

Handbook for Developers and Consultants on SAP Build - Part 1

Deepak Kumar

Wilmington, USA

Email: [deepak3830\[at\]gmail.com](mailto:deepak3830[at]gmail.com)

Abstract: SAP Build offers a range of user-advanced tools to speed up app development and automation using smart AI technology. It lets users quickly build, enhance and automate apps, with drag-and-drop functions and AI-powered coding support. The platform makes it easy to expand applications by providing to-use connectors and business content for systems, not just SAP. Moreover, SAP Build promotes teamwork between business and development teams through integrated governance and lifecycle management features. Its main highlights include creating apps with both drag-and-drop methods and advanced coding techniques using AI assistance for coding tasks extending SAP and third-party apps rapidly and ensuring access, to SAP app data through pre-built authentication methods.

Keywords: Low-code and no-code (LCNC), Citizen Development, SAP Build

1. Introduction

Low-code and no-code (LCNC) refer to a new style of visual programming that enables application development without traditional programming. This approach significantly lowers the barrier to entry for anyone interested in developing applications.

Low-code combines traditional programming environments with no-code platforms, catering to developers with at least basic technical knowledge. It offers a hybrid approach, utilizing both coding and visual development tools.

No-code, on the other hand, replaces traditional programming languages with visual development tools, such as drag-and-drop components. This straightforward approach is accessible to both technical and non-technical individuals, including citizen developers.

Citizen development addresses the current gap in technical expertise, particularly the shortage of experienced professional developers, by empowering non-technical users to create applications.

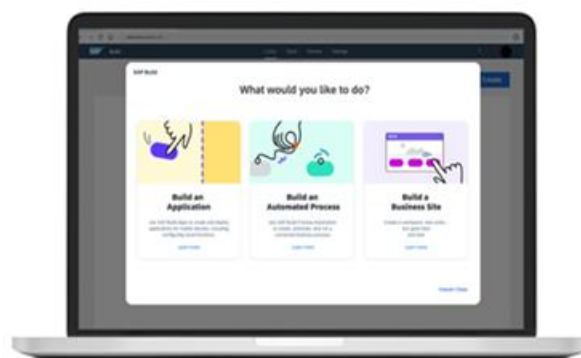
SAP Build's Low-Code portfolio merges no-code and low-code solutions, enabling users of all skill levels to easily create and enhance business applications. It allows business experts to build with drag-and-drop simplicity, integrate with SAP and non-SAP systems, and facilitate collaboration between business and development teams.

2. SAP build offerings

SAP Build products are divided into three categories: SAP Build Apps, SAP Build Process Automation and SAP Build Work Zone.

SAP Build Apps is a user app builder that enables business professionals to create high-quality applications, for desktop, web, and mobile platforms without needing to code. By using a drag and drop interface and made modules users can easily build flexible applications. This tool empowers technical users to develop front-end apps with customized features,

independent apps, and add-ons for enterprise software like SAP S/4HANA or SAP SuccessFactors solutions. Deploying these designed applications is just as simple, as creating them – they can be launched with a few clicks.



SAP Build Process Automation enables business users to adapt, improve, and innovate business processes. Utilizing visual features that require no coding expertise, even non-technical workers can complete standard development tasks more quickly, enhancing business agility. By integrating SAP Workflow Management, SAP Intelligent Robotic Process Automation, and embedded AI capabilities, this tool scales process automation to meet evolving business needs. It offers native integration with other SAP offerings, pre-built process content including RPA bots for SAP applications, and AI-assisted intelligent document processing. Built on an enterprise-grade platform, SAP Build Process Automation fosters innovation with remarkable speed, simplicity, and confidence.

SAP Build Work Zone empowers IT professionals and business users to easily build and design engaging business sites for employees, partners, and customers. It enhances user productivity by providing central access to relevant applications, processes, and information on both desktop and mobile devices.

