

DevOps Service Portfolio

CLOUD SERVICES

- Cloud Enablement
- Cloud Migration
- Cloud Engineering
- Hybrid platform development

DEVTESTSECOPS

- Continuous Integration
- Continuous Delivery
- Test automation
- Security automation
- Test Data management / Service Virtualization
- DevTestSecOps Transformation

CYBER SECURITY

- Secure SDLC
- Application, infrastructure and cloud security
- Security testing

AMS & IMS

- Clouds & Private infrastructure Support
- Custom and COTS Applications Support
- Application and Infrastructure modernization
- Operational Excellence transformation
- Functional testing

AIOPS

- Application and Infrastructure monitoring and performance management
- Operational Analytics
- ITSM automation
- Robotic Operations

TESTING SERVICES

- QA Consulting
- Test Automation & Test Data Management
- Functional Testing
- Specialized Testing

Understanding the Scope

EPAM to provide recommendations for improvements of its approach to the Continuous Delivery and Continuous Integration process.

The client would like to carry out a detailed assessment of its CI/CD processes.

As an outcome of this engagement, we will provide recommendations and a roadmap for the future improvements based on the Discovery findings.

PROPOSAL HIGHLIGHTS:



3-week phased discovery delivered by a team of seasoned experts in CI/CD and Test Automation.



Combination of onsite and offshore presence to ensure effective collaboration within reasonable costs



Proven Assessment methodology and tools to capture and analyze the current state and define the future roadmap.

Our Proposal

Our goal is to conduct a detailed analysis of the existing development, testing and deployment practices and provide the best possible recommendation to implement a robust and efficient DevOps processes.

TIMELINE: The DevOps Discovery will take 3 weeks, and is comprised of three phases:

- Weeks 1: **Assessment Phase**, focused on obtaining holistic information regarding the current state
- Weeks 2: **Recommendations Design** based on the Assessment Phase findings
- Weeks 3: **Implementation Strategy and Roadmap Design**, including long term and short term improvements and quick wins.

TEAM: A team of onsite and offshore experts experienced in conducting DevOps audits:

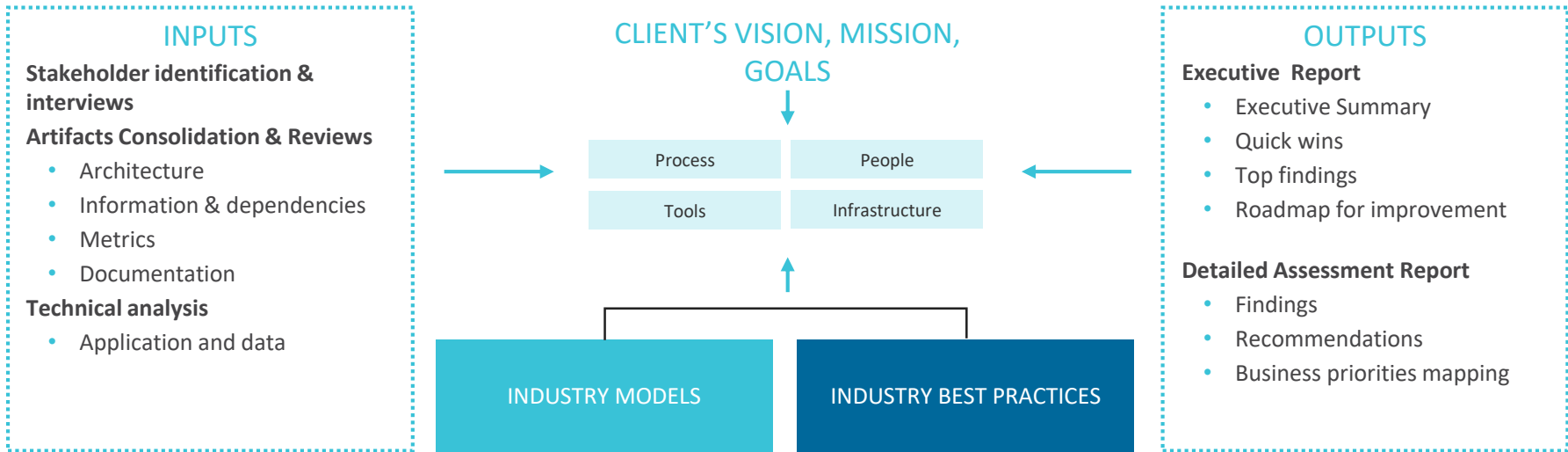
- CI/CD Architect
- Azure Architect
- Test Automation Expert

