

A quick reference material

Kallil de Araujo

May 24, 2023

Abstract

When studying for job interviews I frequently had to spend quite some time researching possible concepts that could come up. So, to make my life less complicated, I created this file with the objective of helping me save some time when studying for those situations.

Chapter 1

Methodologies

In the IT context, a methodology refers to a structured approach or set of principles and practices that guide the planning, development, implementation, and management of IT projects or processes. It provides a systematic framework for organizing and executing tasks, as well as a common language and set of tools to facilitate collaboration among team members. A methodology encompasses a range of activities, techniques, and guidelines tailored to address specific challenges and ensure successful outcomes in IT projects.

A methodology typically outlines a step-by-step process for undertaking IT initiatives, including project initiation, requirements gathering, design, development, testing, deployment, and maintenance. It may incorporate various methodologies or frameworks, such as Agile, Waterfall, Scrum, or Lean, depending on the specific needs of the project. The choice of methodology often depends on factors such as project size, complexity, team structure, and organizational culture.

A well-defined methodology brings several benefits to the IT context. It promotes consistency and standardization in project execution, allowing for more predictable outcomes and better resource management. It helps identify and mitigate risks early in the project lifecycle, facilitating proactive decision-making. Additionally, a methodology supports effective communication and collaboration among project stakeholders, ensuring shared understanding and alignment of objectives. By providing a structured approach, a methodology enhances project transparency, facilitates progress tracking, and enables continuous improvement through feedback and lessons learned.

1.1 Agile

Agile is an iterative and collaborative approach that focuses on flexibility and adaptability. It emphasizes frequent communication, incremental development, and delivering working software in short iterations.

1.2 Waterfall

Waterfall is a sequential and linear approach where each phase of a project is completed before moving to the next one. It follows a structured process with defined milestones, making it suitable for projects with well-defined requirements and minimal changes.

1.3 Scrum

Scrum is an Agile framework that utilizes cross-functional teams to deliver software incrementally. It involves iterative sprints, daily stand-up meetings, and continuous collaboration to foster flexibility and rapid development.

1.4 Kanban

Kanban is a visual project management approach that focuses on continuous delivery and workflow optimization. It uses a Kanban board to visualize tasks, limit work in progress, and optimize the flow of work.

1.5 Lean

Lean methodology aims to eliminate waste and maximize value in the software development process. It emphasizes continuous improvement, customer focus, and efficient resource utilization.

1.6 DevOps

DevOps combines software development (Dev) and IT operations (Ops) to enhance collaboration and streamline the software delivery lifecycle. It focuses on automation, continuous integration, continuous delivery, and close collaboration between development and operations teams.

1.7 Rapid Application Development (RAD)

RAD is a methodology that emphasizes rapid prototyping and iterative development. It involves user feedback and frequent iterations to quickly build and refine software applications.

1.8 Extreme Programming (XP)

XP is an Agile methodology that emphasizes close collaboration, continuous feedback, and a high degree of customer involvement. It emphasizes practices like test-driven development, pair programming, and frequent releases.

1.9 Spiral

The Spiral methodology is a risk-driven approach that combines elements of both waterfall and iterative development. It involves iterative cycles where the project progresses through planning, risk analysis, prototyping, and customer evaluation.

1.10 Feature-Driven Development (FDD)

FDD is an iterative and incremental methodology focused on delivering tangible features quickly. It emphasizes domain modeling, iterative development, and feature-level planning and tracking.

Chapter 2

Important ETL Concepts