

Projeto de Bloco

Engenharia Disciplinada de Softwares

Graduação em Engenharia de Software - 2020

Tarefas Solicitadas na Aula Passada - 23/07/2020

1. Estudar o capítulo 1 do livro-texto do Projeto de Bloco.
2. Ler o Documento de Visão do Projeto de Bloco. Anotar as dúvidas.
3. Ler as Regras para Construção de TPs e AT.
4. Ler o enunciado do TP1 a ser entregue no dia 03/08/2020.

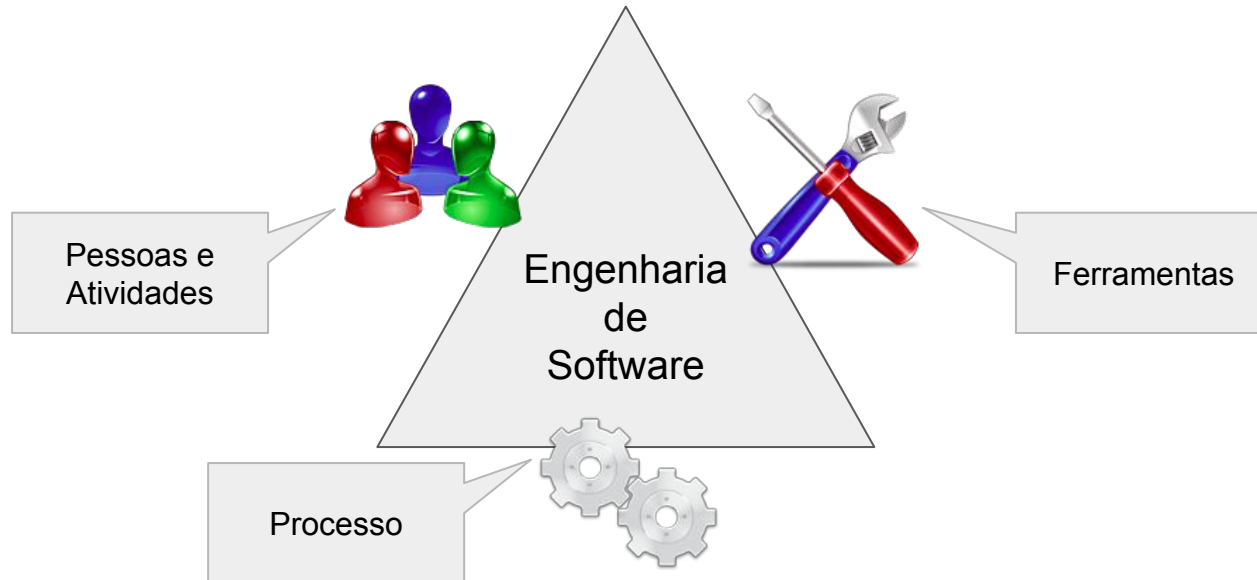
Etapa 1 Aula 2

Disciplined Agile Delivery

Engenharia de Software

Engenharia de Software

- É a combinação e integração de Pessoas, Ferramentas e Processos para o desenvolvimento de software com eficiência e qualidade.



Processo de Software

- A base para a Engenharia de Software, segundo Pressman, é composta pelo conceito de Processo.
- Ele define o Processo de Software como um conjunto de atividades, ações e tarefas realizadas na criação de algum produto de trabalho.



