Forms and Workflow
Buying Guide

Get Started

kuali®

Table of Contents

3 Introduction

6 How to Begin Searching for a Forms and Workflow Solution

11 What to Look for in a No-Code Forms and Workflow Solution

11 The Foundation

12 The Form Builder

13 The Workflow Engine

14 The Reporting

14 The Integrations

15 Identity Management

15 The Technology Platform

16 The Pricing Model

17 What to Look for in a Vendor Partnership

18 Three Next Steps to Take

19 About Kuali Build

20 Sources

In the wake of a global pandemic and dramatic pressures to do more with less, higher education institutions are seeking ways to innovate, reduce costs, and improve student success outcomes. In April 2020, The American Council on Education (ACE) estimated an enrollment drop of 15% percent, including a 25% percent decline in international students. These statistics are forcing hard evaluations of current institutional business practices.

For years, digital transformation initiatives have been infiltrating campus classrooms, libraries, and systems. As of 2019, Educause reports[1] that over 80% of institutions are currently utilizing or exploring a transformation strategy. While institutions have historically taken a hyper-focused approach to continuous improvement, recent economic pressures and future uncertainties are forcing rapid optimization.

In 2020, Educause surveys on COVID-19 unveiled that right now is a unique opportunity to accelerate transformation. Half of the survey respondents reported focusing on Digital Transformation as a way to reduce institutional costs.[2] Respondents reported significant increases in institution-wide openness to transformation initiatives:

> "The move to digital has been monumental. We have achieved more change on these fronts over the last six weeks than we have over the last two years. It has enabled people to change, and the outcome is a new path forward."

To meet the agility level demanded by the unforeseen events of 2020 and the years to come, institutions must think beyond IT as the central powerhouse of application enablement.

Institutions are becoming creative with automation-focused technology to adapt to increasingly demanding challenges, such as:

Flinders University in Australia **saved 1,800 annual hours** by utilizing Robotic Process Automation to **automate updates of over 17,000 student records.**[3]

The University of Oregon utilized a portfolio of solutions, including Sharepoint and PowerBI, to create a COVID-19 **contact screening** and **tracing app.**[4]

Davidson College utilized no-code tools Kuali Build and SnapLogic to **automate daily COVID-19 screening and results reporting**, enabling the campus to return in-person for Fall 2020.[5]

While transformation strategies can take many forms, automating core business processes is critical in optimizing institutional software environments. Many institutions are beginning to explore low-code forms and workflow solutions as part of their shift to embrace digital transformation.

Low-code or no-code platforms support staff, faculty, and campus-wide administrators leading the charge in their automation efforts. These citizen developers[6] can develop and launch form and workflow applications to solve critical business needs on their own. Gartner predicts that by 2024, low-code application development will be responsible for more than 65% of application development activity.[7] To meet the agility level demanded by the unforeseen events of 2020 and the years to come, institutions must think beyond IT as the central powerhouse of application enablement. Instead, consider how to empower the democratization of technology access and use across the institution. Through focusing intently on developing digital literacy and open access to powerful tools like low- and no-code platforms, institutions can transform to achieve faster operational efficiency and focus on increasing enrollment and surpassing student expectations.

The benefits of implementing low-code automation platforms are insurmountable:

1. More efficiency and transparency

Workflow automation technology gives users the tools and ability to eliminate manual data entry, increasing efficiency across departments and the institution as a whole. In addition, tools offer informative reporting capabilities, helping institutions gain insight into process status and optimization opportunities.

2. Increased agility

Institutions will be equipped to respond to market demands and changes, such as quickly moving to remote learning and working.

3. Better data security

Workflow automation technology offers much more protection than traditional paper and PDF processes. With controlled access and data encryption, private information is kept safe.

This guide will walk you through what to consider in your search for no-code forms and workflow solutions if your institution is ready to join the ranks and launch your digital transformation initiative.

