

Resource Management Plan

Source File: generated-documents\management-plans\resource-management-plan.md

Generated: 16/07/2025 at 13:57:19

Generated by: Requirements Gathering Agent - PDF Converter

Resource Management Plan

Generated by adpa-enterprise-framework-automation v3.2.0

Category: management-plans

Generated: 2025-07-14T21:18:52.550Z

Description: PMBOK Resource Management Plan

Resource Management Plan

Project: ADPA - Advanced Document Processing & Automation Framework

Version: 3.2.0

Date: July 2025

Document Owner: Project Management Office (PMO)

Prepared By: ADPA Core Team

1. Purpose

The purpose of this Resource Management Plan is to define the approach for identifying, acquiring, managing, and monitoring resources—human, technical, and material—required to successfully deliver the ADPA (Advanced Document Processing & Automation) Framework. This plan ensures resource

alignment with the project’s modular, scalable, and enterprise-ready objectives, emphasizing compliance, automation, and AI-powered document management.

2. Resource Management Objectives

- Ensure availability and optimal utilization of all resources (human, technical, external).
 - Support standards-compliant automation for BABOK v3, PMBOK 7th Edition, and DMBOK 2.0.
 - Enable secure, scalable, and maintainable deployments for enterprise environments.
 - Facilitate smooth integration with AI, document, project, and data management platforms.
 - Proactively anticipate resource risks and bottlenecks, ensuring business continuity.
-

3. Resource Types

3.1 Human Resources

Role	Core Responsibilities	FTEs / Allocation	Key Skills / Requirements
Project Manager	Oversight, resource coordination, stakeholder management	1 (Full Time)	PMBOK, Agile, Enterprise IT
Lead Developer	Architecture, code reviews, release management	1 (Full Time)	Node.js, TypeScript, Microservices
Backend Developers	Core logic, API development,	2–3 (Full/Part	Node.js, TypeScript,

Role	Core Responsibilities	FTEs / Allocation	Key Skills / Requirements
	AI/ML integration	Time)	Express.js, AI APIs
Frontend Developers	Next.js admin portal, UI/UX, CLI tools	1–2 (Part Time)	React, Next.js, Tailwind CSS
DevOps Engineer	Deployment automation, CI/CD, Docker/Kubernetes, security	1 (Part Time)	Docker, Kubernetes, Azure/AWS, CI/CD
QA/Test Engineer	Test planning, automation, performance & integration testing	1 (Part Time)	Jest, TDD/BDD, API/UX testing
Business Analyst	Standards mapping, template design, compliance validation	1 (Part Time)	BABOK, PMBOK, DMBOK, documentation
Technical Writer	User guides, API documentation, onboarding resources	1 (Part Time)	Markdown, OpenAPI, GitHub, Confluence
Integration Engineer	SharePoint, Confluence, Adobe, SSO integrations	1 (On Demand)	OAuth2, API integration, Microsoft Graph

Role	Core Responsibilities	FTEs / Allocation	Key Skills / Requirements
Support/Community Lead	Issue triage, enterprise support, user community management	1 (On Demand)	Customer support, GitHub, ITSM

Resourcing Notes:

- Team skills must cover both TypeScript backend and modern frontend stacks.
- Roles may be filled by the same individual in smaller deployments.

3.2 Technical Resources

3.2.1 Development & Test Environments

- **Source Code Repository:** GitHub (private and open-source branches)
- **CI/CD:** GitHub Actions, Azure DevOps (for enterprise customers)
- **Node.js Runtime:** 18.0.0 or higher
- **TypeScript:** 5.7.2 or higher
- **Docker:** For containerized builds (Q2 2025 roadmap)
- **Test Suites:** Jest, ts-jest for unit/integration/performance

3.2.2 Infrastructure & Hosting

- **API Servers:** Express.js, scalable via Docker/Kubernetes
- **Admin Portal:** Next.js/React, hosted on Azure/AWS/On-prem as needed
- **Database:** JSON config (default), extensible to SQL/NoSQL for scale
- **Cache/Queue:** Redis (for high throughput/performance scenarios)
- **Monitoring:** Winston, Morgan, custom health checks, and metrics endpoints

3.2.3 AI & Cloud Providers

- **AI APIs:** OpenAI, Google AI, GitHub Copilot, Ollama, Azure OpenAI

- **Document APIs:** Adobe PDF Services, Microsoft Graph (SharePoint/Office365)
- **Integration APIs:** Atlassian Confluence, SSO (Active Directory, SAML, OAuth2)

3.2.4 Licensing & Compliance

- **Software Licenses:** MIT License (core project), commercial/enterprise licensing for integrations as needed (Adobe, Microsoft, Atlassian)
 - **Regulatory Compliance:** GDPR, SOX, PCI DSS, Basel III, MiFID II, FINRA, ISO 27001/9001
-

4. Resource Acquisition Plan

4.1 Human Resources

- **Internal Staffing:** Assign core roles from the existing IT/Engineering team.
- **External Support:** Engage with contractors for specialized integrations (Adobe, SharePoint).
- **Community Contributions:** Leverage open-source contributions for templates and plugin modules.
- **Training:** Provide onboarding for standards (BABOK, PMBOK, DMBOK) and secure coding.

