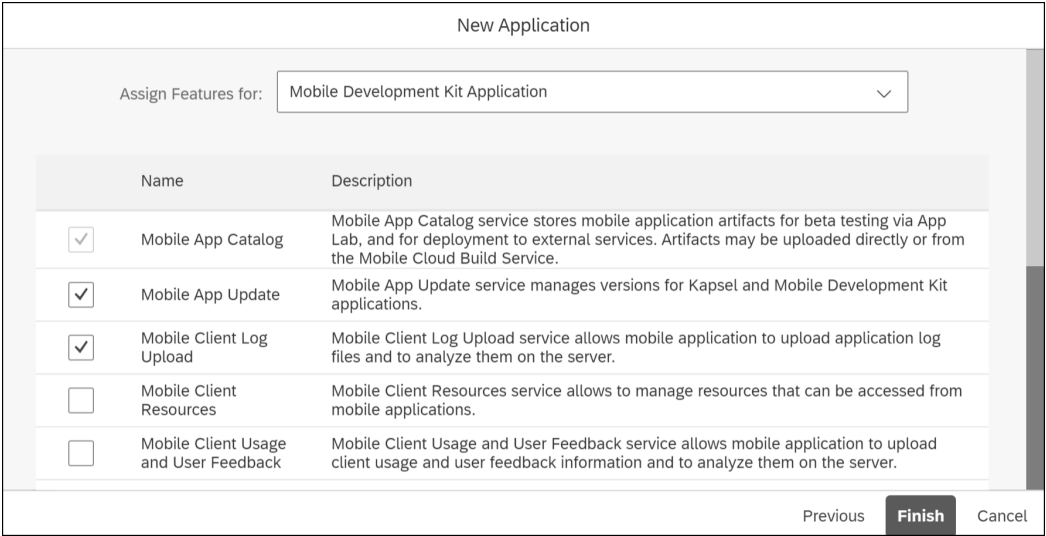


Figure 7.7 Mobile Development Kit Application Feature Selection

5. Select the features for the application.
6. Click **Finish** to create the application.
7. In the created application screen, update the **Mobile Connectivity** feature with the SAP S/4HANA destination.

Now you can further develop and configure this application using SAP Cloud Platform Business Application Studio, as follows:

1. Navigate inside your Cloud Foundry environment subaccount to **Subscriptions** and select **SAP Business Application Studio**.
2. Click the **Subscribe** button (if the service is not already subscribed).
3. Click **Manage Roles** and assign your user to the rights required to access SAP Cloud Platform Business Application Studio.

4. Click **Go to Application** to launch SAP Cloud Platform Business Application Studio.
5. Click **Create Dev Space**.
6. Enter a **Dev Space Name** and select the **SAP Cloud Platform Mobile Services** radio button as shown in Figure 7.8. Select other development tools you need in the space.

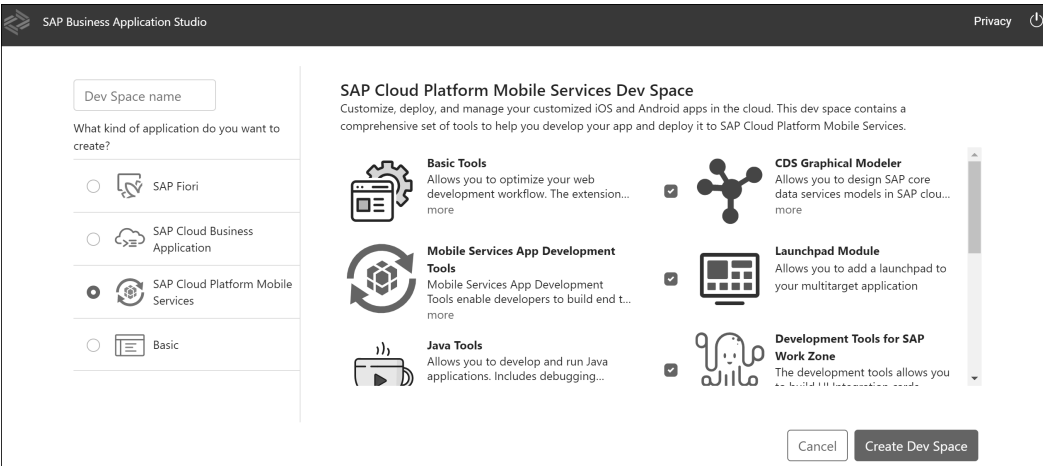


Figure 7.8 SAP Cloud Platform Mobile Services Dev Space Creation in SAP Cloud Platform Business Application Studio

7. Click **Create Dev Space** and a new dev space is created in SAP Cloud Platform Business Application Studio.
8. Select the space created. On the **Welcome** screen, select **New Project from Template**.
9. Specify a target path, select **MDK Project** for the template, and choose **Next**, as shown in Figure 7.9.

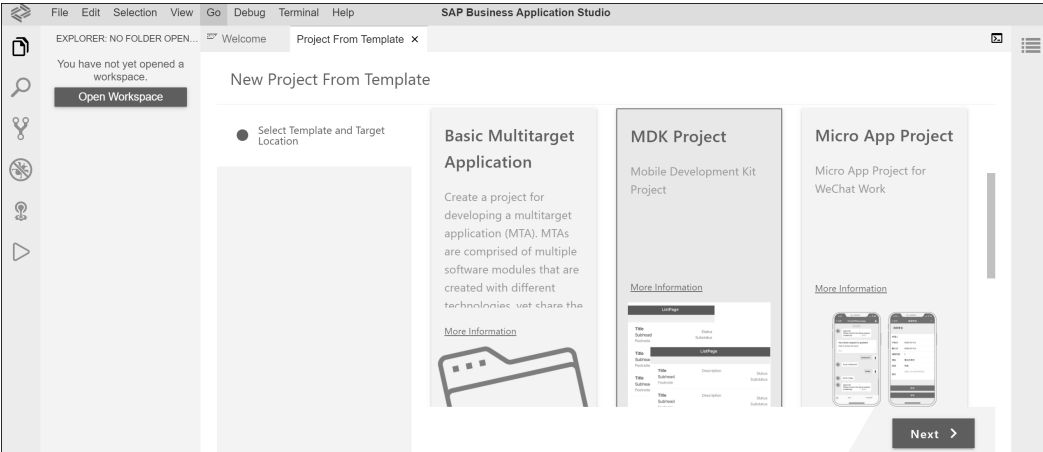


Figure 7.9 Create New Project from Template for Mobile Development Kit

10. Enter the **MDK Template Type**, **Project Name**, and **Application Name** and select **Next**.
11. Select your Cloud Foundry **Organization** and **Space** and choose **Next**.
12. Provide a **Service Name** and select the **Application ID** you provided while creating the application from the SAP Cloud Platform Mobile Services cockpit. Choose **Destination** and select **Next**.
13. Select the OData entity sets and choose **Next**. The project is generated with all necessary files.
14. Once you open the project in the workspace, you'll see the project configuration in an editor for mobile applications, as shown in Figure 7.10. The editor has following functionalities:
 - Edit pages with drag-and-drop controls, map them to OData service entity sets, and add custom controls or business logic.
 - Use the object browser to quickly locate and map elements to the screen, including actions, rules, services, styles, and UI elements.
 - Use the action editor to define custom actions like offline access, messages, and business logics.Here you define the application properties and application lifecycle events—for example, `OnLaunch` and `OnExit` in the *Application.app* file.
15. Right-click **Application.app** and choose **MDK:Deploy** to deploy the app to SAP Cloud Platform Mobile Services.