How to Begin Searching for a Forms and Workflow Solution

Gartner reported that by 2024, the average enterprise institution will have over four low-code development platforms.[8] This approach to tool adoption is an absolute path to consider in an institution's search for low-code automation, as needs may vary widely across different departments.

With this in mind, there are five core steps when searching for an automation platform. These steps will help institution leaders define the scope of their process optimization opportunity, illustrating if a single or multiple solutions should be considered.

Step #1 CLARIFY THE PROBLEM YOU'RE TRYING TO SOLVE.

Institutions that have successfully embraced digital transformation initiatives, such as North Carolina-based Davidson College, agree that taking a user-centered approach to scope identification of IT projects is vital.

Document the identified problem as you know it. Next, connect with subject matter experts across campus to both validate and clarify the problem scope. Specifically, identify:

1. **Does the process touch multiple people across campus? If so, who are they?**
2. **Does the process interact with information held in core systems (such as the Student Information System)?**
3. **Is the process static or relatively dynamic?**
4. **What type of information is the process about, and are there security implications with this information?**
5. **What kind of archival requirements (if any) do we have about this type of information?**

If there are multiple processes or problems in question, try to find the lowest hanging fruit first, such as a single process or a group of processes associated with a team. If you are struggling to determine viability, consider employing an impact-effort matrix,[9] a tool associated with Lean Six Sigma process improvement methods.

Step #2

OUTLINE WHAT IS HOLDING YOU BACK FROM THE SOLUTION TODAY

Identifying why this problem has yet to be solved can help you arrive at a meaningful solution. There are often three core categories that contribute to a lack of progress, and an issue can fall into multiple categories:

People

People may have limited bandwidth, causing manual process execution to be a considerable strain on their current role.

Alternatively, a solution could have been introduced before but not widely adopted due to a flawed roll-out, a complicated platform, or insufficient executive buy-in.

Understanding how people, particularly the subject-matter experts, expect and respond to proposed solutions is critical in identifying the problem scope.

Systems

It is common for institutions to have several tools already that support some version of process automation. Unfortunately, PDFs and paper still exist rampantly on campuses today. Why?

Often, subject-matter experts lack the necessary access or technical knowledge to utilize existing tools on campus. Many legacy systems are highly customized and specialized, making it difficult for the typical department or business user to operate them with ease. A plan should be centered around user design, enabling anyone on campus to use it while also being technically advanced enough to meet IT security, accessibility, and compliance requirements.

Process

A technological solution applied over a broken process is a band-aid that will not heal a root cause.

Processes can be broken through poor documentation, unnecessary steps, bureaucratic approval hierarchy, or too much variability depending upon minute factors.

Identifying process issues and opportunities for improvement is a crucial step in determining if now is the right time for a technological solution and what may need to happen before a successful implementation.

Step #3 UNDERSTAND THE TYPES OF TOOLS IN THE MARKETPLACE

Once you've identified the scope of the problem you're trying to solve and what may have prevented successful solutions previously, you can now explore the types of automation solutions in the marketplace today.

When beginning your exploration into low-code platforms, you can think of tools as being based on a continuum of user's technical experience.

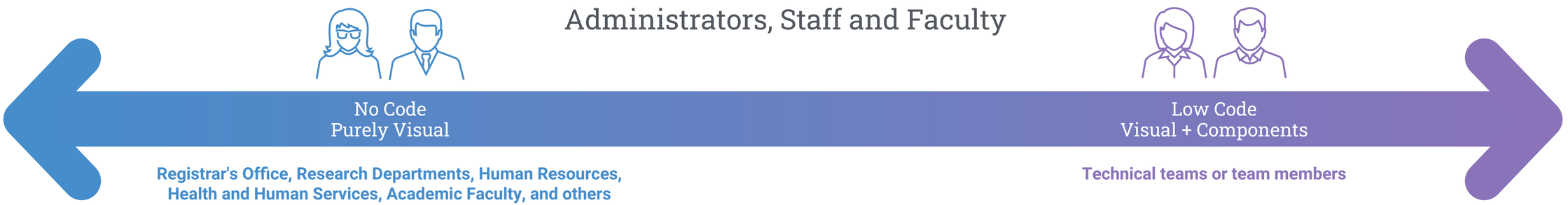
One of the ends of the spectrum supports business users with no technical experience, such as many staff, faculty or administrators within an institution. On the other side, technical users, either within a technical team or a functional department are supported.