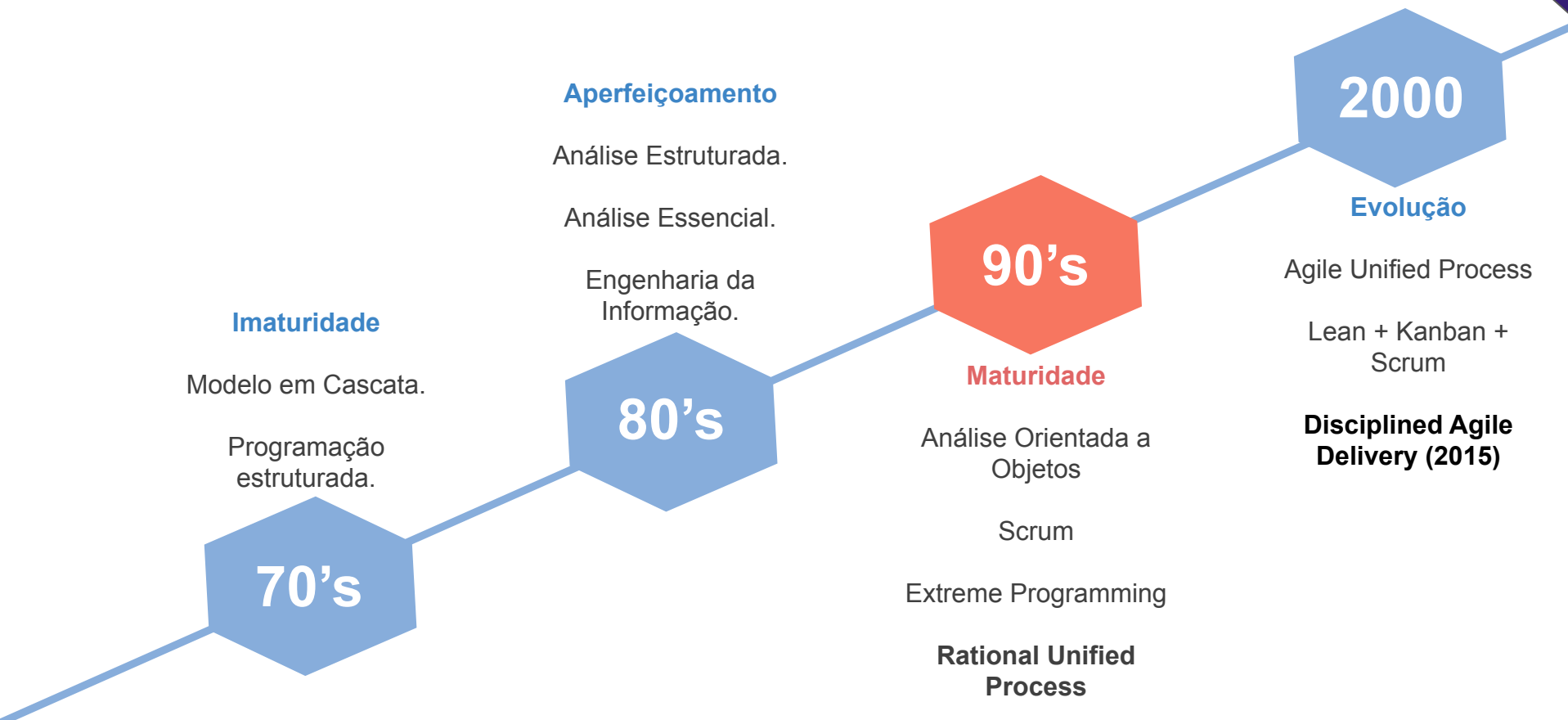
Processos X Metodologia

- O **Processo de Desenvolvimento de Software** é derivado de uma metodologia e tem por objetivo guiar o passo-a-passo para a construção de um sistema. **Cada empresa tem o seu processo.**
- **Metodologia de Desenvolvimento de Software** é o **estudo** de como implementar e melhorar os processos de desenvolvimento de sistemas.