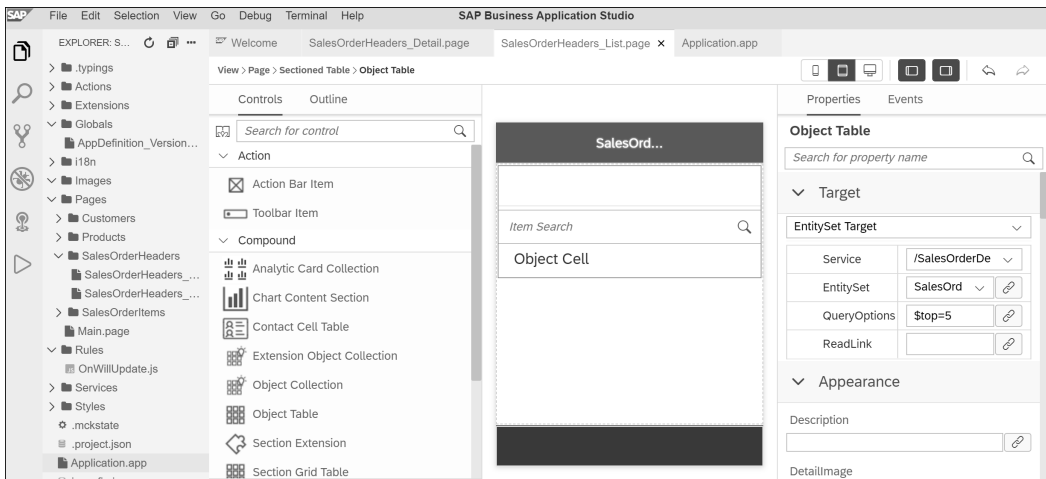


Figure 7.10 SAP Cloud Platform Mobile Development Kit Editor in SAP Cloud Platform Business Application Studio

You can now view this app in the SAP Cloud Platform Mobile Services client from any device.

Note

The mobile development kit provides the functionality to define and create your own mobile client to connect to your SAP Cloud Platform mobile application.

Mobile apps generated by the cloud-build service can only be distributed using enterprise distribution mechanisms.

To support this scenario, you can download the SDK for the mobile development kit from software downloads and build the client as explained at <http://s-prs.co/v515748>.

7.3.4 SAP Cloud Platform SDK for iOS

SAP Cloud Platform SDK for iOS is provided by SAP Cloud Platform Mobile Services. It's based on Apple's Swift programming language and helps in development of apps using the Xcode IDE. The SDK includes the SAP Cloud Platform SDK for iOS Assistant and comprehensive API documentation to support developers in quickly getting started and enabled to develop apps by generating code templates familiar to iOS developers.

The SDK Assistant builds the code template based on the components, and developers can further enhance their application code based on their requirements. SAP Cloud Platform SDK for iOS has following components for end-to-end application development:

- **SAPFiori framework**
This provides and implements the user experience for iOS apps. It implements SAP Fiori for the iOS design language and helps customize and brand applications to meet organizational standards.
- **SAPFioriFlows framework**
This framework provides the capability to complete user onboarding scenarios in the app.
- **SAPData frameworks**
This provides a data integration layer for both online and offline data access for your app. This framework parses OData payloads, produces OData requests, and handles responses.
- **SAPCommon and SAPFoundation frameworks**
These provide components to integrate your app with the SAP Cloud Platform environment. This includes functionalities like backend connectivity, authentication and authorization, logging functionality, and so on.
- **SAPML framework**
This framework provides general-purpose text-recognition capabilities, which helps in distributing new versions of core machine learning models to devices and handles local compilation and persistence. The models can be distributed remotely using SAP

Cloud Platform Mobile Services, thus allowing the app to use an updated version without redistribution of the app.

Developing in Xcode also allows your Apple developers to integrate the app with Apple UIKit or other third-party Swift components.

Before you begin to work with SAP Cloud Platform SDK for iOS, you must have met the following prerequisites:

- You have the Xcode editor installed (which only works on Macbooks).
- You have downloaded SAP Cloud Platform SDK for iOS from the download center at <https://developers.sap.com/trials-downloads.html>.

To set up and configure app development using SAP Cloud Platform SDK for iOS, follow these steps:

1. Extract the downloaded SAP Cloud Platform SDK for iOS on your device.
2. Open the DMG file to find the SAP Cloud Platform SDK for iOS Assistant.
3. Drag and drop **SAP Cloud Platform SDK for iOS Assistant** to the **Applications** folder on your Apple Macbook.
4. Click **SAP Cloud Platform SDK for iOS Assistant** in the **Applications** folder to launch the assistant.
5. In the SAP Cloud Platform Mobile Services cockpit, navigate to **Mobile Applications • Native/Hybrid**.
6. Choose **New**, enter application fields as shown in Table 7.3, and click **Next**.
7. Select **Native Application** under **Assign Features For**. Select the features required for the application and choose **Finish**. The app is created.
8. In the **Mobile Connectivity** feature, update your destination to the OData service you want to use for the application.
9. In the SAP Cloud Platform Mobile services cockpit, navigate to **Important Links** and click the **Import URLs Directly (Only Supported for iOS Assistant)** link. This will open your SAP Cloud Platform SDK for iOS Assistant with populated links for SAP Cloud Platform Mobile Services, as shown in Figure 7.11.
10. Enter a **Name** for your SAP Cloud Platform mobile account and a **Username** and **Password** to login to the account, and click **Save**.
11. In the SAP Cloud Platform SDK for iOS Assistant, click **Manage Accounts • Add new....** For **Type**, select from the following:
 - Select **SAP Cloud Platform Developer Portal** to configure your SAP Cloud Platform API Management developer portal to use APIs exposed using this service while creating an iOS application.
 - Select **SAP Cloud Platform Translation Hub** to configure SAP Translation Hub, covered in the next section, to localize the application for various languages.

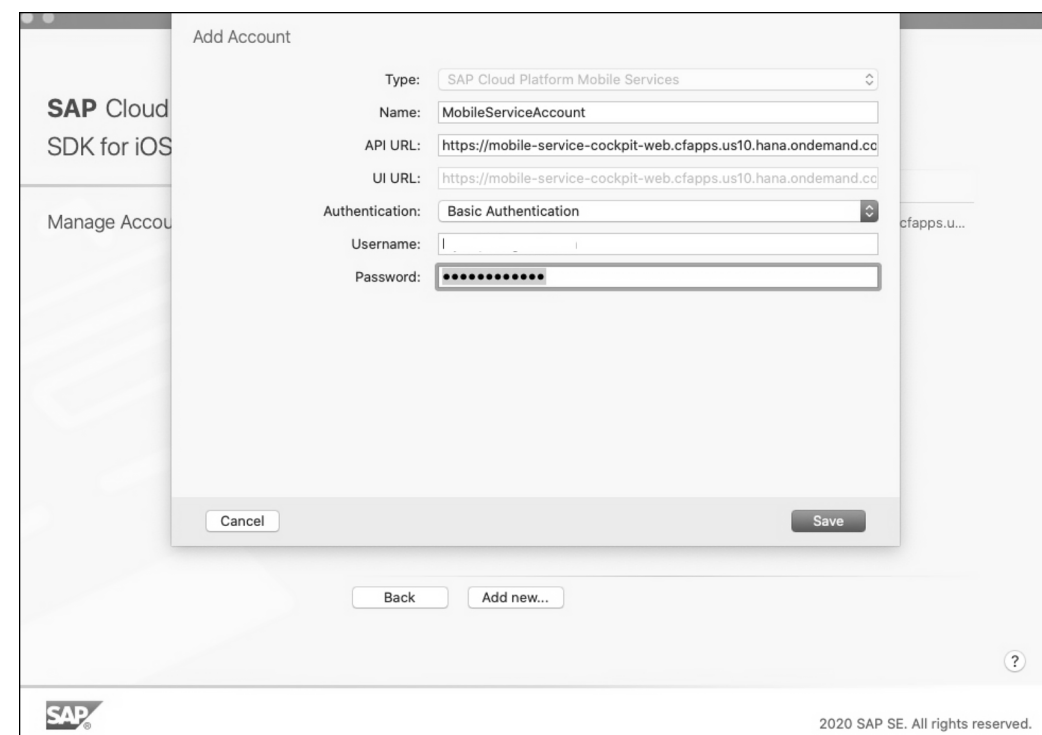


Figure 7.11 SAP Cloud Platform SDK for iOS Assistant: Manage Accounts

12. In SAP Cloud Platform SDK for iOS Assistant, click **Create New**. On the next screen, you can select a project template to create your application. For this example, select the **Reuse Existing Application** template.
13. In the next screen, **Account**, select the SAP Cloud Platform Mobile Services account you added in the previous step and click **Next**.
14. On the next screen, **Cloud Application**, select the cloud application you had created in the SAP Cloud Platform Mobile Services cockpit in previous step and choose **Next**.
15. On the next screen, **Xcode Project**, provide a **Product Name**, **Organization Name**, **Organization Identifier** (these names need not match the names that you have set in the SAP Cloud Platform Mobile Services cockpit), and **Path** to save the project. Click **Next**.
16. On the next screen, **Proxy Classes**, choose the destination to create the app and click **Next**.
17. On the next screen, **UI Configuration**, select the templates for UI generation and choose **Finish**.
18. The generated project will open in your Xcode IDE, where you can make required changes to the project and run the app in simulator mode, as shown in Figure 7.12.