On the less technical spectrum, no-code tools enable staff, faculty, and non-technical administrators to act as "builders," creating business applications such as a change of major request, parking permit form, or a vacation request process. These tools typically include visual, drag and drop elements with little to no coding required.

No-code platforms are ideal for either standalone or integrated evolving processes with shorter life-cycles of months to years. These platforms empower institutions to move faster, quickly responding to changing market demands, as anyone across campus can quickly build and launch apps.

Alternatively, on the more technical side of the spectrum, low-code tools use declarative, visual modeling tools to lessen the amount of coding needed to build complex, enterprise-grade apps.

Full low-code platforms are for integrated, mission-critical business processes with an extended life-cycle of over five years. Low-code is ideal for creating long-lasting, stable applications requiring advanced, integrated frameworks, often requiring some level of advanced technical expertise to build and maintain.

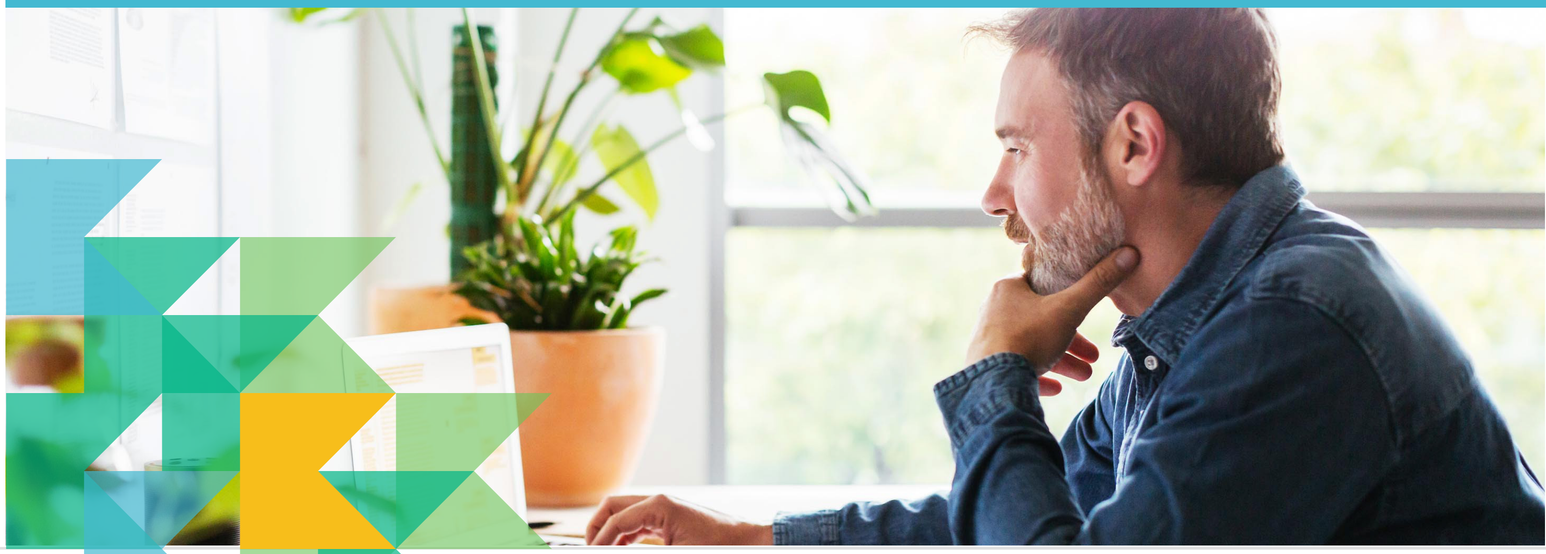


So how do you choose what you need?