METHODOLOGY: The audit will be conducted following a time-proven [assessment framework](#), designed to provide maximum results with minimal disruption to the client's operations.

COMMERCIALS: Fixed Cost engagement, with total estimated cost of \$

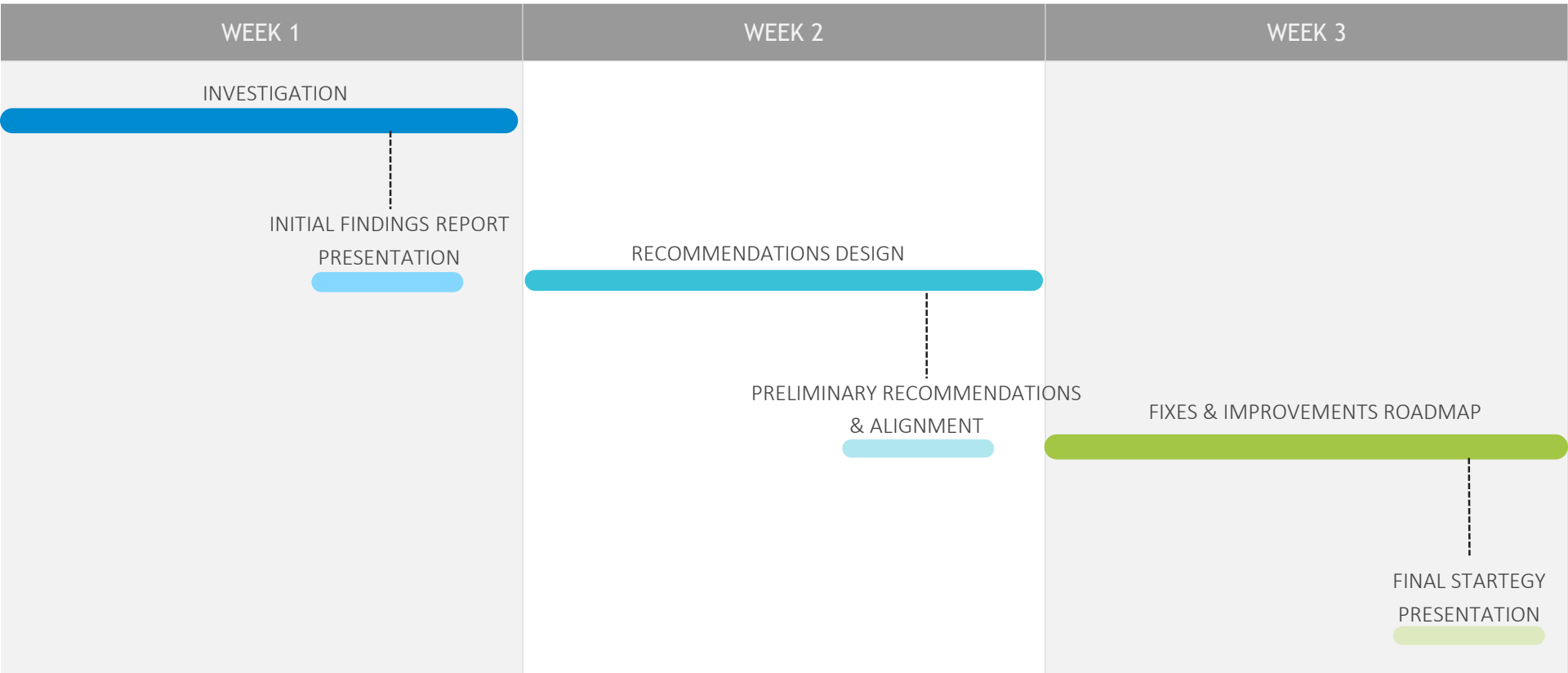
Our Assessment Framework

THE EPAM ASSESSMENT FRAMEWORK USES A COMBINATION OF VARIOUS INDUSTRY BEST PRACTICES AND MODELS TO UNDERSTAND CURRENT STATE, BUSINESS OBJECTIVES AND FUTURE STATE



Focus on desired business outcomes
Actionable report with transformation roadmap
Mastering collaboration art

Full Assessment Timeline



Phase 1: Assessment

ACTIVITIES

- Identify key Customer stakeholders for interviews and schedule assessment interview meetings
- Define business objectives, key drivers, product roadmap
- Assets inventory – tools, infrastructure, dependencies
- Architecture review – application, infrastructure, monitoring
- People dynamics – teams and their ways of working
- SDLC review – development, testing, operations
- Current CI/CD process review
- Test strategy review (current manual and automated tests)
- Review configuration management and deployment automation

DELIVERABLES

- Assessment Report (DRAFT), including:
 - Current State Overview
 - Findings report
- Weekly status reports

Phase 2: Recommendations Design

ACTIVITIES

- Workshop session(s) with key stakeholders
