Table 1. Proposed fundamentals and complements DevOps practices

Id.	Name	Description	T	Pr	C	Pe
1	Continuous Integration (CI)	Software development practice that requires developers to embed code in a shared repository many times a day [24]. Each new version of the project is then verified using an automated build, allowing teams to detect bugs more easily and quickly [24].	+	+		+
2	Continuous Delivery (CD)	Practice that allows developed source code that has passed all validation tests to be ready for a pre-production environment as soon as possible.	+	+		+
3	Continuous Testing (CT)	This practice incorporates continuous, pre-programmed, and automated code testing as the application code is written or updated. These tests can speed the delivery of code to production.	+	+		+
4	Requirements Management (RM)	Practice related to requirements management in tools to make it easier to track, test, analyse, visualize, and communicate project requirements to stakeholders.		+	+	+
5	Data Management (DM)	Practice related to selecting, obtaining, maintaining and using data safely and efficiently.	+	+		+
6	Security Monitoring (SM)	Practice related to integrating testing and security controls into daily QA, operations, and development work.	+	+		+
7	Strategic Direction (SD)	Practice in charge of mobilizing people and resources towards the use of DevOps environments, making things happen without overvaluing the benefits that the tools can offer to work teams, generating action within the organization.		+		+
8	Configuration Management (CM)	The practice of controlling and managing changes to software using version control in a standard and repeatable way.	+	+		+
9	Continuous Monitoring and Observability (CMO)	The practice of proactively monitoring, alerting, and acting in key areas to provide teams with visibility into application health throughout the lifecycle of a software product.	+	+		+
10	Education around DevOps (EaD)	Practice that drives better software delivery and organizational performance.		+	+	
11	Continuous feedback and innovation (CFI)	Practice in charge of permanently maintaining the socialization and / or documentation of the work team's learning.			+	+
12	Measurement of Culture (MC)	Practice that allows to measure the capabilities that drive the delivery of software and organizational performance to determine improvement actions.		+	+	
13	Continuous Deployment (CD)*	Practice that strives to automate the deployment of software to production environments without human intervention.	+	+		+
14	Infrastructure as Code (IaC)*	Practice in charge of automating the provisioning of the infrastructure necessary for the construction of a software product at any stage, using descriptive or high-level languages to code more versatile and adaptable implementation and provisioning processes.	+			+
15	Privileged Access Management (PAM)*	A practice that integrates policies, processes, and tools to protect, manage, and monitor access, users, and privileged credentials.	+	+		+
16	Continuous Learning (CL)*	Practice that facilitates learning new skills and knowledge continuously.		+	+	+
17	Continuous Experimentation (CE)*	Practice in charge of motivating the participation and generation of new ideas independently, writing and changing specifications during development.		+	+	+
18	Work Satisfaction (WS)*	Positive emotional state of the staff regarding the company and the work carried out in it.			+	+

Acronyms used: T (Tools), Pr (Processes), C (Culture), Pe (People), more information about the proposed dimensions can be consulted in section 3.1.