First, focus on the scope of the problem from steps #1 and #2. How complex is the process you're trying to solve? Does it need to be rolled out quickly? Do subject matter experts have access to technical resources to build the solution? If not, do they have the bandwidth to create one themselves? Answering these questions will help you identify which side of the spectrum to focus on when acquiring a tool for your department or institution.

Next, be critical of tools on the market and realize the terms no-code or low-code are fundamentally marketing driven. Develop a robust set of requirements to solve the scope of your problem and while you focus on a type of toolset, gauge the vendor(s) against your requirements. You'll quickly find not all tools, despite their categorization, are made alike.

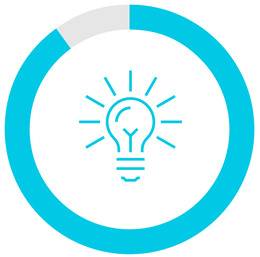
If you are truly aiming to transform your institution's reliance on paper and PDF-based processes, you must look for a no-code solution that focuses on ease-of-use and powerful capabilities. Subject matter experts across your campus know their pain. They understand the root of their process challenges more than anyone. They must be empowered with tools quickly to solve their most important process challenge with little-to-no reliance on technical support. While low-code tools serve a purpose in empowering technical teams to move faster, they will not solve the fundamental challenge of making your campus move faster if the majority of the process challenges fall outside of IT.



Step #4 DETERMINE IF YOU SHOULD ACQUIRE A SINGLE TOOL OR TAKE A PORTFOLIO APPROACH.

Case study:

In Davidson College's recent digital transformation initiative, they sought a tool to support removing PDFs and paper across campus. After in-depth user interviews and analysis, they identified three core use cases to solve, ranging from simple form submissions, to integrated forms and complex enterprise-grade applications. Within their interviews, they found 90% of use cases centered around simple to advanced form submissions, with only 10% focusing on complex-grade applications. With this in mind, their digital transformation team elected to take a portfolio approach to acquire applications, choosing Kuali Build to solve 90% of their use cases and low-code platform Mendix to solve 10%.[10]



Kuali Build solved

90%

of their use cases and low-code platform Mendix solved 10%.

When determining whether to take a single tool or portfolio approach, consider some of the following questions:

1. How complex is the problem you're solving? Can it be addressed through a single tool?
2. What tools in your portfolio today can help you solve your problem? Why aren't they working?
3. What is the bandwidth for implementing solutions, and do you need technical expertise to do so?
4. How much spend can you allocate to a solution?

Step #5 BEGIN EXPLORING SOLUTIONS

Once you've identified your target approach to either an individual tool or portfolio solution, you may begin to create a requirements framework for your search.

If electing to search for a no-code platform to help solve your institution's process automation needs, the following section will help you identify core solution requirements.

What to Look for in a No-Code Forms and Workflow Solution:

When seeking a no-code forms and workflow automation solution built for higher education, there are several core elements you should consider to ensure the tool meets and scales with your needs.

1

THE FOUNDATION

Easy (and Fun) to Use: The platform should be built with user-centric design in mind, free from technical jargon, and quickly understandable, so even the staff member who drags their feet with every change will be excited to jump in.

Expert tip: Software people enjoy using leads to more usage and therefore, more return-on-investment. Software that people hate using ends up being used as little as possible and reduces overall value.

Powerful: No-code apps can range dramatically in terms of power, with some only creating primary forms while others empower advanced data connections, automated workflows, and integrations. Look for a tool that delivers the necessary ability to scale with your needs.

