appian

Selecting the Right Appian Delivery Partner

Selecting the Right Appian Delivery Partner

Selecting the right delivery partner is just as important as selecting the best low-code platform. Appian is proud of our growing global partner community. There are a wide array of highly qualified Appian delivery partners to choose from. However, finding a partner that matches your organization's needs is very important. Industry and systems familiarity, security requirements, as well as general Appian experiences and skills, among other factors, can all be key determining factors. This guide will help you to select the right Appian delivery partner for your organization.

Whatever the project, aligning the right services, qualifications, and ensuring the partner follows best practices helps set the stage for success. Key factors to consider include:

- Does the partner have a clear understanding of my solution requirements and an effective plan for delivery? Successful projects begin with a detailed plan. As part of the contract development process, make sure you have provided sufficient detail on functional and nonfunctional requirements, but in return, make sure the partner has clearly articulated a plan for each phase of project delivery. More detail on ensuring partner readiness through effective contracting can be found in Appendix A.
- Does the partner have the necessary Appian team composition (technical and managerial skills) to be successful? When evaluating team resources, organizations should assess the number of years of direct Appian platform and customer domain experience. Always review team resumes and request partner team Appian Certifications to ensure partners can provide qualified personnel. For more information on recommended team roles and responsibilities for Appian projects, please refer to Appendix B.
- Does the partner's delivery plan adhere to Appian delivery best practices? Select delivery partners who understand the best practices for delivering Appian-based solutions. Assess the vendor's approach to integrating and applying best practices within their solution. Additional best practices can be found in the Appian Success Center on Community and in Appendix C.
- Does the partner team have relevant industry/domain expertise and a proven track record of Appian delivery success for similar solutions? Where possible, ask to speak with references within your organization's industry or domain. Make sure that not just the partner, but the specific individuals on the project team have relevant industry/domain experience. Past performances should also demonstrate not only Appian experience, but also experience with any additional in-scope technologies.

Appian Customer Success Support for Projects

Appian Customer Success offers a range of implementation and advisory services that deliver expert guidance and support for all aspects of Appian platform strategy, architecture, design, and delivery. By working closely with both you and your chosen delivery partner, Appian's Success Services are proven to increase delivery velocity and quality, while reducing implementation risks. Appian's services can be integrated together and scaled up and down to support stand-alone implementation projects or provide sustained support for large digital transformation programs. For more information, contact your Appian account representative or reach out to the Appian Customer Success team directly.

Appendix A - Contracting Guidance.

This appendix provides organizations with guidance on submitting effective RFPs to delivery partners for Appian projects or programs. It is intended to assist in providing the delivery partner with clear scoping statements and standardized task, personnel, and deliverable requirements for agile-based development on the Appian platform. Organizations should tailor and adapt this guidance to the specific vision for each Appian implementation contract and adjust on a project-by-project basis.

Scope.

Within the scope section, organizations should provide prospective delivery partners with detailed descriptions of the scope of services and the desired solution's functional and non-functional capabilities and requirements. These descriptions should be written to the level of detail that facilitates accurate estimation for the selected contracting approach. Organizations should also clearly outline the bounds of project scope to avoid confusion around project deliverables and scope creep. The following subsections outline the scope suggestions.

Services.

Below is a list of services you would expect to see from a partner when proposing to manage your project. This full list may not apply in all project scenarios.

- Agile Project Management
- Stakeholder Management
- Requirements Management
- Configuration and Development
- Data Management
- Architecture, Infrastructure Support, and System Administration
- Test Planning and Execution (e.g. System, Regression, Integration)
- Performance Testing
- Automated Test Script Development
- User Acceptance Testing Planning and Execution Support
- Training Development and Delivery
- IT Security Testing, Security Documentation, and Accreditation
- Release Management and Configuration Management
- Operations and Maintenance, including Service Desk
- Contract Transition

Functional requirements.

The following factors will help accurately estimate the scale and scope of an Appian solution. It enables organizations and partners to quickly begin "Sprint 0" where project goals and scope is confirmed, and initial sprints are planned.