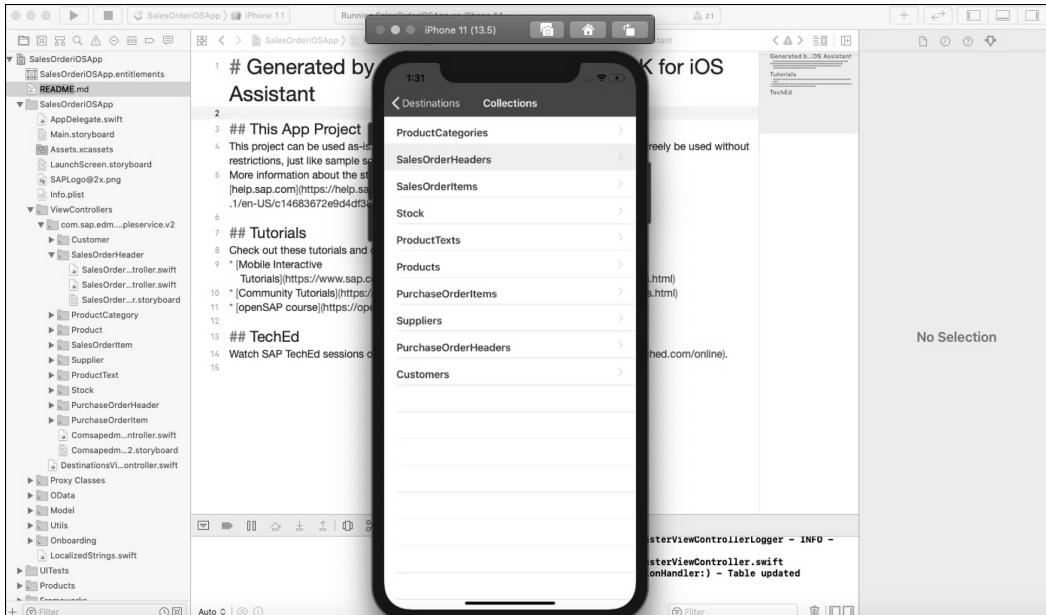


Figure 7.12 SAP Cloud Platform SDK for iOS Project in Xcode

Your first SAP Cloud Platform SDK for iOS-based app is now ready.

7.3.5 SAP Cloud Platform SDK for Android

SAP Cloud Platform SDK for Android provides tools and capabilities to create enterprise-grade native Android applications using SAP Cloud Platform Mobile Services. The SDK is based on Google’s Android SDK and has an Android Studio project plug-in that simplifies project development.

The SAP Cloud Platform SDK for Android has the following components, similar to SAP Cloud Platform SDK for iOS:

- **SAP Fiori component**
Provides a library of UI modules for creating mobile applications that follow the SAP Fiori design language and Android Material Design
- **Flows component**
Provides implementation for an app’s onboarding scenarios
- **OData component**
Provides capabilities that allow your app to access OData for both online and offline mode
- **Foundation component**
Provides capabilities that allow your app to use features and capabilities of SAP Cloud Platform like authentication, push notification, logging, and more

Before you begin to work with SAP Cloud Platform SDK for Android, you must have met the following prerequisites:

- You have the Android Studio IDE installed.
- You have downloaded SAP Cloud Platform SDK for Android from the Download Center at <https://developers.sap.com/trials-downloads.html>.

To set up and configure app development using SAP Cloud Platform SDK for Android, follow these steps:

1. Extract the contents of the downloaded Android SDK and run the installation script.
2. Set the SAP_ANDROID_HOME environment variable to your installation folder (optional).
3. From the SAP Cloud Platform Mobile Services cockpit, navigate to **Mobile Applications • Native/Hybrid**.
4. Choose **New**, enter application fields as in Table 7.3, and click **Next**.
5. On the next screen, select **Native Application** in the **Assign Features For** dropdown. Select the features required for the application and choose **Finish**. The app is created.
6. In the **Mobile Connectivity** feature, update your destination to the OData service you want to use for the application.
7. From the SAP Cloud Platform Mobile Services cockpit, navigate to **Important Links** and select **Copy Admin API** and **Copy Admin UI**.
8. Launch Android Studio. You’ll see the wizard for SAP Cloud Platform Mobile Services-based Android projects, as shown in Figure 7.13.

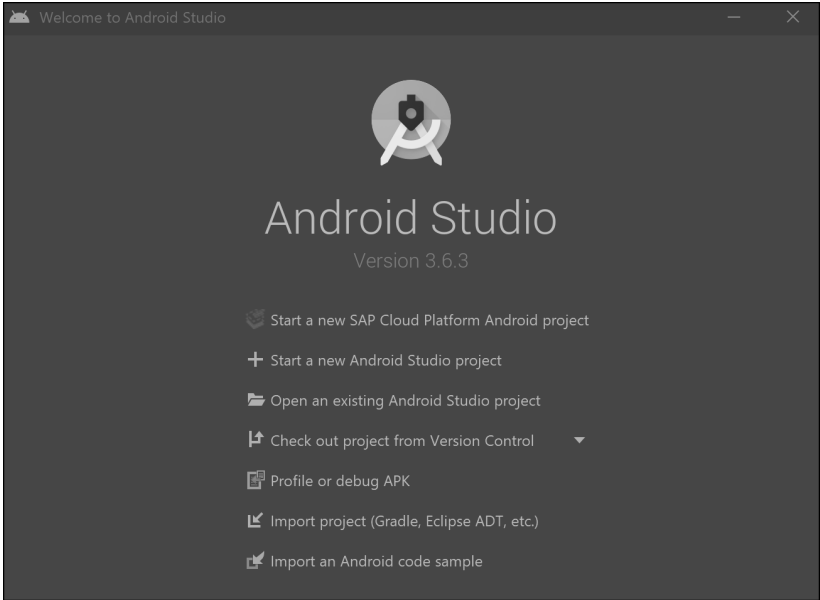


Figure 7.13 Android Studio with SAP Cloud Platform SDK for Android Installed

9. Select **Start a New SAP Cloud Platform Android Project** and provide the following details on the screens that follow:
- **Server Configuration**
Provide the **Account name**, the **Admin API URL** and **Admin UI URL** copied from the previous step, and authentication details to log in to your SAP Cloud Platform Mobile Services account, then click **Next**.
 - **Cloud Configuration**
Choose to use an existing app, create from scratch, or create a sample app from here. Because you created an existing app from the SAP Cloud Platform Mobile Services cockpit earlier, choose **Use Existing App**, provide the **Application ID**, and click **Next**.
 - **OData Services**
Select the OData service for the application and click **Next**.
 - **Android Studio Project**
Provide the **Project Name**, **Project Namespace**, and **Project location** and choose the target language (Java or Kotlin) for your project to be created. Click **Next**.
 - **Project Features**
Select the relevant project features, like enabling logging, usage reporting, push notification configuration, OData access for online or offline use, and so on.
10. Click **Finish**.

This will generate a project template with lines of code that supports CRUD access to the OData service selected. The project will include SAP Fiori screens and OData and foundation libraries used for the project’s specific activities. You can open the project in Android Studio and run it in the emulator to test it.

7.4 SAP Translation Hub

In previous sections, we’ve explained why an optimum user experience is important in this digital transformation era and how you can use certain tools to create these experiences for your customers and employees quickly. What also contributes to the end user experience and further enhances it is the ability to provide a UX in the user’s native language. A portal application or mobile application presented in a user’s native language not only enhance the user experience, but also helps drive user adoption and end user satisfaction.