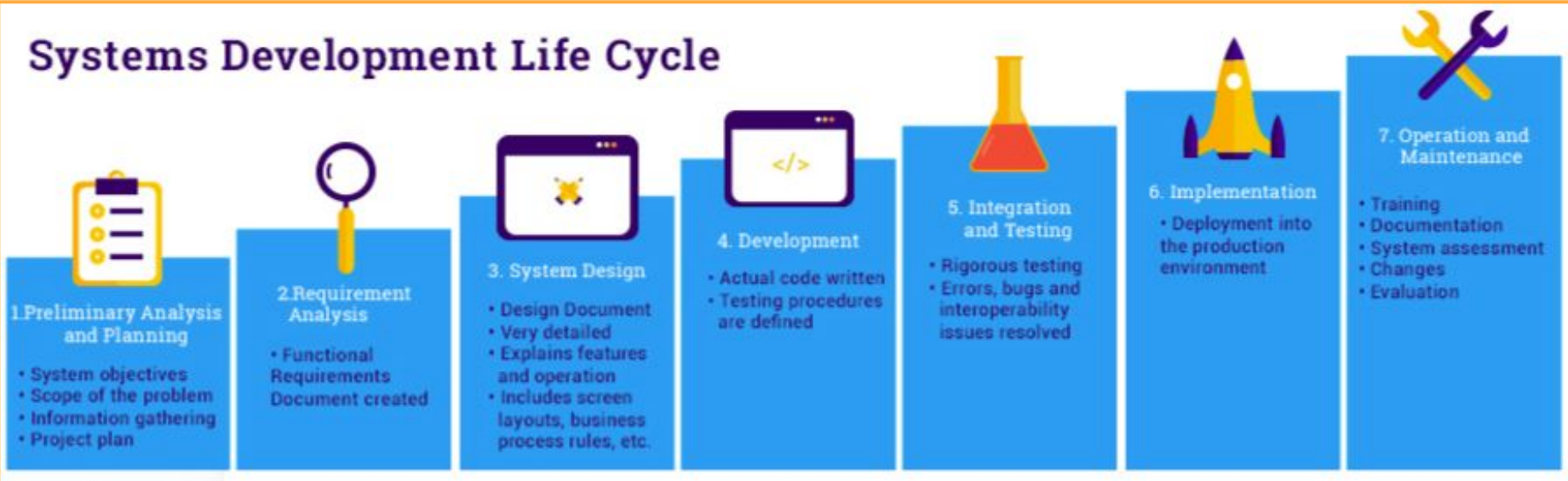
-logia = um campo de estudo ou um estudo de um assunto em particular.

Métodos e Práticas ao Longo do Tempo



Pessoas e Atividades

Systems Development Life Cycle



Pessoas e Atividades

Atividade	Responsável
Levantamento de Requisitos	Analista de Requisitos
Análise de Requisitos	Analista de Sistemas
Arquitetura	Arquiteto de Software
Projeto (<i>Design</i>)	Projetista
Implementação (Construção)	Programador
Testes	Analista de Testes
Implantação	Analista de Suporte (Infraestrutura)

Pessoas e Atividades

- **Levantamento de Requisitos** determina quais são os requisitos que o sistema deve suportar – as **funcionalidades** que o sistema precisa implementar e atender.
- **Análise** visa a modelagem do “**O Que**” o sistema deve fazer, que é conhecido sobre o domínio do negócio que o sistema se propõe a melhorar.

Pessoas e Atividades

- O papel do **Arquiteto** é incluir detalhes sobre ferramentas e tecnologias na solução proposta para o sistema. Ele trata dos requisitos não funcionais.
- **Projeto** (também conhecido como Design) visa a modelagem, “**Como**” o sistema suportará os processos de negócio.

Pessoas e Atividades

- As atividades de **Implementação** (Construção) desempenhadas pelos programadores produzem o sistema na plataforma escolhida (como .Net, Java ou PHP, por exemplo).
- Durante o processo de construção, conforme cada componente vai sendo liberado pelo programador, são aplicados **Testes** para garantir a qualidade do produto final.