 - Define strategical goals and DevOps roadmap
 - Prioritize the gaps, objectives and deliverables
 - Analyze technical debt and define improvements
 - Initial design of the new capabilities (CI/CD, Monitoring, Test Automation, Metrics, SDLC flow)
 - Identify skills and training required
- Document the solution and recommendations
- Scope and document implementation strategy (Projects, Resources/Costs, Timeline)
- Conduct Playback sessions with key stakeholders on observations and key findings

DELIVERABLES

- Implementation Plan (DRAFT)
 - Timeline & milestones
 - Resource/skillsets/locations
 - Implementation cost estimate
- Assessment Report (DRAFT)
 - High-Level Architecture Design
 - DevOps Technology Roadmap
 - Technical Debt Analysis

Phase 3: Implementation Strategy & Roadmap

ACTIVITIES

- Identify recommendations and prioritized implementation roadmap CI/CD Enablement, technical debt reduction and better alignment with industry best practices
- Create implementation project plan
- Estimate the effort required to execute the proposed roadmap
- Potential efficiency gains

DELIVERABLES

- Executive Summary Presentation
- Assessment Report (FINAL)
 - Architecture Design Document
 - DevOps Technology Roadmap
 - Technical Debt Analysis
- Implementation Plan (FINAL)
 - Timeline & milestones
 - Resource/skillsets/locations
 - Implementation cost estimate