Volume 12 Issue 3, March 2023

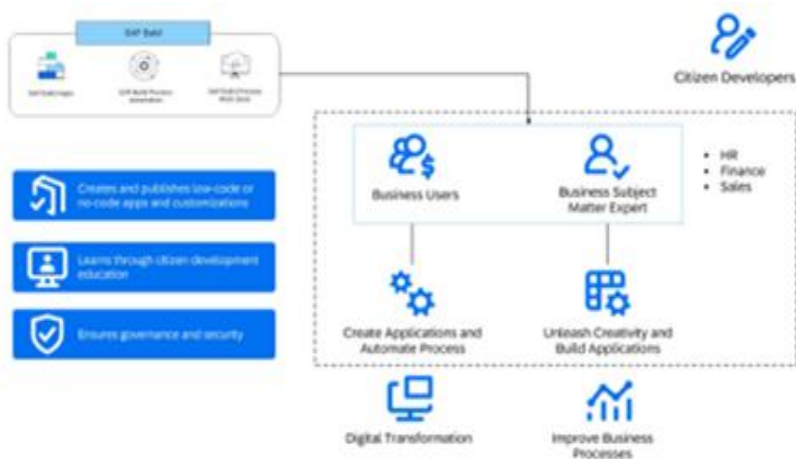
www.ijsr.net

Licensed Under Creative Commons Attribution CC BY

3. Advantages of SAP Build

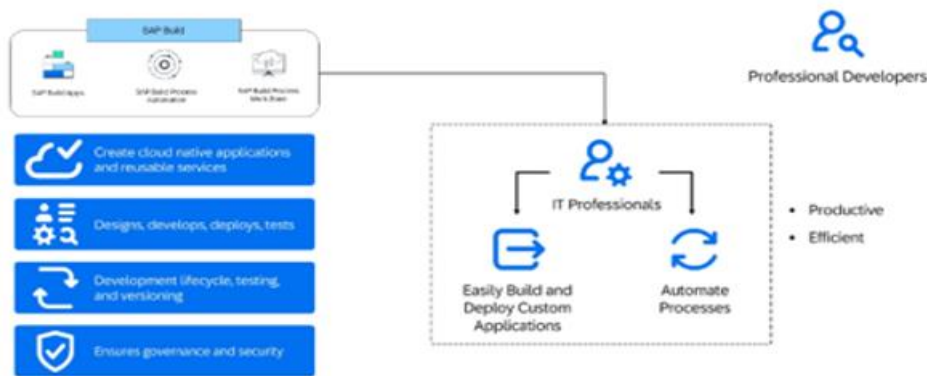
Both everyday users and experienced developers can find value in using SAP Build. Everyday Users; SAP Build enables individuals, in business areas like resources, finance, and sales to develop applications and streamline tasks without requiring a technical background. This platform empowers

them to express their ideas by crafting custom applications that cater to their requirements fostering digital advancement and enhancing operational efficiencies from, within the business realm.



Citizen Developers: Citizen developers create and publish low-code and no-code apps and customizations closely linked to business contexts. They might not initially be aware of aspects like testing, going live, versioning, and termination but can learn these through citizen development education. Governance and security are ensured through IT-sanctioned tools and platform-supported processes.

Professional Developers: SAP Build allows IT professionals to build and deploy custom applications or automate processes more easily, without requiring deep expertise in specific programming languages or frameworks. This enhances their productivity and efficiency, enabling them to focus on more complex and value-adding tasks.