For enterprise applications, it’s quite important to understand the business context for a translation. This is defined as a *domain* in SAP Translation Hub. SAP Translation Hub helps you translate your UI texts via SAP-approved translations and terminology that take the business context (the domain) into consideration during machine translation.

The translation can be done using this service at a quick pace, thus enabling you to roll out translated screens and user experiences to end users quickly. You can access the translation resources by using integrated workflow scenarios in a UI or by consuming a range of API methods. SAP Translation Hub integrates natively with the following SAP Cloud Platform user experience categories to help quickly translate applications:

- SAP Cloud Platform Mobile Services
 - Mobile development kit
 - SAP Cloud Platform SDK for iOS
 - SAP Cloud Platform SDK for Android
- Git repository
 - On-premise Git repository
 - Web-based Git repository
 - SAP Cloud Platform Git repository

SAP Translation Hub comprises the following translation providers:

- A repository of multilingual texts from SAP applications referred as the SAP-provided MLTR (multilingual text repository). You can optionally create your own multilingual repository by uploading your own language data to SAP Translation Hub to use this as the first-choice translation provider. This is referred to as *company MLTR*.
- Machine translation, which is verified by SAP-certified language experts.

SAP Translation Hub runs in the Neo environment. Before you begin work, you must have a subaccount based on the Neo environment. To start using SAP Translation Hub, follow these steps:

1. Navigate inside your subaccount to **Services** and select the **SAP Translation Hub** tile.
2. Click **Enable**.
3. Click **Configure Service** and navigate to **Roles**. Make sure your user is assigned to the predefined role called *user*.
4. Click **Go to UI for Translation Workflow** to open the translation project home screen, as shown in Figure 7.14.

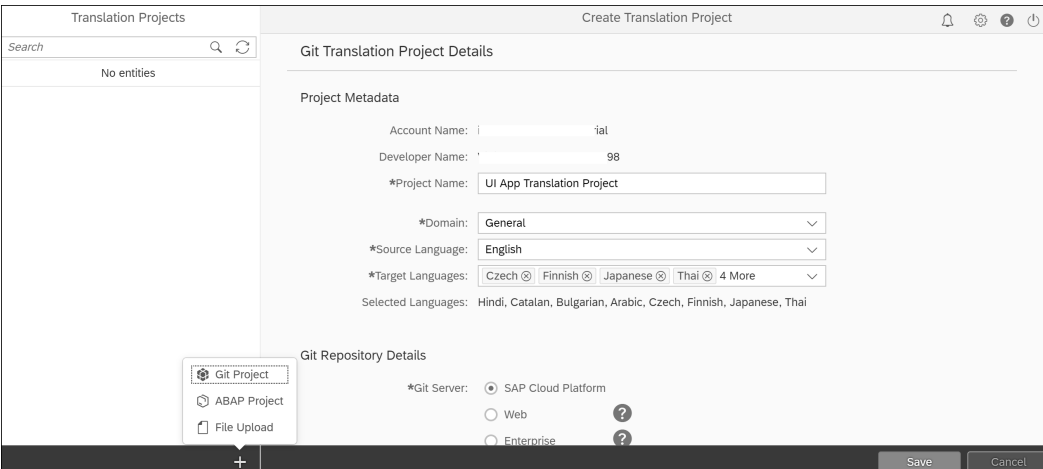


Figure 7.14 SAP Translation Hub

5. Click **+** to create a translation project from a **Git Project**, from an **ABAP Project**, or via **File Upload**. For this example, choose **Git Project**.
6. On the next screen, enter a **Project Name**, **Domain**, **Source Language**, **Target Language**, and **Git Repository Details**. You can select multiple target languages as required.
7. You can choose additional settings under **Project Configuration** for **Keep Translations**, **Post-Editing Required**, **Review Required**, or **Use Company MLTR**.
8. Select **Save**. An overview of the translation project is displayed.
9. To start the translation project, choose **Translate and Push**.
10. You'll be asked for your Git repository password. Enter it and click **Submit**.
11. On the subsequent screen, choose the **Translations** tab to see the translated text. This can be updated based on the *quality index*, a score of 1–100 indicating the quality of the translation.
12. Click **Push** to push the translation to the Git repository.
13. The translation is made available to the project by executing a pull request from Git. You'll find the translations in the following places:
 - For SAP Fiori apps, the i18n folder in your project now contains a new properties file for translated languages.
 - For SAP Cloud Platform SDK for iOS apps, the translations are defined in `localizable.strings`.
 - For SAP Cloud Platform SDK for Android apps, the translations are defined in `strings_localized.xml`.

Note

As mentioned, SAP Translation Hub integrates with other SAP Cloud Platform services like SAP Cloud Platform SDK for iOS. Hence the translation project can be triggered from those services as well from a developer/user flow perspective.

7.5 Summary

Defining a consistent and seamless user experience for end users of your system is very critical in your transformation journey. This ensures end user adoption and satisfaction, adding value to your business. Furthermore, the user experience and expectations are fast changing in the current world and you need tools and technologies to catch up with these ever-changing requirements.

In this chapter, we looked at the SAP Cloud Platform Launchpad service, which can bring applications from your multiple SAP systems and custom applications together in a single place, providing your employees with a consistent user experience. We also

looked into SAP Work Zone, which brings together different SAP Cloud Platform capabilities in a coherent way, providing a seamless and delightful end user experience.

Because mobile devices are changing UX expectations and as organizations are considering *work from anywhere* approaches, it becomes increasingly important for organizations to provide enterprise application access on mobile devices. We looked at various SAP Cloud Platform Mobile Services capabilities for different use cases and developer experiences, which can quickly and rapidly create mobile apps in both online and offline mode and provide a delightful user experience.

Finally, we saw how you can enable business translations for your user experience applications in an easy and consistent way across the enterprise, thus further helping you to enhance your business user experience and drive adoption and satisfaction.

Contents

Preface	15
---------------	----

PART I Foundation

1	Future-Ready Enterprise Architecture	29
----------	---	----

1.1	What Is a Future-Ready Enterprise?	30
1.2	Why Is SAP Cloud Platform Essential for a Future-Ready Enterprise?	31
1.2.1	Role of SAP Cloud Platform	32
1.2.2	SAP Cloud Platform and the Intelligent Enterprise	34
1.2.3	Elevate Your Enterprise Integration and Extension Experience	35
1.2.4	Pricing and Packaging	37
1.2.5	Service Catalog	38
1.3	Realms in the Future-Ready Enterprise	39
1.3.1	Business Foundation: Understanding Your Current Process	41
1.3.2	Future Aware: Architecting and Designing	42
1.3.3	Future Focused: Cloud Application Development	42
1.3.4	Trailblazer: Manage Innovations	43
1.4	Summary	43

2	Understanding Your Current Business Processes and Landscape	45
----------	--	----

2.1	Current Business Processes	46
2.2	Future-Aware Business Processes	47
2.3	Tools and Frameworks	50
2.3.1	SAP Process Mining by Celonis	50
2.3.2	Spotlight by SAP	54
2.3.3	SAP API Business Hub	54
2.3.4	Ruum by SAP	56
2.4	Outcomes	59
2.5	Summary	61