4.2 Technical/Cloud Resources

- **Cloud Accounts:** Secure subscriptions for OpenAI, Google AI, Azure, Adobe, and Microsoft Graph.
 - **API Keys & Credentials:** Store securely in environment files (`.env` , `.env.adobe.creative`) with limited access.
 - **Development Environment:** Provision VMs/containers for isolated development and testing.
 - **Access Management:** Role-based access for production, staging, and test environments.
-

5. Resource Management and Allocation

5.1 Human Resource Scheduling

- **Agile Delivery:** 2-week sprints with clear resourcing per sprint backlog.
- **Key Milestones:**
 - Q1: BABOK and PMBOK features, multi-provider AI
 - Q2: DMBOK, Docker/Kubernetes, analytics
 - Q3: SSO, workflow automation, mobile support

5.2 Technical Resource Planning

- **Capacity Planning:** Monitor usage spikes (e.g., during batch document generation, API load) and scale infrastructure accordingly.
- **Failover/Redundancy:**
 - Multi-provider AI fallback (OpenAI, Google AI, Ollama, etc.)
 - Redundant API nodes/load balancing for production
- **Integration Readiness:** Maintain up-to-date credentials and test connections for each integration (Adobe, SharePoint, Confluence).

5.3 Third-Party Resource Utilization

- **AI Usage:** Monitor API quotas/limits for OpenAI, Google AI, etc.; implement failover and usage reporting.
- **Cloud Document Services:** Ensure licensing and quotas for Adobe, SharePoint, and Confluence are monitored.
- **Compliance Audits:** Schedule periodic compliance checks (GDPR, SOX, PCI, etc.) with legal/security teams.

6. Tools, Templates, and Automation

- **DevOps Tooling:** Use GitHub Actions and Azure DevOps for CI/CD and release automation.
- **Templates:** Maintain a library of compliant templates for all supported frameworks (BABOK, PMBOK, DMBOK).

- **Admin Portal:** Centralized resource and project management via Next.js admin interface.
- **Monitoring & Reporting:** API health endpoints, analytics dashboard, and system logs for resource usage tracking.

7. Resource Monitoring and Control

- **Performance Metrics:** API endpoints and admin portal dashboards to monitor CPU, memory, request volume, and AI provider status.
- **Testing Automation:** Jest and integration tests run on each commit; failure triggers alerts.
- **Access Reviews:** Regularly audit user and API key access; rotate secrets as per policy.
- **Change Management:** Use conventional commits, code reviews, and pull request templates for controlled updates.

8. Risk Management

Risk	Mitigation Strategy
AI API Quotas/Outages	Multi-provider failover, usage alerts, local fallback (Ollama)
Staff Turnover	Knowledge base, onboarding docs, code documentation
Integration API Changes	Automated integration tests, monitoring for breaking changes
Compliance Breaches	Regular audits, automated policy checks
Infrastructure Bottlenecks	Automated scaling, load testing, performance monitoring

Risk	Mitigation Strategy
Credential Leaks	Secure vaults, .env files excluded from source, key rotation

9. Communication and Reporting

- **Daily Standups:** Resource bottlenecks flagged for immediate resolution.
- **Weekly Status Updates:** Resource utilization and risks reported to PMO and stakeholders.
- **Issue Tracking:** GitHub Issues/Discussions for bugs, features, and resource blockers.
- **Documentation:** All resource plans, onboarding guides, and usage docs maintained on GitHub Wiki and Confluence.

10. Continuous Improvement

- **Retrospectives:** Post-sprint reviews to optimize resource allocation and identify gaps.
- **Community Feedback:** Encourage enterprise users and open-source contributors to suggest improvements.
- **Metrics-Driven Tuning:** Adjust resource plans based on real-world usage and analytics.

11. Appendices

- **A. Key Contacts:**
 - Project Sponsor: [Contact Us](#)
 - Core Team: See [GitHub Contributors](#)
- **B. Resource-Related Scripts:**

- See package.json scripts for build, test, deployment, and integration operations.

- **C. Reference Standards:**

- BABOK v3, PMBOK 7th Edition, DMBOK 2.0, GDPR, SOX, PCI DSS
-

Approval:

This Resource Management Plan is approved by the ADPA Project Sponsor and PMO. All team members and contributors are expected to adhere to the guidelines and processes herein to ensure efficient, standards-compliant, and successful project delivery.