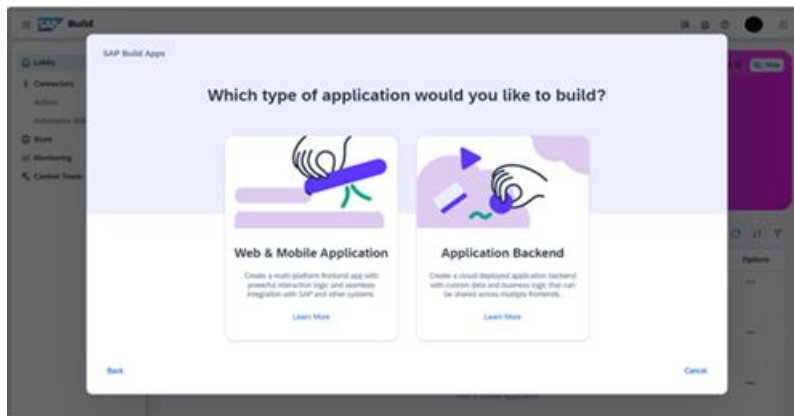


Skilled developers utilize their tools to craft native applications and reusable services. They play a role in every stage of the process from conceptualizing and coding to deploying, testing, and transitioning to production. They possess a grasp of the development life cycle encompassing testing and version control while prioritizing governance and security at each step.

allows the creation of apps for both web and native mobile use. Building on the foundation of SAP AppGyver, SAP Build Apps now includes backend capabilities. This enables business users to confidently build enterprise-grade apps that work seamlessly across all operating systems without writing a single line of code. Professional developers can reduce coding effort in creating complex data models and business logic, significantly boosting productivity. Available in the Cloud Foundry environment, SAP Build Apps supports the development of two types of projects: applications and backends, which are databases for storing application data.

4. SAP BUILD APPS

SAP Build Apps empowers users to develop full-stack applications visually. As a low-code development platform, it



Features

Build intricate user interfaces: Create web and mobile applications using a drag-and-drop approach, leveraging a comprehensive library of customizable user interface components, logic flows, and data transformation formulas. Collaborate with your team members in real-time and preview changes instantly.

Persist data and run application logic: Develop no-code application backends with built-in cloud functions capabilities. Design solutions that can operate independently or integrate seamlessly with your existing SAP data models.

Integrate with SAP and other systems: Connect your app to your no-code backend effortlessly with just one click. Integrate modern APIs in minutes, or access data from the cloud or SAP systems using SAP BTP Authentication and Destinations.

How to approach SAP Build Apps?:

Starting with some checks;

Make sure that SAP Build Apps aligns with your security needs. Check if you have the access to data for your app using SAP Build Apps and assess if any extra services are required to integrate this data. Validate that the features you need are present in SAP Build Apps. Get a grasp of your needs by inquiring; Who will be using your app? What tasks do they aim to achieve through the app? What tools are they currently using for these tasks? What steps do they need to take to complete these tasks?

Defining these requirements will aid in comprehending the extent and purpose of your application ensuring it effectively caters to user requirements.

Principles you can follow for good development:

Usefulness: Determining the effectiveness of the app, in addressing user needs is crucial. An app that fails to solve a problem might not be well received. It's best to start with a product (MVP) that focuses on solving a core issue rather than trying to tackle multiple problems at once. This helps keep the app simple and user friendly preventing it from becoming too complex and difficult to use. Additional features can always be added based on user feedback and changing requirements.

Usability: When assessing usability consider whether the app simplifies tasks compared to existing methods. If the app isn't more intuitive or easier to use than solutions users may prefer sticking with what they know. A designed app should be easy to navigate and use allowing users to complete tasks efficiently. If you're not familiar with usability design consider working with a designer or consultant who can help create a user experience. Implement error prevention measures and clear communication, for when errors occur.

Maintainability: For maintainability make sure the app is structured and documented in a way that other team members can understand and update it easily when needed.

Follow guidelines, like DRY (Don't Repeat Yourself) to ensure consistency and make maintenance easier. Having a defined strategy for organizing and naming app features is crucial, for maintenance and long term viability.