PART II Future Aware

3 Architecture and Design Considerations for a Future-Ready Landscape 65

3.1 Future-Aware Landscape 65

3.2 Cloud Architecture and the TOGAF Standard 67

3.3 Platform Advisory Methodology 68

3.3.1 Platform Capability Model Overview 69

3.3.2 Foundation 71

3.3.3 Future Aware 73

3.3.4 Future Focused 76

3.3.5 Trailblazers 77

3.4 Data Architecture Principles 78

3.5 Design Outcomes 80

3.5.1 Technology Platform Principles 81

3.5.2 Integration Principles 81

3.5.3 Development Principles 82

3.5.4 User Experience Principles 82

3.5.5 Database Requirement 83

3.6 Summary 84

4 Master Data Distribution 85

4.1 Master Data Distribution Use Cases 85

4.2 Developing a Master Data Distribution Strategy 88

4.2.1 SAP Cloud Platform Master Data for Business Partners 89

4.2.2 SAP Cloud Platform Data Enrichment 98

4.2.3 SAP Data Quality Management, Microservices for Location Data 101

4.2.4 Data Attribute Recommendation 104

4.3 Compliance with Master Data Retention and Deletion 105

4.4 Summary 110

5 Process and Data Integration 111

5.1 Integration Use Cases and SAP Integration Solution Advisory Methodology 112

5.1.1 Process Invocation Use Case Patterns 113

5.1.2 Data Movement Use Case Patterns 114

5.1.3 User Consumption 115

5.1.4 Internet of Things Integration 116

5.2 SAP Cloud Platform Connectivity and Cloud Connector 117

5.3 SAP Cloud Platform Integration Suite 121

5.3.1 Configuration 122

5.3.2 Cloud Integration 124

5.3.3 API Management 128

5.3.4 Open Connectors 132

5.3.5 Integration Advisor 134

5.4 SAP Cloud Platform Enterprise Messaging 136

5.4.1 Product Overview 136

5.4.2 Configuration 139

5.5 SAP Data Intelligence 142

5.6 SAP Internet of Things 144

5.7 Defining Integration Pattern Design 146

5.8 Integration with Legacy Applications 148

5.8.1 Creating User-Provided Service Instances 148

5.8.2 Service Management 149

5.8.3 Other Integration Methods 150

5.9 Summary 151

6 Business Process Automation and Optimization 153

6.1 Build Efficient Automations in Business Processes 153

6.1.1 Services and Products on SAP Cloud Platform 154

6.1.2 SAP-Delivered Scenarios 157

6.2 Robotic Process Automation 159

6.2.1 Cloud Factory 160

6.2.2 Desktop Studio 162

6.2.3 Desktop Agent 163

6.3	Business Process Visibility and Management	165
6.3.1	Activating SAP Cloud Platform Workflow Management and SAP Cloud Platform Process Visibility	166
6.3.2	Integrating SAP Cloud Platform Process Visibility with SAP Cloud Platform Workflow Management	168
6.3.3	SAP Cloud Platform Process Visibility Product Attributes	170
6.4	Automation Use Cases and Decision Matrix	171
6.4.1	Selection	171
6.4.2	SAP-Delivered Scenarios	172
6.4.3	Transformation Cases	173
6.4.4	Decisions and Criteria for Success	175
6.5	Summary	176
7	User Experience and Mobile Consumption	177
7.1	SAP Cloud Platform Launchpad	178
7.2	SAP Work Zone	180
7.3	SAP Cloud Platform Mobile Services	182
7.3.1	Product Overview and Configuration	182
7.3.2	Mobile Card Kit	186
7.3.3	Mobile Development Kit	189
7.3.4	SAP Cloud Platform SDK for iOS	193
7.3.5	SAP Cloud Platform SDK for Android	196
7.4	SAP Translation Hub	198
7.5	Summary	200
8	Security	203
8.1	Authentication Management	204
8.1.1	Cloud Foundry Environment	204
8.1.2	Default Authentication in SAP Cloud Platform	204
8.1.3	Identity Authentication Service	205
8.1.4	SAML 2.0 Identity Provider	208
8.2	Authorization Management	209
8.2.1	Cloud Foundry Environment	209
8.2.2	Neo Environment	218

8.3	Credential Store Service and Keystore Service	220
8.3.1	Cloud Foundry Environment	220
8.3.2	Neo Environment	221
8.4	OAuth 2.0 Application Protection	222
8.5	Secure Platform Operations	222
8.5.1	Predefined Roles and Functions	222
8.5.2	Custom Platform Roles	225
8.5.3	Platform Identity Provider	226
8.6	Identity Provisioning Service	227
8.7	SAP Cloud Identity Access Governance	229
8.8	Audit Logging	230
8.8.1	Cloud Foundry Environment	230
8.8.2	Neo Environment	231
8.8.3	Identity Authentication Logs	232
8.9	Summary	232

PART III Future Creation

9	Developing SAP S/4HANA Extensions	235
9.1	Building Extensions to Differentiate	236
9.2	SAP Cloud Platform’s Extension Capabilities	238
9.3	SAP Cloud Platform Runtimes	242
9.3.1	SAP Cloud Platform Application Runtime	242
9.3.2	SAP Cloud Platform Serverless Runtime	253
9.3.3	SAP Cloud Platform, ABAP Environment	261
9.3.4	SAP Cloud Platform, Kyma Runtime	266
9.3.5	Decision Matrix	268
9.4	Integrating Applications with Hyperscaler Services	269
9.4.1	Resource Provider	270
9.4.2	Service Broker	271
9.5	Summary	274

10	Continuous Integration and Delivery	275
10.1	Continuous Integration and Delivery Overview	275
10.1.1	Continuous Integration	276
10.1.2	Continuous Delivery	277
10.1.3	SAP Cloud Platform Services	279
10.2	Building a DevOps Strategy and Plan	280
10.2.1	Building Your Strategy and Landscape in SAP Cloud Platform	280
10.2.2	Managing Roles and Responsibilities	283
10.3	SAP Cloud Platform Transport Management	283
10.3.1	Multitarget Applications and Multitarget Application Archive	284
10.3.2	Set Up SAP Cloud Platform Transport Management	288
10.4	Project “Piper”	294
10.4.1	Overview	294
10.4.2	Configuring SAP Cloud SDK Pipeline	296
10.4.3	Integrate SAP Cloud Platform Transport Management with Pipeline	299
10.5	SAP Cloud Platform Continuous Integration and Delivery	301
10.5.1	Service Configuration	301
10.5.2	Access the Service	302
10.5.3	Configure GitHub Hook	304
10.5.4	Access SAP Cloud Platform Continuous Integration and Delivery from SAP Web IDE	305
10.6	Summary	306
11	Continuous Monitoring	307
11.1	Logging	307
11.1.1	Cloud Foundry	307
11.1.2	Neo	311
11.1.3	Application Performance Monitoring with Dynatrace	313
11.2	Alerts and Remediation	315
11.2.1	SAP Cloud Platform Alert Notification	315
11.2.2	SAP Cloud Platform Automation Pilot	317
11.3	Summary	321

12	Developing Cloud-Native Applications	323
12.1	Resiliency Principles	324
12.1.1	Service-Level Agreements	324
12.1.2	Availability and Resiliency Patterns	326
12.2	High-Availability Best Practices with SAP Cloud Platform	330
12.2.1	Cloud Connector High Availability	331
12.2.2	Caching Setup	332
12.2.3	Rate Limiting	334
12.2.4	Application Autoscaler	335
12.2.5	Application Resiliency	337
12.2.6	SAP Cloud Platform Application Deployment	338
12.2.7	Logging	340
12.3	Multi-Data Center Failover	340
12.4	Summary	341
PART IV Final Steps		
13	Trailblazing Enterprise Innovations	345
13.1	Industry Trends and Technology Trends	345
13.2	Industry Challenges and Reimagined Business Processes	347
13.2.1	High-Tech Industry	347
13.2.2	Consumer Products Industry	348
13.2.3	Industrial Machinery and Components Industry	348
13.2.4	Travel and Transportation	349
13.2.5	Banking Industry	350
13.2.6	Telco Industry	351
13.2.7	Utilities Industry	352
13.2.8	Engineering, Construction, and Operations Industry	352
13.2.9	Automotive Industry	353
13.2.10	Retail	354
13.3	Addressing Whitespaces and Gaps	354
13.4	Frameworks and Prioritization	356
13.4.1	Innovation Engagement Framework	356
13.4.2	Tools	360
13.5	Summary	362

14 Measuring Success	363
14.1 Farming Framework	364
14.2 Harmonization and Optimization	365
14.2.1 System Harmonization	366
14.2.2 DevOps for Continuous Integration and Continuous Delivery	367
14.2.3 Replatforming the System Landscape	368
14.3 Innovation Framework	369
14.4 Value Realization Engagement	370
14.4.1 Business Benefits	370
14.4.2 Planning Tool	371
14.5 Summary	372
 The Authors	 373
Index	375