- High-level description statements of system functionality, in support of the objectives specified in the prior sections. *Note that Appian does not recommend using detailed requirements documents (eg. BRUF), but instead high-level objectives which support an agile requirements process.*
- Supported business processes, including providing business process flow diagrams
- System Personas, including high-level descriptions of how the personas will use the application and the number of anticipated users by persona.
- User Interfaces, Appian Sites, Reports, and Generated Documents, including high-level descriptions, estimated or expected quantities and complexities.
- Integrations, including descriptions, methods (e.g. file, e-mail, messaging, database connection, etc.), data volumes, interface flows, data specifications, and formats.
- Document Management, including the types and formats of stored documents and the anticipated document sizes and volumes.
- Application Security, including specialized security or privacy requirements, security providers/single sign-on requirements.
- Advanced Application Features, including high-level use cases or desired capabilities of Artificial Intelligence/ Machine Learning, Robotic Process Automation, etc.
- Applicable laws, regulations, or standards that must be adhered to.

Non-functional requirements.

Similar to the Functional Requirements, the Non-Functional Requirements will help formulate an accurate proposal and cost estimate for the in-scope solution.

- Identify technical environments for pre-production and production. State the partner's responsibility for infrastructure/network engineering, configuration documentation, hardware, network, and software installation support and/or administration
- Continuous Integration/Continuous Deployment (CI/ CD) pipeline requirements, including automated testing, automated deployment, software version control, etc.
- Technical standards, enterprise architecture standards, and governance processes and review requirements
- Performance requirements, including peak concurrent users, total user counts, application response times, anticipated data volumes etc.
- User experience standards and user interface requirements.
- System Service Level Agreement (SLAs) requirements.
- Data Migration requirements, including identifying data sources, underlying technologies, and anticipated data volumes; identifying responsibilities and approaches for data cleansing and conversion testing; specifying whether parallel operations are likely required.

Appendix B - Team Requirements

Project delivery team staffing and structure should align to the scope, scale, and complexity of the planned Appian solution, as well as with organization-specific processes and requirements. While the specific program and/or project designs may vary, organizations should ask for personnel with the right qualifications, training, and experience to fill defined roles that efficiently deliver high quality outcomes.

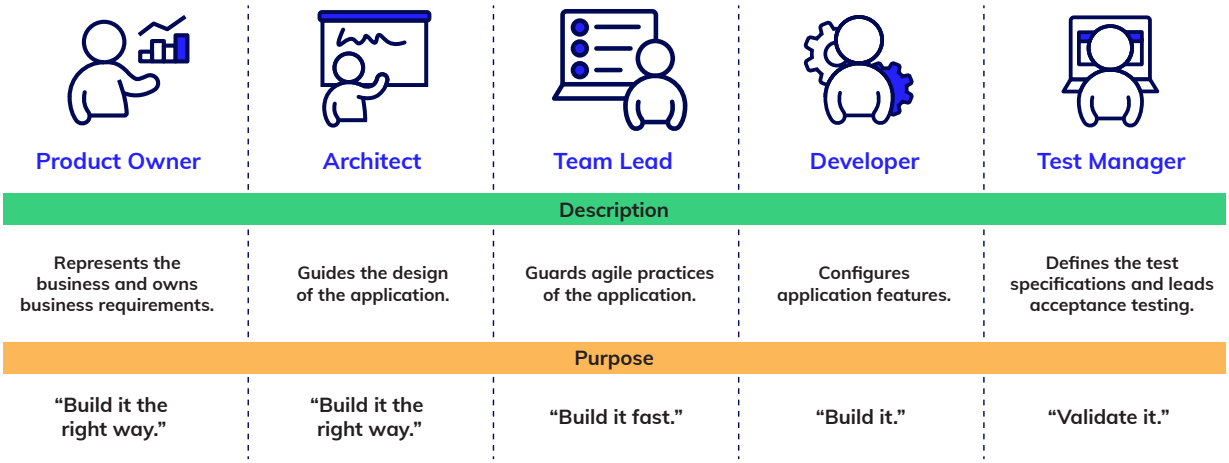
Table 1: Program Team and Certifications