Practices of good development:

Variables, Application State, and Scope: SAP Build Apps offers various types of variables, each suited to different purposes. These include data variables, theme variables, translation variables, system variables, page parameters, and internal component variables. Understanding their specific uses will enhance both your development experience and the overall maintainability of your application. Page variables are limited to the page on which they are defined, while app variables are accessible across all pages within the application. To keep your application efficient, always limit a variable's scope to the minimum necessary. If your application starts accumulating over 20 or 30 app variables, reconsider your approach. Group related information in Object or List-type variables to simplify management.

Navigation: Effective navigation between pages is crucial for user experience in applications with multiple pages. Each instance of a page opened via the "Open page" flow function creates a separate instance of the page and its data variables. These variables are reset when the page is removed from the navigation stack, either by navigating back or replacing it with a different page. Properly managing navigation and variable scope ensures a smoother user experience.

Logic: The logic of your application dictates how it responds to user interactions and operates internally. Place logic in relevant and easily accessible locations, adhere to the DRY

(Don't Repeat Yourself) principle by using events, and implement robust error handling. For logic that needs to run continuously or periodically, consider placing it on the Global canvas unless it is specific to a single page.

Error Handling: Integrating comprehensive error handling within your application helps users understand potential issues and prevents data inconsistencies due to improper inputs or data removal. Anticipate possible error states during development to safeguard both user experience and application data.

Debugging: When your application's logic does not perform as expected, debugging helps identify and resolve issues. Use your browser's developer tools to troubleshoot—the Network tab is useful for issues with backend requests, while the Console tab provides logs generated by your application. Producing your own logs can also aid in diagnosing problems.

User Interface (UI): The UI encompasses everything users see and interact with. Effective UI design directly impacts the user experience. A foundational understanding of UX, UI design principles, or design thinking will greatly benefit your development process.

General Usability: Consider two main aspects while designing your UI: communicate the current state of the application and prevent user errors. Indicate loading processes or ongoing tasks with visual cues like spinners or

progress bars. Utilize components such as the Progress Bar to track progress in multi-step processes.

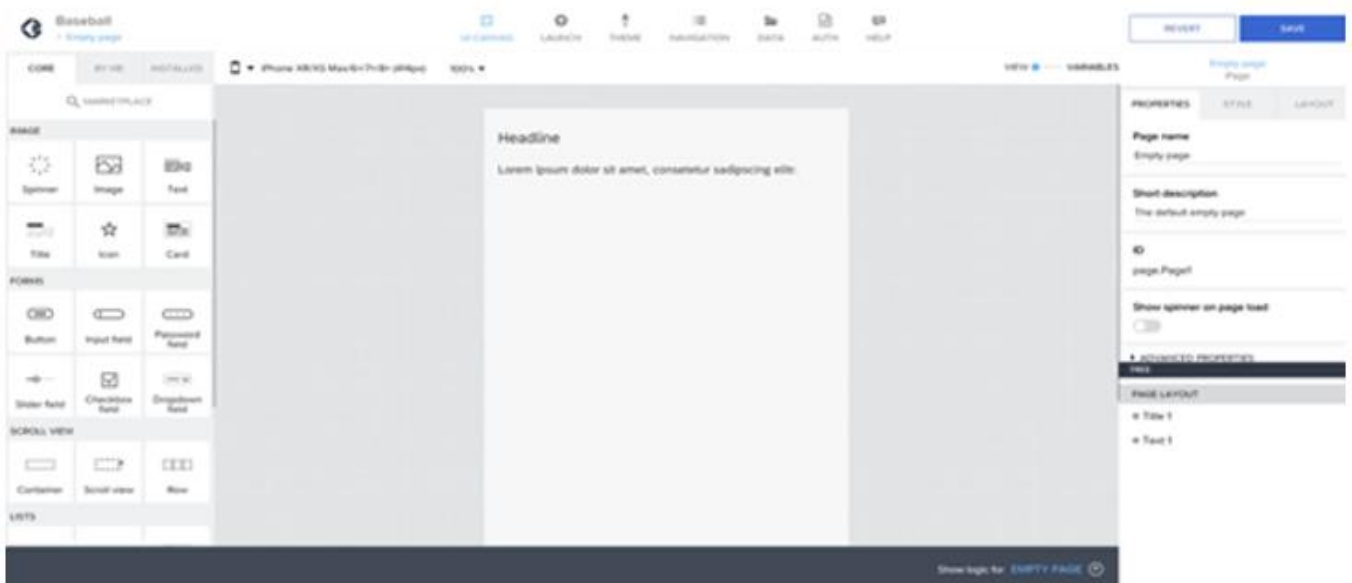
Theming: For consistency and maintainability in your app's user interface, use theme variables and style classes. Instead of applying static values, base your style changes on theme variables. This approach allows for easier adjustments later by modifying theme variables in one place, rather than updating each instance of a static value.