Index

A

ABAP Test Cockpit	261
Actions	315
Advanced Message Queuing Protocol	137, 150
Aggregated View app	100
Alerts	315
AND logic	315
Android Studio	197
ANSI X12	135
API designer	129
API endpoint	310
API gateway	129
API Management	73, 128–129, 317
<i>caching policy</i>	332
<i>components</i>	129
<i>onboard API portal</i>	131
<i>rate limiting</i>	334
API portal	129
<i>account type</i>	131
<i>home screen</i>	132
<i>role collections</i>	130
<i>roles</i>	131
APIs	128, 146, 186, 241, 366
<i>manage, govern, and secure</i>	146
Application architecture	69, 73, 76
Application Autoscaler	335
<i>setup</i>	336
Application monitoring	312
Application patterns	237
Application performance monitoring	313
Application resiliency	337
Application router	204
Application security	210
Application security descriptor file	212–213
Application SLA	325–326
Application-level security	204
Application-to-application (A2A)	112, 114
Architecture and design	65
<i>design outcomes</i>	80
Architecture development model (ADM)	67
Artificial intelligence (AI)	346
ASC X12	134
Assertion attributes	207, 220
Attended RPA	158
Audit Log Viewer	231
Audit logging	230

Authentication	204, 216
<i>default</i>	204
Authorizations	102, 209, 216
<i>Cloud Foundry</i>	209
<i>Neo</i>	218
Automation	153
<i>benefits</i>	159
<i>customer service and onboarding</i>	174
<i>data maintenance</i>	175
<i>decision matrix</i>	171
<i>financial transformation</i>	173
<i>good candidates</i>	172
<i>operations</i>	175
<i>process selection</i>	171
<i>SAP-delivered scenarios</i>	172
<i>scenarios</i>	164
<i>shared services</i>	174
<i>suitable processes</i>	175
<i>use cases</i>	171
Automotive industry	353

B

Banking industry	350
Bimodal IT approach	39
Blue-green deployment	338
Bots	116, 158, 164
<i>deploy</i>	162
Business Configuration app	91
Business Logs app	92
Business models	60
Business object instance	97
Business partners	247, 366
Business process discovery workshop	49
Business Process Model and Notation (BPMN)	124
Business processes	45
<i>as-is</i>	46
<i>automation</i>	153
<i>future-aware</i>	47
<i>mapping</i>	60
<i>outcomes</i>	59
<i>reimagined</i>	347
<i>SAP API Business Hub</i>	54
<i>tools and technologies</i>	50
Business Technology Platform	31, 34, 41, 46, 59, 66, 78
<i>capabilities</i>	79

Business-to-business (B2B) 112, 114
Business-to-government 114

C

Cache-aside 333
Caching 330, 333
 setup 332
Case explorer 51
Cases 51
Catalogs 318
CDQ 87, 99
Change management 277
CIDR IP range 143
Circuit breaker 328
Client secret 232
Cloud architecture 67
Cloud auditor 71
Cloud broker 71
Cloud carrier 71
Cloud connector 118
 access 119
 admin 224
 administration 120
 authentication 121
 backend system 120
 configure 118
 high availability 331
 home screen 119
 subaccount ID 119
Cloud consumer 70
Cloud Foundry 33, 89, 160
 artifacts for authorization 212
 audit logging 230
 authorizations 209
 CLI 217
 command-line interface 310
 Dynatrace 314
 logging 307
 organizations 23
 security 204
 spaces 224
 subaccount roles 223
Cloud Integration 124
 design interface 126
 home screen 126
 monitoring 127
 prepackaged integration content 126
 prerequisites 124
 roles 125
Cloud provider 70
CloudEvents 139

Cloud-native applications 323
Commands 318
Company MLTR 199
Concurrent rate limit 335
Conditions 315–316
Configure Business Scenarios app 169
Consumer products industry 348
Consumption-based model 37
Continuous delivery 277
 cycle 277
Continuous integration 276
Continuous integration and delivery
 (CI/CD) 275, 367
 overview 275
 pipelines 301
 SAP Cloud Platform 279
Continuous monitoring 307
Core data services (CDS) 249
 views 237
Corporate Git 279
Cost separation 280
Cx Server 297

D

Data architecture pattern 79
Data architecture principles 78
Data Attribute Recommendation 104, 156
 setup 105
Data Consumption app 101
Data exploration 80
Data governance 79
Data integration 80
Data management 78
Data marts 114
Data migration 80, 115
Data model fields 103
Data modeling 79
Data movement 114
Data orchestration 79, 115
Data quality 79, 102, 115
Data regulation 280
Data retention 105
Data scope 98
Data Search app 101
Data security 80
Data storage 80
Data subject 108
Data subscription plans 99
Data transparency 346
Data virtualization 115
Data warehouse 114

Database requirements 83
Delete Data Subject Information app 109
Deployment pipeline 276
Design thinking 47–48
Design-to-value 358
Dev spaces 279
Developer portal, role collections 130
Development principles 82
Device management 145
DevOps 278, 367
 strategy 280
Digital innovation 40
Digital prototyping workshop 359
Digital Twins 352
Directories 22, 280
Discovery workshop 358
Display Distribution Status app 95
Distribution models 95
DMZ 118
Docker images 295, 297
Document Classification 156
Document Information Extraction 156
Domains 198
Dun & Bradstreet 87, 99
Dynatrace 313
 application logs 314
 connectivity 314

E

Edge platform 144
End-to-end process blueprints 54
Engineering, construction, and operations (EPC)
 companies 352
Enterprise application extensions 37
Enterprise architecture 29, 67
Enterprise integration 35–36
Entitlements 20
Events 141
Execution framework 357
Executive briefing workshop 358
Extended service for UAA (XSUAA) 204, 217
 authentication 190
Extension Center 257
Extensions 35, 236, 260

F

Failover 341
Farming framework 355, 364
Formulas 134

Future-ready enterprise 30, 47
 business foundation 41
 future aware 42
 future focused 42
 stages 39
 trailblazers 43

G

Gartner's top 10 strategic technology 347
General Data Protection Regulation
 (GDPR) 85, 105
Git repository 199
GitHub webhooks 304
Global accounts 20, 226, 239
 roles 223
Google Cloud Platform 272
Graph databases 84

H

High availability 327
 best practices 330
 master instance 331
High-tech industry 347
Hire to retire 55
Horizontal scaling 335
HTML5 applications 218
HTTP RESTful APIs 138
Hyperautomation 346
Hyperscalers 36
 services 269

I

Identity Authentication 205, 226
 administration console 206
 configuration 207
 custom identity provider 209
 establish trust 207
 Neo environment 208
 service logs 232
 setup 205
 user groups 216
Identity provider 208, 213
 assign group to role 219
 attributes 211
 corporate 208
 platform 226
 role collections 215
 user store 226

Identity Provisioning 227

- setup* 227
- source systems* 227
- target system* 228
- transformations* 229

In-app extensibility 236

Industrial machinery and components

- industry 348

Industry trends 345

Initial data load 115

Innovation 46, 345

- framework* 356, 360, 369
- planning workshop* 357

Inputs 318

Insight to action 47

Insight to design 47

Integration Advisor 134

- home page* 135
- setup* 135

Integration competency center (ICC) 81

Integration flows 124

- endpoints* 127
- failure* 317

Integration patterns 112

- design* 146

Integration platform as a service (iPaaS) 81

Integration principles 81

Integration styles 146

- design patterns* 147

Intelligent enterprise 31, 121

Internet of Things (IoT) 116, 346

J

Java applications 218

Java Connector (JCo) 312

JCache 332

Jenkins instance 297

Jenkins server 297

Jenkinsfile 295, 298

- project structure* 299

JSON file 140, 242, 254, 300

K

Key mapping 98

Key Mapping app 91

Key performance indicators (KPIs) 54

Kibana dashboard 311

Kubernetes 150, 320

L

Landscape 45

- as-is* 67
- future proof* 363
- future-aware* 65–66
- harmonization* 365
- optimization* 365
- system types* 112
- tiering* 281–282
- to-be* 67
- user experience* 82
- visualization* 292