Mobile-Friendly: Not all mobile is created equal! A tool should empower mobile-readiness with no additional requirements on behalf of the form submitter. Look for a tool with mobile-ready forms out of the box (not one that requires an app to be mobile-friendly).

Accessible: Compliance is king in higher education. With this in mind, seek a WCAG 2.0 AA accessible tool, ensuring equal access and opportunity to all who need to build an application or submit a form.

Brandable: The platform should be able to easily mirror the look and feel of your institution. Look for a tool where branding is central throughout forms and custom notifications to ensure a seamless user experience.

Expert tip: Initiate a trial version or subscribe to a sandbox. Have a non-technical team member try it out to gauge how user-friendly it is.

2

THE FORM BUILDER

Drag-and-Drop Builder: Users should easily drag and drop key form components onto an easy-to-design layout, complete with sections, pagination, and customizable WYSIWYG text.

Simple and Complex Field Types: Forms should be complete with a broad array of field types, ensuring maximum flexibility in form creation. Field options should range from a simple text input or multiple-choice options to more complex types such as table gadgets and data connection look-up fields.

Expert tip: A system should have just enough fields to add a relevant form of information without being over-bloated with field types. Look for a system where each field serves a unique, practical purpose as it will showcase the platform's focus on user-centric design.

Progressive Disclosure: Forms should only show relevant fields based upon what a user submits as field responses, ensuring that a form is as simple as necessary. Look for a tool that allows conditional visibility of form fields based upon logic in data input.

Advanced Field Logic: A form with advanced logic such as field validation, option limitations, character requirements, help or placeholder text, and custom field names. Advanced settings help to ensure data accuracy and form usability.

Data Look-Ups: Form fields should easily connect to core campus systems, such as your student information, curriculum management, or research administration system, to automatically populate fields with accurate information. For example, when you input a student name, the form should automatically pull the student ID number, GPA, major, and email address from the student information system onto your form.

3

THE WORKFLOW ENGINE

Drag-and-Drop Process Map: The workflow component should be included out of the box and should empower staff and faculty to quickly map out a process flow using drag and drop functionality. Workflow engines should be complete with only the necessary activities to support workflow routes such as approval or acknowledgment tasks, custom notifications, and updating relevant integrations.

Powerful Branching: The platform should support both conditional branching, or modifying routing conditions based upon decisions in the workflow or form, and parallel branching, or running unique workflow routes in tandem. As an example, the system should support a different set of approvals if “History” is selected as a student’s major. If approval from a department chair is denied, the system should support executing specific steps accordingly.

Expert tip: Complex parallel branching is a critical component in differentiating a simple versus advanced no-code solution. Ensure parallel branching can support complex logic criteria and test various scenarios in a sandbox or trial environment to validate the tool will scale to meet future complex process needs.

Role-Based Routing Logic: Approval routes should honor your institution’s hierarchical complexity by using Role-Based Routing Logic. For example, you should be able to route to the dean of a selected department, so every approval will follow a similar logic. The system should also have group-based logic, so if a committee is approving a decision, the system knows how many committee members need to approve. Finally, the system should enable custom notifications so builders can send personalized emails to approvers, submitters, and more, informing them of the form status or their task.

Easy-to-Use Integrations: Staff and faculty should be able to quickly drag and drop integration requests to facilitate a core system update, such as updating the student information system with a student’s newly approved major.

Streamlined Testing: Quality-assurance testing is often one of the most labor-intensive portions of application design and launch. The solution should facilitate an easy-to-use testing simulator, empowering non-technical staff and faculty to quickly test all complex workflow components and identify and resolve issues prior to launch.

4

THE REPORTING

Customizable Reports: The platform should enable builders and administrators to create customized, filtered, and savable reports showcasing both submitted data and relevant associated information, such as that proposed by an individual or submitted date.

Advanced Metadata: Reporting should include advanced information with each form submission to add context. For example, the reporting should include how long a form has been waiting on a particular step, as well as the submission's current status.