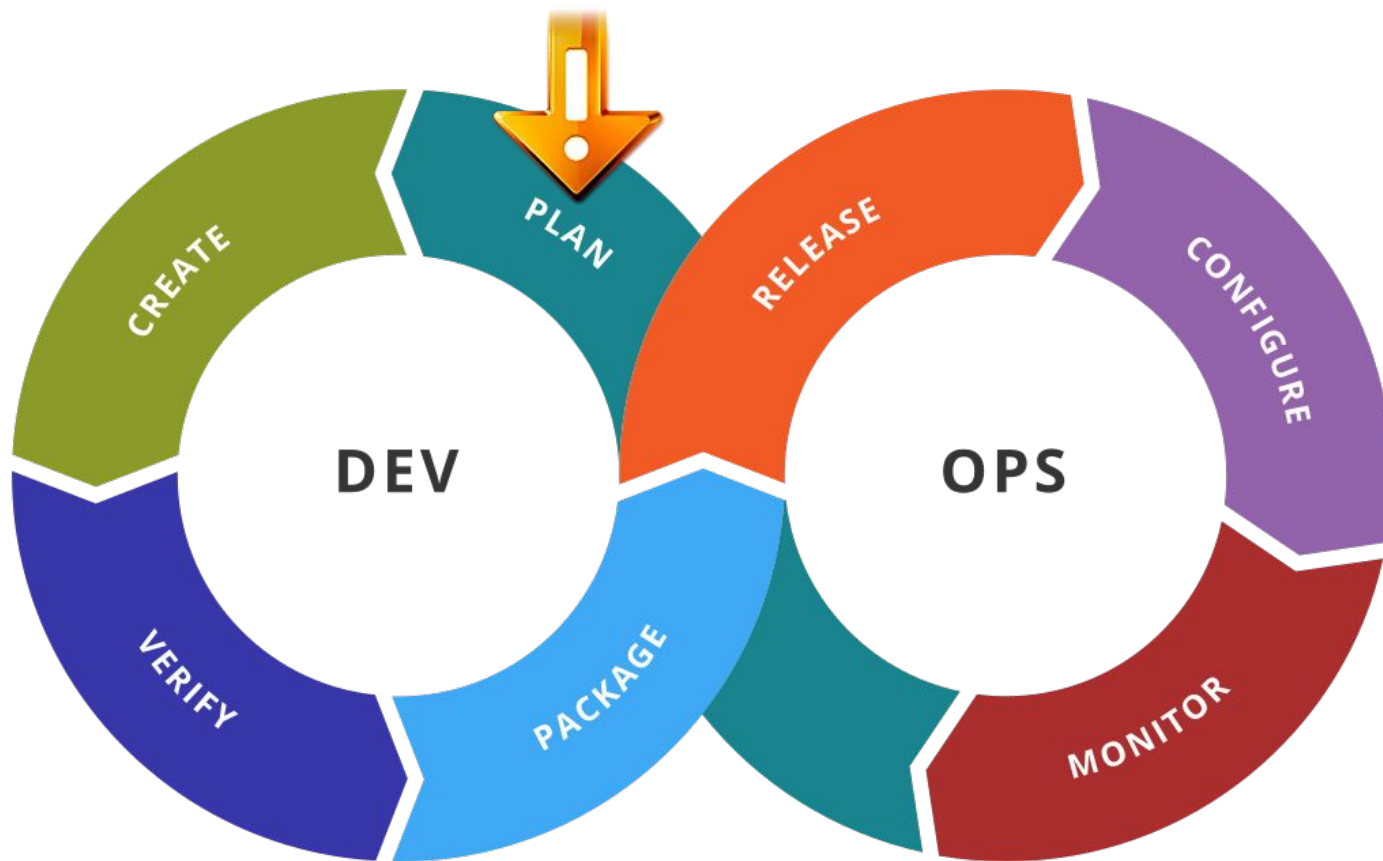
Pessoas e Atividades

- Tendo sido aprovado, cada componente propõe um módulo a ser **Implantado** de forma a permitir o seu uso pelo cliente.
- Os **Analistas de Infraestrutura** fazem a ligação entre o sistema desenvolvido e a plataforma de hardware e software onde este será implantado.

Ferramentas

- **DevOps** deriva da junção das palavras "desenvolvimento" (development) e "operações" (operations).
- É uma prática de **Engenharia de Software** que tem por objetivo unificar e harmonizar o desenvolvimento com a operação.
- A característica principal do movimento **DevOps** é defender a **automação** e **monitoramento** em todas as fases da construção do software, da integração, teste, liberação para implantação e gerenciamento de infraestrutura.