Lazy loading pattern 333

Lead to cash 54–55

Legacy application integration 148

Legal entity 108

Legal ground 108

Load balancing 330

Locate Object app 95

Location master data 88

Logging 307, 340

- activate* 308
- bind service* 308
- dashboard* 310
- monitoring* 312

M

Machine learning 155, 164, 346

Machine learning models 143

Maintain Business Users app 265

Maintain Employees app 265

Manage Business Partner app 91

Manage Business Purposes app 109

Manage Distribution Model app 94–95

Manage Object Type Owners app 95

Manage Orders app 100

Mapping guidelines 135

Master data distribution 85

- strategy* 88
- use cases* 85

Master data management 80

Master data orchestration 93–94, 100

Master shadow instance 331

Maximum CPU consumption 312

Mckinsey’s three horizons 30

Mean time to failure (MTTF) 326

Mean time to recovery (MTTR) 326

Measuring success 363

Message implementation guidelines 135

Message orchestration 114

Message processing 144

Message Queuing Telemetry Transport (MQTT) 138

Metadata 142

Metadata cache 332

Microsoft Azure 271

Mobile app catalog 184

Mobile card kit 186

- app creation* 187
- creation* 186
- gatekeeper settings* 187

Mobile cards 182

Mobile consumption 177

Mobile development kit 183, 189

- deploy app* 192
- editor* 192
- feature selection* 190
- native/hybrid application definition* 189
- new project* 191

Mobile integration 116

Mobile transaction bridge 183

Model-driven architecture (MDA) 66

Monitor Master Data Orchestration app 95

MTA descriptor file 310

mta.yaml file 217

mtar files 284, 286, 288

Multicloud architecture 363

Multicloud environment 33

Multi-data center failover 340–341

Multilingual text repository (MLTR) 199

Multitarget applications (MTAs) 284, 368

- archive* 284, 287, 299
- archive builder* 287
- build application* 286
- create* 284

N

Natural language processing (NLP) 159

Neo 33, 102, 311

- audit logging* 231
- authorizations* 218
- custom platform roles* 225
- Dynatrace* 313
- logging* 312
- SAP Cloud Platform Keystore* 221
- subaccount roles* 224

NIST’s cloud computing reference architecture 70

nodejs10 258

nodejs8 258

NoSQL databases 83

O

OAuth 2.0 222

- flows* 222

OAuth access token 230

OAuth authentication 204

OData 231, 253, 332

- frameworks* 193

Open Connectors 132, 150

- authenticated connections* 134
- components* 133
- home page* 133
- prerequisites* 133

OpenID Connect 207

OR logic 315

Organizations 23

OSBAPI-enabled platform 149

P

PACE layering application strategy 236

pipeline_config.yaml 295

Platform advisory methodology 68

Platform capability model 69

- foundation* 72
- future aware* 74
- future-focused* 77

Platform security 222

Platform-as-a-service (PaaS) 15

Ports 119

Prioritization matrix 372

Process analysis 52

Process discovery 53

Process excellence methodology 47

Process explorer 51

Process integration 111

Process intelligence 73

Process invocation 113

Process mining 47, 50

Process modeling 73

Process performance indicators (PPIs) 54

Process transparency 73

Project 294–295, 298, 368

Provider interface 96

Publisher/subscriber patterns 329

Push notifications 184

Q

Quality index 200

Qualtrics 155, 170, 351

Queue subscriptions 137

Queues 136, 141

Quota plans 283

Quota policy 334

Quotas 283

R

Rate-limiting 334

Redis on SAP Cloud Platform 333

Redundancy 330

Reference dates 108

Regions 21

Remediation 315

Replatforming 365, 368

Resilience4j 337

Resiliency patterns 326

Resiliency principles 324

Resource provider 270

REST API 220

Retail industry 354

Return on investment (ROI) 370

RFC protocols 118

Role collections 90, 125, 130, 140, 179, 210, 214, 290, 319

add roles 126

assign role 215

assign to users 216

create 215

map to user groups 215

Roles 90, 125, 210

assign to groups 219

create 213

custom platform roles 225

define 218

inside subaccount 214

manage 283

map groups 219

predefined 222

templates 93, 99, 106, 210, 213, 319

Ruum by SAP 56–57

canvas 57

process automation 59

process self-service 59

templates 57

S

SAML 2.0 identity provider 204–205, 208, 215–216

SAML 2.0 metadata 205

SAML attributes 211

SAML service provider 206

SAP AI Business Services 104, 156–157

SAP API Business Hub 54, 128–129, 147

SAP Ariba Cloud Integration Gateway 97

SAP Business Workflow 59

SAP Cloud Application Programming

 Model 248, 294, 368

development tools 249

project explorer 252

project template 251

SAP Cloud for Customer 97

SAP Cloud Identity Access

 Governance 229

SAP Cloud Identity Services 205, 227

SAP Cloud Platform 31

application deployment 338

automation 154

benefits 32

intelligent enterprise 34

pricing 37

regions 33

role 32

runtime decision matrix 268

runtimes 242

service catalog 38

user types 203

SAP Cloud Platform Alert Notification 315

Cloud Foundry 316

Neo 316

setup 316

subscriptions 317

SAP Cloud Platform Application Runtime ... 38, 242, 268

SAP Cloud Platform Automation Pilot 317

catalogs 320

home screen 319

SAP Cloud Platform Alert Notification 320

setup 319

SAP Cloud Platform Business Application

 Development Tool add-on 284

SAP Cloud Platform Business Application

 Studio 181, 183, 189, 191, 250, 279

dev space 191

project 284

role templates 249

SAP Cloud Platform Business Rules 229

SAP Cloud Platform cockpit 96, 160, 167, 213, 265, 308

SAP Cloud Platform Connectivity 117, 146, 331

SAP Cloud Platform Continuous Integration and Delivery 301, 341

access 302

configuration 301

SAP Cloud Platform Continuous Integration and Delivery (Cont.)