Expert tip: If data look-ups are available, the form metadata should be able to pull additional information stored in the database about each integrated field, such as identifying a student's total credits taken even if it was not included on the form submission.

Export and Integration Capabilities: You should be able to export form data and process it further using a tool such as Microsoft Excel. Alternatively, the system should pass submission information into a core business intelligence tool, such as Tableau, enabling central storage and advanced reporting and analytics.

THE INTEGRATIONS

5

Open Online APIs: The platform should be built on the latest cloud-based API technology, providing fully open and documented access so that, if desired, technical teams can write customizable integrations with any third-party system.

Expert tip: In addition to an open API, finding a vendor with several pre-built integration options provides a great foundation to get started with little technical support required.

Database Look-Ups: Verify whether the tool can integrate with critical campus systems, such as your student information system, and easily look up relevant form submission data with no additional coding required.

Flexible Submission Data: Form submission data should be readable, writable and searchable via a rich API, enabling administrators to push core campus systems such as the enterprise content management system.

6

IDENTITY MANAGEMENT

Integrated Directory Management: The platform should integrate natively with your institution's single-sign-on capabilities, ensuring users only need to remember one login and manage one set of credentials. Connect the platform with your directory services solution to quickly register all respective departments, colleges, roles, and users within an institution. The system should scale natively to support a large number of users.

Flexible Role-Based Hierarchy Management: Custom system roles adapt to your institution's unique hierarchy structure, providing long-term flexibility for matching identity management to ever-changing roles and departments on campus.

Shared Governance: The platform supports granular permission sets, so while IT can maintain control over permissions, users, integrations, and specific applications, department administrators and application administrators can control access to their particular group's forms, users, and custom roles.

7

THE TECHNOLOGY PLATFORM

Cloud-Native: Scalable tools are built from scratch on a cloud-native framework, often hosted on a leading industry provider such as Azure or Amazon Web Services, using the latest cloud-first technology. A software-as-a-service solution ensures no technical overhead is required and it can still scale to meet the demands of processes and users.

Security Foundation: Ensure the platform encrypts data at all times, both at rest and in transit. Seek a vendor who maintains industry-leading uptime standards of at least 99.9% with a publicly accessible uptime page and service-backed service level agreement (SLA). Request and validate approaches to multi-location data redundancy, data breach handling, and privacy policies.

Expert tip: Involve your IT department and your IT security team in advance of looking to purchase the solution. Understand what information IT will need and work closely with the vendor to get all questions answered, ensuring procurement will not be held up once you've selected your platform(s) of choice.

8

THE PRICING MODEL

Simple: Complicated pricing is commonplace in the low-code and no-code market. It shouldn't have to be. Find a vendor that believes in a simple pricing model and stands by that commitment. Extreme feature segmentation, usage-based models, or excessive add-ons over-complicate the buying process. Find a pricing model that allows you to easily expand your value, either in users or usage, with no additional significant costs incurred.

Predictable: Budget approvals are becoming increasingly difficult, especially in the current climate. Mid-cycle approvals without a pre-approved budget can be nearly impossible. Look for a solution with a pricing model that does not change throughout the year, or does not have overages risks associated with many usage-based models.

Expert tip: If a pricing model is usage-based, ask for all details regarding overages before the purchase. Overages can cause extreme overspend in some cases. Unpaid overages may result in service disruption, which could be catastrophic for a mission-critical application.

Adoption-Focused: User-based pricing models are indicative of a legacy solution based on transitioning an on-premises software model into cloud-hosted technology. Be wary of fully user-based models, especially if the long-term goal is to empower champions across your campus to build automated solutions. If a system is user-based, inquire how granular the pricing functions are, such as individual users or user blocks. Also, validate how user-based pricing impacts submission capabilities for both known and anonymous users, and which users can build forms. Look for a vendor with unlimited users to focus on spreading adoption widely.