Proposed Assessment Catalogue

ASSESSMENT CATEGORY	ASSESSMENT AREA	ROLE				TIME (days)	COST ESTIMATES	DELIVERABLES (recommendations)
		Delivery Manager	DevOps Architect	DevOps Engineer	QA Engineer			
SDLC Process	Way of Working: teams' collaboration, structure and processes	+				5	\$	RACI matrix Team organization roles and responsibilities Governance model overview including organizational flow inside the teams and collaboration with other teams
	GitFlow/GitHub flows and code review		+			3	\$	Source code organization with code review process and its promotion through different environments
	Software stages and dependencies (technology)		+	+		5	\$	List of technical risks and bottlenecks Software stage readiness (definition of Done and technical debts)
CI/CD Readiness	CI/CD stages and flows		+	+		5	\$	Unified and centralized CI/CD approach Continuous integration elements and automated execution CI/CD workflow which includes full automation of those processes
	Non-functional (Load, Performance, Security, Web Vulnerabilities) requirements		+		+	2	\$	Security and performance quality gates Analytics metrics Risks of non-functional coverage absence
	CI/CD Strategy and Toolset		+	+		4	\$	CI/CD strategy overview Toolset mapping to CI/CD processes Automation coverage of CI/CD execution process
Environment Support	Automated provisioning coverage		+	+		4	\$	Environment automation provisioning and configuration management level
	Scalability and Risks		+	+		2	\$	Scalability and software adoption readiness Risks mitigation plan
	Stress Resistance (immutable infrastructure)		+	+		2	\$	Infrastructure stress test report Disaster Recovery readiness Fault tolerance
	Reporting and Monitoring		+	+		2	\$	Overview of reporting, monitoring and alerting systems with defined layers and approaches Identified gaps of monitoring and reporting processes
Quality Assurance and Test Automation Readiness	Quality Gates and Test coverage		+		+	3	\$	Types of quality gates and automated coverage CI/CD implementation readiness
	Regression and integration test coverage		+		+	2	\$	Percentage of regression and integration testing automation coverage
	Overall coverage and implementation in CI/CD pipeline		+		+	2	\$	Quality gate as continuous improvement feedback report
Managed Services and Global Roll-out	Managed Services Team readiness	+	+			5	\$	SLA and quality metrics Support team composition and scalability assessment
	Global Roll-out readiness	+	+			5	\$	List of activities of "industrialized" roll-out Roll-out Team readiness "Local"/individual market's needs

Defined Transformation Journey Based on Your DevOps Maturity

	CULTURE	PROCESS	PEOPLE	TECHNOLOGIES
LEVEL-5 OPTIMIZED	<ul style="list-style-type: none"> Articulated business strategy Clear business requirements Rapid feedback Ownership mindset 	<ul style="list-style-type: none"> Distributed orchestration Performance management Continuous deliveries Six Sigma Continuous testing Predictive reporting 	<ul style="list-style-type: none"> Interdisciplinary teams KPI-centric teams Continuous education Common knowledge transfer 	<ul style="list-style-type: none"> Tooling as a product Toolset optimization Self services
LEVEL-4 OPERATIONALLY MANAGED	<ul style="list-style-type: none"> Articulated business vision Clear product requirement Frequent collaborative communication Strategic innovation 	<ul style="list-style-type: none"> Complex orchestration Frequent deliveries Lean development Qualitative testing Strategic reporting 	<ul style="list-style-type: none"> Maintain orchestration Product-centric teams In-level tech skills Common knowledge 	<ul style="list-style-type: none"> Analytics and intelligence Self-healing Continuous deployment
LEVEL-3 DEFINED	<ul style="list-style-type: none"> Articulated business goals Clear project requirements Rapid Inter-team communication Innovation by design 	<ul style="list-style-type: none"> Simple orchestration Automated deliveries Automated testing Innovation by design Integrated testing Consolidated reporting 	<ul style="list-style-type: none"> Maintain scripts and creating orchestration Project-centric teams In-level tech skills 	<ul style="list-style-type: none"> Knowledge management Toolset integration Build/Test automation Integrated monitoring Continuous delivery Automated env. creating
LEVEL-2 MANAGED	<ul style="list-style-type: none"> Clear delivery requirements Rapid intra-team communication Innovation on demand 	<ul style="list-style-type: none"> PM Scheduled deliveries Agile Simple Scripting Requirements based testing Reporting on demand 	<ul style="list-style-type: none"> Maintaining scripts Delivery-centric teams In-level tech skills Knowledge base 	<ul style="list-style-type: none"> Project planning tools Core monitoring Build automation Functional tooling testing Standardized environment build
LEVEL-1 INITIAL	<ul style="list-style-type: none"> Uncommunicated vision Restricted communication Sub-innovating 	<ul style="list-style-type: none"> Manual process Inconsistent deliveries Ad-Hoc testing Inconsistent reporting 	<ul style="list-style-type: none"> Working on tickets Silos Exclusive knowledge 	<ul style="list-style-type: none"> No collaboration tools No Automation Manual environment build Manual testing

EFFORT ↑

COST ↓