home screen 303

new job 303

SAP Web IDE 305

SAP Cloud Platform Credential Store 220

setup 221

SAP Cloud Platform Data Enrichment ... 87, 98

activate 99

apps 100

service instance 101

SAP Cloud Platform Data Retention

 Manager 105

apps 109

configure 106

home page 107

SAP Cloud Platform Destination 117, 266

SAP Cloud Platform Digital Process

 Automation 73

SAP Cloud Platform Enterprise

 Messaging 112, 136, 254–255

configuration 139

home screen 140

instance creation 139

message clients 141

overview 136

queue 260

roles 140

standard libraries 138

SAP Cloud Platform Extension Suite 37, 76, 140, 146

SAP Cloud Platform Feature Flags 279, 338

dashboard 339

setup 339

SAP Cloud Platform Git service 279

SAP Cloud Platform Identity

 Authentication 226

SAP Cloud Platform Integration Suite 36, 121, 354, 363, 366

API Management 128

capabilities 122

Cloud Integration 126

configuration 122

home page 124

Integration Advisor 134

Open Connectors 132

provisioning 123, 132

SAP Cloud Platform Keystore 221

SAP Cloud Platform Launchpad 116, 178

Cloud Foundry 178

roles 179

site manager 180

SAP Cloud Platform Master Data for business

 partners 89, 105

apps 91

authorizations 89

orchestration 93

service instance 92

setup 89

SAP Cloud Platform Master Data

 Integration 87

SAP Cloud Platform Mobile Services .. 182, 199

client 189

cockpit 185

features 184

overview 182

setup 185

SAP Cloud Platform Process Visibility 155, 165, 167

activate 166

integration 168

new instance 167

open APIs 171

product attributes 170

Qualtrics 170

service plan 168

SAP Cloud Platform SDK for Android 196

components 196

new project 198

setup 197

SAP Cloud Platform SDK for iOS 183, 193

manage accounts 194

new project 195

setup 194

Xcode 195

SAP Cloud Platform Serverless Runtime ... 253, 268

setup 257

structure 259

SAP Cloud Platform Transport

 Management 279, 283, 341

activate 289

configure 291

home screen 290

integrate with pipeline 299

roles 289

scenarios 293

setup 288

transport content archives 293

SAP Cloud Platform Workflow

 Management 165, 229

activate 166

integration 168

SAP Cloud Platform, ABAP environment ...	261, 268
<i>business role templates</i>	264
<i>Eclipse</i>	265
<i>launchpad</i>	264
<i>setup</i>	262
SAP Cloud Platform, Kyma runtime ...	266–268
<i>CPEA-based licensing agreement</i>	267
<i>home screen</i>	268
SAP Cloud Platform, serverless runtime	146
SAP Cloud SDK	294, 296, 334, 337
<i>caching</i>	332
<i>fallbacks</i>	338
<i>Jenkins server</i>	297
<i>resilience library</i>	337
SAP Conversational AI	154, 158
SAP Customer Experience	113
SAP Data Intelligence	112, 142, 351, 353
<i>capabilities</i>	142
<i>launchpad</i>	143
<i>setup</i>	143
SAP Data Quality Management,	
microservices for location data	88, 101
<i>configuration</i>	103
SAP Digital Vehicle Hub	369
SAP Edge Services	349
SAP Enterprise Architecture Framework	68
SAP Fiori	193
SAP Fiori launchpad	187, 264
SAP HANA Cloud	115
SAP ID service	204, 216
SAP Integration Solution Advisory	
Methodology	112
<i>domains</i>	113
SAP Intelligent Business Process	
Management	154–155, 165
<i>templates</i>	167
SAP Intelligent Robotic Process	
Automation	56, 155, 159
<i>Cloud Factory</i>	160–161
<i>components</i>	157
<i>Desktop Agent</i>	163
<i>Desktop Studio</i>	162–163
<i>execution modes</i>	163
<i>functional content</i>	172
<i>integration</i>	173
<i>predelivered content</i>	164
<i>SAP-delivered scenarios</i>	157
SAP Internet of Things	112, 144, 349, 353
<i>cockpit</i>	144
<i>setup</i>	145
<i>tenant home</i>	145
SAP Master Data Governance	86
SAP Mobile Cards app	183, 186
<i>emulator</i>	188
SAP PowerDesigner	66
SAP Process Mining	50
<i>analysis sheet</i>	52
<i>data scope</i>	52
<i>process discovery</i>	53
SAP Process Orchestration (SAP PO)	124
SAP S/4HANA	39, 138
<i>customization</i>	16
<i>master data</i>	86
SAP S/4HANA Cloud	99, 138, 140, 238
<i>APIs</i>	242
<i>authentication</i>	247
<i>extensibility</i>	240, 242, 254
SAP Single Sign-On	209
SAP SuccessFactors	115
SAP Translation Hub	198–199
<i>integration</i>	200
<i>Neo environment</i>	199
<i>translation project</i>	199
SAP Web IDE	166–167, 183, 189, 279, 284, 286, 288, 305, 368
<i>plug-in</i>	305
SAP Work Zone	180
<i>overview</i>	181
<i>user personas</i>	181
SAPML framework	193
Scaling policy	336
SCIM-based system	227
Scopes	211
<i>assign</i>	217
SDK Assistant	193
Security	90, 99, 134, 203
<i>Cloud Foundry</i>	204
<i>roles</i>	213
Service broker	271–272
Service fields	103
Service instance	92
Service key	145
Service management	149
<i>log in</i>	149
Service Management Control (SMCTL)	
command-line tool	149
Service SDKs	249
Service Ticket Intelligence	156
Service-level agreement (SLA)	170, 324–326
<i>calculate</i>	326
Service-oriented architecture (SOA)	114
Side-by-side extensibility	236
Siloed data	78

Source to pay	54
Spaces	23
<i>create</i>	251
<i>quote plans</i>	283
<i>roles</i>	224
Spike arrest policy	335
Spotlight by SAP	54
Staged development	280
Static attributes	211
Subaccounts	21, 281
<i>groups</i>	282
<i>roles</i>	223
<i>tiering</i>	280
Subscription-based model	38
Subscriptions	315
System availability	326
System harmonization	366

T

Technical reference model (TRM)	68
Technology architecture	70, 74, 76
Technology platform principles	81
Telco industry	351
Tenants	318
TOGAF	67
Topics	137
Trailblazers	77
Transient fault handling	327
Transport destinations	291
Transport Landscape wizard	292
Transport routes	291
Travel and transportation	349
Travel to reimburse	55
Trial systems	37

U

Unattended RPA	158, 164
Unrestricted attributes	211
User Authentication and Authorization	
(UAA)	204, 207, 211
User consumption	115
User experience (UX)	177
<i>principles</i>	82
User groups	215
<i>create</i>	219
User interfaces	115
User-provided service instances	148
Utilities industry	352

V

Value realization	370
Vertical scaling	335

W

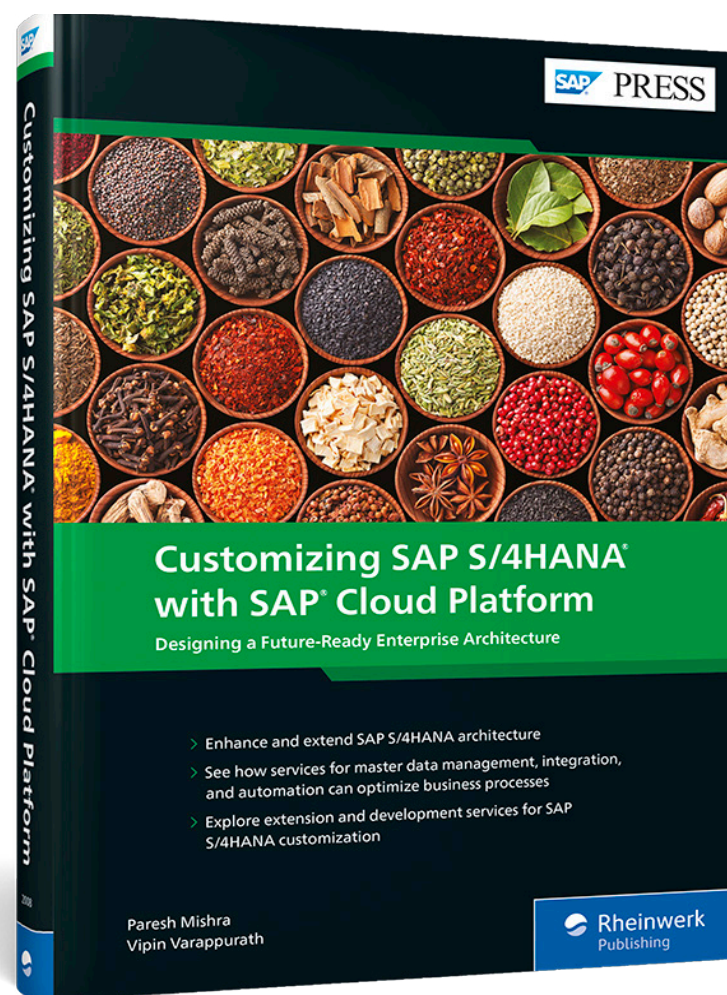
Webhooks	141
WebSocket	137
Whitespaces	48, 354
Workflow Editor	166–167

X

Xcode IDE	183, 193, 195
-----------------	---------------

Z

Zero downtime deployment	338
--------------------------------	-----



Paresh Mishra, Vipin Varappurath

Customizing SAP S/4HANA with SAP Cloud Platform: Designing a Future-Ready Enterprise Architecture

383 Pages, 2021, \$79.95

ISBN 978-1-4932-2008-3

 www.sap-press.com/5157



Paresh Mishra is the vice president of the SAP Cloud Platform customer success team for the Asia Pacific and Greater China (APAC) regions. He is responsible for driving customer adoption and supporting customer success during all phases of SAP Cloud Platform implementation (adoption, extension, and optimization).

His team helps customers to leverage SAP Cloud Platform for digital transformation by building new applications and extending standard applications to better meet business needs.



Vipin Varappurath is the director of product success for SAP Cloud Platform in the SAP P&T Customer Success organization, helping to drive product success for SAP customers on SAP Cloud Platform, focused on strategic engagements. In his previous role, he was the head of the Digital Customer Success Team for EMEA

and MEE and responsible for driving customer adoption for large customer segments in different phases of project implementation. In this role, Vipin helped companies derive value from their investments in SAP Cloud Platform during their transformation journey. Vipin also worked with large SAP ecosystems to help scale SAP Cloud Platform to a larger SAP customer base.

We hope you have enjoyed this reading sample. You may recommend or pass it on to others, but only in its entirety, including all pages. This reading sample and all its parts are protected by copyright law. All usage and exploitation rights are reserved by the author and the publisher.