What to Look for in a Vendor Partnership

Higher Education Experience

Seek out a vendor that has an established, long-standing higher education community. The vendor partnership should be rooted with institutional knowledge and expertise that understands the unique challenges and opportunities presented by working within higher education. The vendor should seek to work with you to further your mission, not just provide another software platform.

Vision and Strategy

Entering into a vendor relationship is, in essence, developing a long-term partnership. Understand the purpose of the vendor's mission as an organization and where they see themselves in two to five years. Ask questions about the future product roadmap to ensure the direction aligns with your vision as you scale. Inquire about how product suggestions are addressed, to ensure your feedback will be not only listened to but can influence future product strategy.

Number of Years in Business

A vendor should have years of proven experience with a long history of success. Confirm the partner has been financially viable historically, with a long runway of success.

Size of Higher Education Customer Base

Inquire as to what percentage of overall customers are in higher education and, specifically, the makeup of those institutions, such as large research schools or small liberal arts colleges. Validate that the vendor has proven success with institutions like yours.

Customer References

Ask for relevant customer references and case studies to confirm that your expectations for the product rollout, use case, and overall vendor partnership will not only be met, but exceeded. Finally, look online for customer references to find unbiased external sources. Seek out higher education-specific reviews to ensure feedback is relevant.

Three Next Steps to Take

As you continue your search for a forms and workflow automation platform, here are three steps we recommend taking:

1. Prepare your system or portfolio requirements

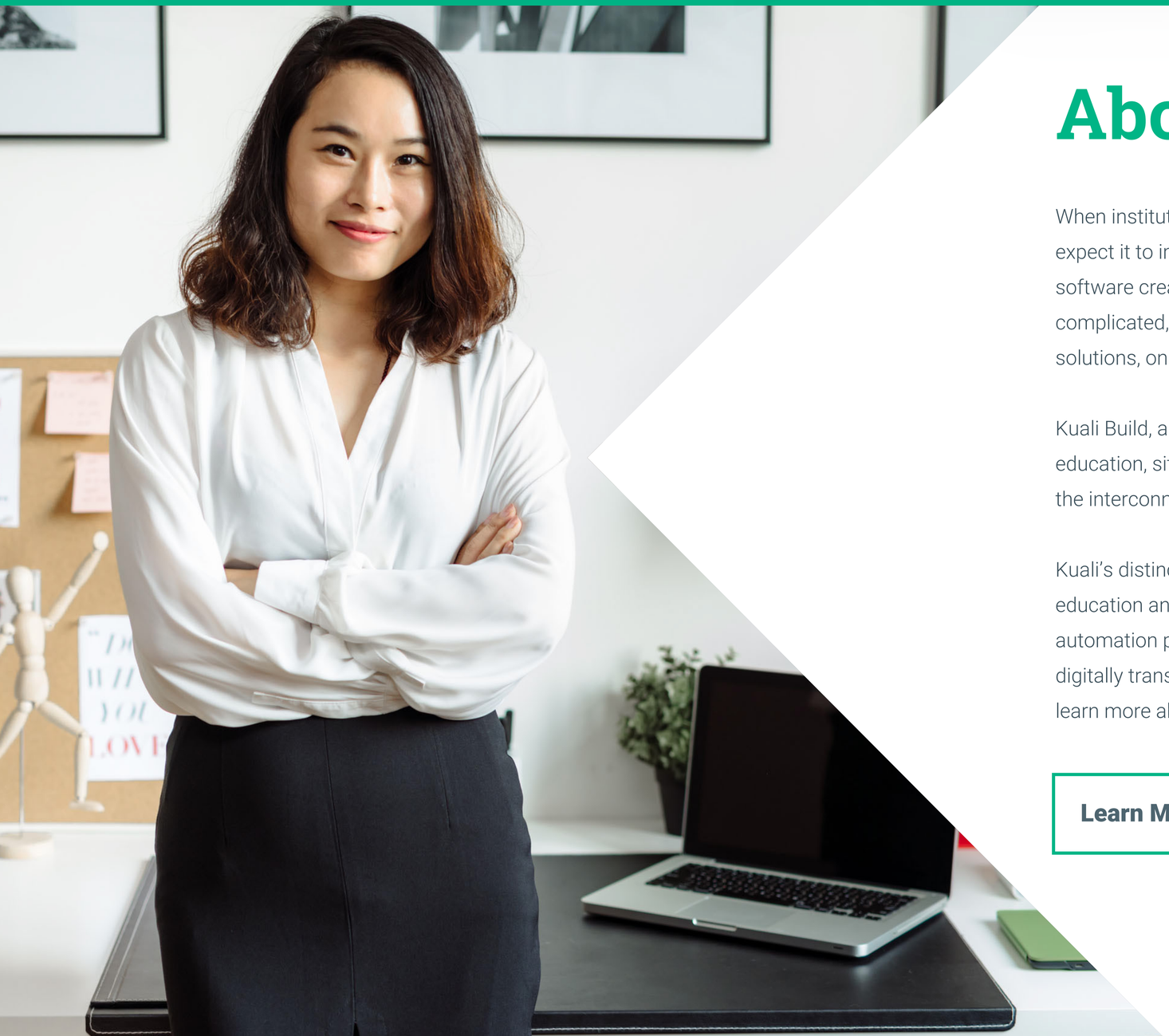
Use the information in this guide to create a requirement matrix. Validate who will be a member of the buying committee to simplify the buying process. Validate this matrix with subject-matter experts initially interviewed in your approach. Involve IT to document all security and technology requirements. Create a grading matrix to create a fair, detailed method of reviewing prospective platforms and vendors.