Role	Qualifications
Team/Delivery Manager	Preferred Qualifications • Appian Certified Analyst (or higher) • Recommended 18+ months of active Appian project experience • Agile project management certification (eg. SAFe Certification, PgMP and/or PMP)
Architect	Preferred Qualifications • Appian Certified Lead Developer • Recommended 3-4 years of active Appian project experience • Certified Scrum Master or equivalent certification
Lead Developer	Preferred Qualifications • Appian Certified Lead Developer • Recommended 2-3 years of active Appian project experience • Certified Scrum Master or equivalent certification
Low Code Developers	Preferred Qualifications • Appian Certified Senior Developer • Junior team members- Appian Certified Associate Developer
Analysts and/or Testers	Preferred Qualifications • Appian Certified Analyst • Note: These roles can be performed by Appian developers rather than specialists
UI/UX Engineers *(optional)*	Preferred Qualifications • Appian Certified Associate Developer • Appian Pro UX Badge (Coming Soon) • Implementing SAFe Certification • Note: These roles can be performed by Appian developers rather than specialists

Project level resourcing.

At the project level, Appian recommends small, multi-disciplinary teams, called Pods, that deliver outcomes through short iterations, called sprints. Typically, these teams consist of 4 to 7 team members, and contain a Scrum Master, a Product Owner*, and the development team consisting of architecture, development, business analysis, and testing capabilities.

In some cases, specialized skill sets, such as DevOps/System Administrator or Integrator team members, are required based on the scope of work the team is delivering. For complex implementations, create dedicated Technical and Infrastructure Pod(s) to focus on the delivery of infrastructure components, DevOps pipelines, and common components and other specialized technical stories.



**Within the delivery team, the product owner role is typically provided by the customer, not the delivery partner.*

Appendix C - Appian Project Delivery Best Practices

Appian suggests that the following best practices be specified as part of a contract:

- The delivery partner will implement the solution using only out-of-the-box Appian platform features and functionality unless otherwise confirmed by an Appian Architect.
- Appian Health Check should be scheduled to automatically run every increment in all non-production environments. All the new high-risk findings must be addressed before the increment ends. Other findings should be added to the product backlog and prioritized along with other backlog items. The list of findings includes but is not limited to: failures to adhere to design best practices, performance issues, process sizing, system resource utilization, etc.
- Appian Health Check should also be scheduled to automatically run every month in the production environment. All high-risk findings must be prioritized for remediation in the following increment and addressed before the next production deployment. Add other findings to the product backlog and prioritize along with other backlog items.
- Architectural reviews and configuration peer reviews will be conducted as part of the implementation. The Senior Appian Designer or Appian Architect will validate that the implemented configurations adhere to the agreed-upon architectural direction and meet technical requirements. During Peer Reviews, the Designers, the Senior Appian Designer and/or Appian Architect review individual configurations ensuring that best practices are followed and that the solution will perform as required. For more information, please review Appian's best practice readiness checks and application reviews.

- All User Interfaces must follow Appian's published UX Design Guide. The Appian Architect should ensure compliance against a project tailored User Interface Style Guide for all applications in the Appian platform.
- The partner will create a high-level system design that describes process, data, integrations, and security for the system's future state. The system architecture will describe the functional design of the system at a high level. This description will include key news events, tasks, records, reports, and actions.
- The partner will proactively collaborate with the stakeholder community to ensure that the project direction, requirements, and designs are aligned to the stakeholder's vision and business needs. As required, the partner will use techniques to collaborate on and confirm direction including developing prototypes as necessary.
- The partner will allow project stakeholders to provide feedback, and will gain on-going approval for completed product features.
- The partner will collect system performance requirements and develop and execute automated performance testing scripts. Performance testing activities will be planned to occur throughout the delivery of the project life cycle, ensuring that issues are identified and corrected early.

appian

Appian is the unified platform for change. We accelerate customers' businesses by discovering, designing, and automating their most important processes. The Appian Low-Code Platform combines the key capabilities needed to get work done faster, Process Mining + Workflow + Automation, in a unified low-code platform. Appian is open, enterprise-grade, and trusted by industry leaders. For more information, visit **appian.com**.

WP-937761531-0822