A look into tools of SAP build apps:

The Logic Panel: functions, as a canvas where you can organize and link widgets to establish a sequence of actions. These widgets referred to as flow functions allow you to structure the order of events and operations in your application. Begin by placing a triggering event on the panel then connect it to flow functions to construct a series of tasks.

User Interface (UI) Component Tools; Located on the left-hand side of the screen the components section grants access to all UI elements that can be incorporated into your application. This section is segmented into three tabs;

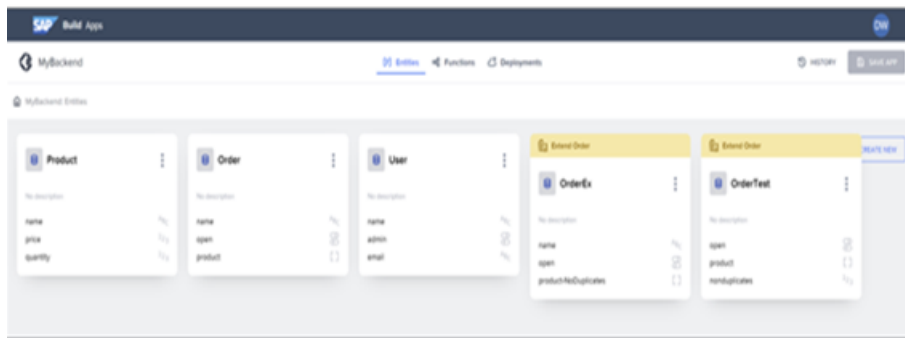
- 1) Core; Contains 25 components like buttons and input fields crucial for basic app design.
- 2) Marketplace; Provides an array of components for enhancing your app design options.
- 3) Installed; Shows all components you have added including those, from both the Core and Marketplace tabs.



The SAP Build Apps backend lets you create and manage your data sources by setting up data tables called entities and filling them with information. This feature allows you to utilize data, in your applications of depending solely on external sources.

The user interface is divided into three sections, which are indicated by the tabs at the top of the page;

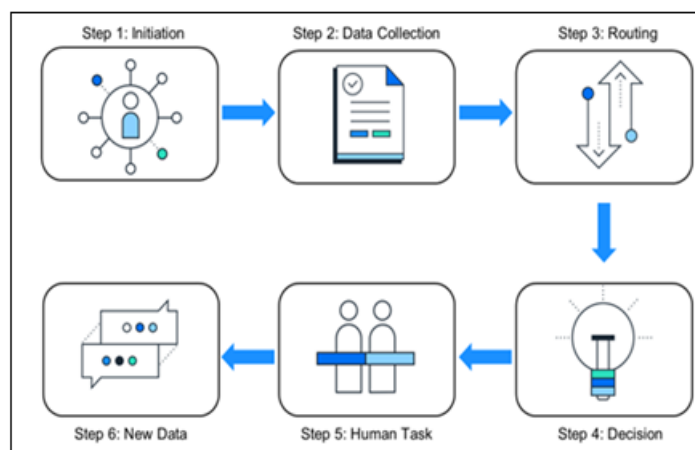
- 1) Entities; This section displays database tables that hold data.
- 2) Functions; In this area you define operations that can process and manipulate the data. Functions carry out calculations. Produce subsets of data based on parameters provided by the application.
- 3) Deployments; After configuring entities and functions you must deploy them. Once deployed these resources are ready for use, in your applications.