2. Ask colleagues, analysts, or procurement managers for recommendations

Begin your search by asking for advice from your core community. Outline your requirements and inquire if they can recommend familiar tools. Educause community groups provide an excellent forum for posting system inquiries. In addition, if your institution has a Gartner or other analyst agreement, consider scheduling time with them to ask for their recommendation. Following initial investigations, search online, follow industry publications or look on review sites for suggestions.

3. Schedule initial vendor discussions

Create a list of five to 10 initial vendors to explore. Begin by researching vendor sites. After this, schedule exploratory meetings with vendors that meet the overview of your initial requirements. If you're comfortable, request a demo or sandbox to begin exploring system capabilities.



About Kuali Build

When institution decision-makers invest in software to drive digital transformation initiatives, they expect it to improve research processes and student success. In many cases, however, the software creates more work and distracts from what matters. Robust solutions tend to be complicated, time-consuming, and require engineers, adding to the overall expense. Simple solutions, on the other hand, often have compliance issues and lack necessary security measures.

Kuali Build, a no-code forms and workflow automation platform built exclusively for higher education, sits in the sweet spot between straightforward and complex campus systems, acting as the interconnected glue.

Kuali's distinct community partnership approach, built over 15 years of collaboration with higher education and a focus on user experience, has resulted in an easy-to-use yet powerful and secure automation platform adopted by business users in every department on campus. If you're ready to digitally transform your institution, we're here to help you take the next steps. Visit **build.kuali.co** to learn more about our software and how we can help your institution achieve its mission.

Learn More About Kuali Build

Sources

1. Driving Digital Transformation in Higher Education | EDUCAUSE
2. EDUCAUSE COVID-19 QuickPoll Results: IT Budgets, 2020–2021
3. RPA case study in higher education
4. Emerging Technologies for Tracking COVID-19 in Higher Ed
5. Davidson College uses Kuali Build for COVID-19 Symptom Tracking and Reporting
6. The Pros and Cons of Citizen Development - Forbes
7. Application Development Research Predicts Low-Code Tooling Takeover
8. Application Development Research Predicts Low-Code Tooling Takeover
9. What is an impact effort matrix?
10. Davidson College Creates Meaningful Change with Launch of Digital Transformation Team