5. SAP Build Process Automation

Before embarking on building no-code automated solutions, it's crucial to have a solid understanding of business processes. SAP Build Process Automation empowers both business users and technologists to adapt, enhance, and innovate processes using no-code workflow management and robotic process automation tools. As a citizen developer

solution, it leverages intuitive low-code and no-code capabilities, enabling users to drive automation effectively. SAP Build Process Automation represents an evolution from the former SAP Workflow Management and SAP Intelligent Robotic Process Automation products, offering a unified citizen developer experience that these earlier products lacked. Key stages in a business process typically include initiation, data collection, routing, decision-making, human tasks, and data generation.



6. Features

The solution provides a range of key features designed to enhance process management and automation:

- **Intuitive Graphical Interface:** Build or adapt processes using an easy-to-navigate graphical interface.
- **Forms-Based Workflows:** Create workflows with drag-and-drop functionality for streamlined form-based processes.
- **Decision Logic Management:** Develop and manage decision logic through tabular, spreadsheet-like decision tables.
- **Robotic Process Automation:** Automate repetitive tasks within existing workflows using robotic process automation.
- **Machine Learning Capabilities:** Create intelligent actions and recommendations powered by machine learning.
- **Unified Launchpad and Task Center:** Work efficiently from a centralized platform that integrates all tasks and processes.
- **Collaboration with Professional Developers:** Hand over projects to professional developers for embedding advanced workflows and actions initiated by citizen developers.

- **Real-Time Process Visibility:** Access event-driven transparency and comprehensive insights into processes and instances through process visibility dashboards.
- **Prebuilt Content:** Leverage prebuilt components, including bots, process steps, business rules, and workflow elements, to accelerate development and deployment.

These features collectively enhance the ability to design, automate, and manage business processes effectively.

Benefits with respect to requirements:

When you use SAP Workflow Management you get access, to features that are meant to simplify and improve workflow processes. This platform offers a range of management tools in a single environment, which means you don't have to worry about moving your data around. It allows you to reuse elements like workflow components, process steps and business rules all of which can be smoothly integrated into your existing workflows using the SAP Business Application Studio. The systems approach makes it easier for users of all levels to develop workflows with coding required. Moreover SAP Workflow Management provides a user experience for citizen developers by offering built in automation and process creation tools.

Similarly SAP Intelligent RPA provides automation capabilities within the environment without requiring any migration efforts. It comes with elements such as applications, data sets and bots that can seamlessly blend in with your bot setups. The platform includes up-to-date process content and an expanding library of features and skills templates. If you're looking to automate end-to-end business processes SAP Intelligent RPA supports citizen developers by providing resources like forms, business rules for decision making, and process monitoring tools along, with a growing collection of automation resources and skills templates.

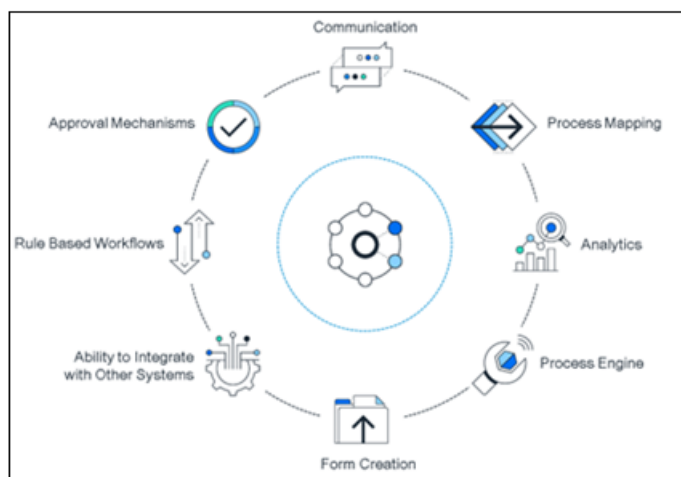
For beginners interested in exploring no code automation processes through a user experience SAP Build offers an opportunity tailored for citizen developers.

Need for Business Process Management:

When overseeing operations, in a company pinpointing problem areas can pose a challenge. It's essential to have business process management in place to streamline and

execute processes efficiently. Business Process Management (BPM) software plays a role in organizing the multitude of business processes within a company by defining and managing them to prevent potential chaos. Structured business processes can improve agility, speed, accuracy and productivity leading to increased employee success and satisfaction while also making work more meaningful. Furthermore it enhances visibility boosts company efficiency and profitability and promotes communication and accountability among employees.

Managed processes enhance communication, among individuals and departments providing benefits in terms of accuracy, speed and efficiency. This not helps eliminate confusion and frustration but also enhances team accountability. By addressing process related issues team members can take on responsibility focus on their objectives and avoid attributing delays or failures solely to the process itself.



Business Process Management (BPM) comes with benefits, at stages of the business process cycle. It offers insight into what's happening within the organization and more control over business activities. By pinpointing and resolving obstacles or hindrances BPM can cut down on time and expenses. Enhanced communication and teamwork, across the company allow employees to focus on tasks while clear and well documented processes boost flexibility. Ultimately BPM supports businesses in striving for improved efficiency and effectiveness.

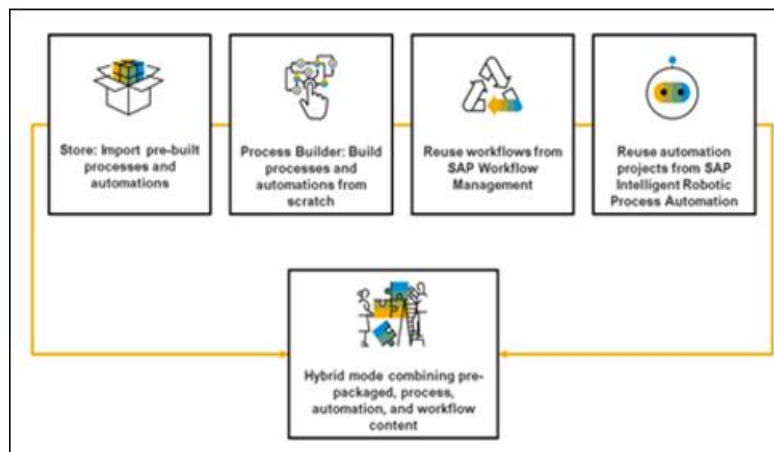
Key Roles in a Business Process

In a business process, various roles play crucial parts in ensuring smooth operations. The Process Owner holds overall responsibility from a business perspective, making necessary adjustments when issues arise. The Process Initiator begins the process through specific actions, whether intentional or not. The Decision Maker influences the process direction by making critical decisions, with their approvals or rejections

affecting the process outcomes. Process Participants are responsible for completing tasks, which may involve physical activities or operations within software systems. Meanwhile, the Process Instance Reviewer and Process Metrics Reviewer focus on evaluating specific process instances or overall metrics, aligning their interests with their job roles. These roles can be grouped into three categories: Primary Participants, Secondary Participants, and Stakeholders.

SAP Build Process Automation Working Modes

Import prebuilt processes and automation from the built-in Store. You can make use of a wide variety of prebuilt, directly usable packages that are made available via the built-in Store. The packages span across lines of business, industries, and shared functions. The package content ranges from business processes to template automation, from form templates to process visibility projects, as well as actions, also called skills.



Developers, whether experienced or new, to the field find creating custom processes and automations using the Process Builder tool essential. This tool offers a user environment for designing and executing business scenarios without the need for coding. It enables the creation of forms, approval processes, automations (bots) business rules (decisions) actions (skills) controls and workflows within SAP Business Application Studio.

For those who already have workflows in place SAP Build Process Automation allows integration of these workflows from SAP Workflow Management into processes. Users can also create workflows in SAP Business Application Studio. Deploy them to the SAP Build Process Automation runtime.

Moreover existing automation projects developed with SAP Intelligent Robotic Process Automation can be repurposed in SAP Build Process Automation. By exporting automation projects from the system and importing them into the platform users can continue leveraging their previous work.

SAP Build Process Automation offers flexibility by supporting an approach where prebuilt content can be combined with custom workflows and processes. This versatility enables users to enhance existing workflows, from SAP Workflow Management by incorporating steps or components created within SAP Build Process Automation.

7. Conclusion

This guide is a resource, for developers and consultants who want to make the most of SAP Builds tools for transformation in their organizations. By exploring the offerings within SAP Build, such as SAP Build Apps and SAP Build Process Automation users can unlock the platforms capabilities to create, customize and automate enterprise applications efficiently.

SAP Build brings benefits, including development cycles improved collaboration between business and IT teams and seamless integration with both SAP and non SAP systems. Its code and no code features empower developers of all levels to build and deploy high quality applications and processes.

SAP Build Apps, an advancement from SAP AppGyver offers an environment for building full stack applications using an interface. With drag and drop UI elements, backend

setup without coding and seamless integration with SAP systems developers can ensure their apps are user friendly and sustainable through development practices.

Incorporating management with robotic process automation capabilities SAP Build Process Automation simplifies business processes. Designed for citizen developers in mind to enable business users to optimize workflows while providing tools for enhancing innovation, in business practices.

In a business process it's important to have roles defined so that everyone knows what they need to do and can contribute effectively.

Using SAP Builds approach and wide range of features helps companies improve efficiency, flexibility and creativity. By following practices and making use of SAP Builds built in tools and capabilities developers and consultants can develop solutions that make an impact, on their businesses changing needs. This guide offers the knowledge and practical advice needed to start a journey with SAP Build empowering teams to reach their goals, for digital transformation.

Declarations

Ethics approval and consent to participate: Not Applicable

Consent for publication: All authors have consent to submit this paper to the Journal of Cloud Computing. Also, we confirm that this paper or any part of this paper was not submitted anywhere.

Availability of data and materials: Not Applicable

Competing interests: Not Applicable

Funding: Not Applicable

References

- [1] "Low-Code App Development & Automation Solutions | SAP Build," SAP. <https://www.sap.com/products/technology-platform/low-code.html>
- [2] "Compose and Automate with SAP Build the No-Code Way | SAP Learning," learning.sap.com.

- <https://learning.sap.com/learning-journeys/compose-and-automate-with-sap-build-the-no-code-way>
- [3] “SAP Build Apps | AppGyver | Visual Low-Code App Builder,” SAP. https://sap.com/products/technology-platform/low-code-app-builder.html?url_id=text-glo-404-reclink
- [4] “SAP Business Accelerator Hub,” hub.sap.com. <https://hub.sap.com/build>
- [5] “Business Process Automation Software | Low Code | SAP,” www.sap.com. <https://www.sap.com/products/technology-platform/process-automation.html>
- [6] “SAP Build Work Zone | Digital Workplace Experience,” SAP. <https://www.sap.com/products/technology-platform/workzone.html>
- [7] “SAP Help Portal,” help.sap.com. <https://help.sap.com/docs/build-apps/service-guide/what-is-sap-build-apps>
- [8] “SAP Help Portal,” help.sap.com. <https://help.sap.com/docs/build-apps/service-guide/right-way-to-sap-build-apps>
- [9] “Creating Processes and Automations with SAP Build Process Automation | SAP Learning,” learning.sap.com. <https://learning.sap.com/learning-journeys/create-processes-and-automations-with-sap-build-process-automation>