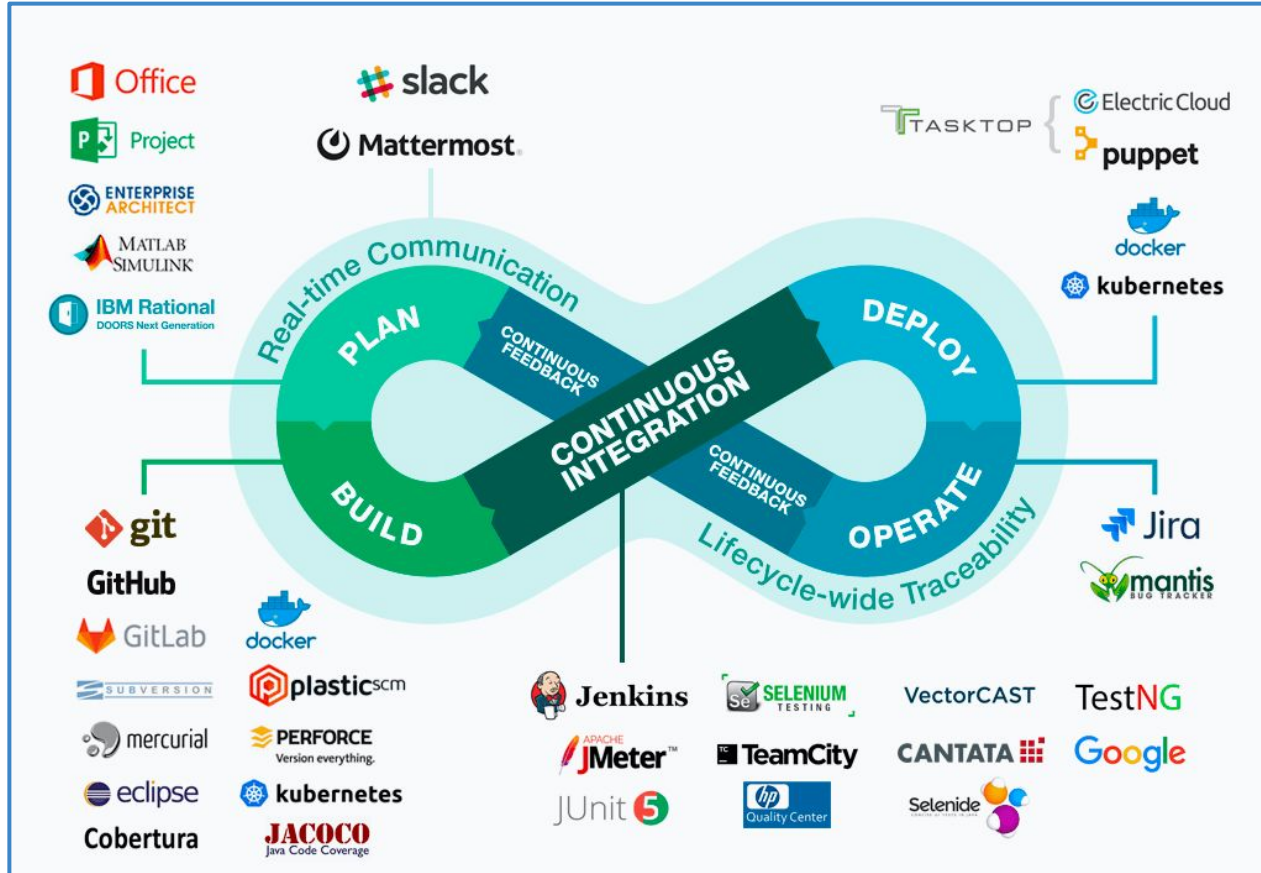
Ferramentas



Ferramentas

- **DevOps** é destinado a ser um modo de trabalho integrado: em vez de uma única ferramenta **DevOps**, existem conjuntos de várias ferramentas.
 - **Codificação** - desenvolvimento e revisão de código, ferramentas de gerenciamento de código-fonte, fusão (merge) de código.
 - **Compilação** - ferramentas de integração contínua, estado de compilação.
 - **Teste** - ferramentas de teste contínuo que fornecem feedback sobre riscos do negócio.
 - **Pacote** - repositório de artefato, etapa de pré-implantação de aplicação.
 - **Liberação** - gerenciamento de mudança, aprovações de liberação, automação de liberação.
 - **Configuração** - configuração e gerenciamento de infraestrutura, ferramentas de Infraestrutura como Código.
 - **Monitoramento** - monitoramento de desempenho de aplicações, experiência do usuário final.

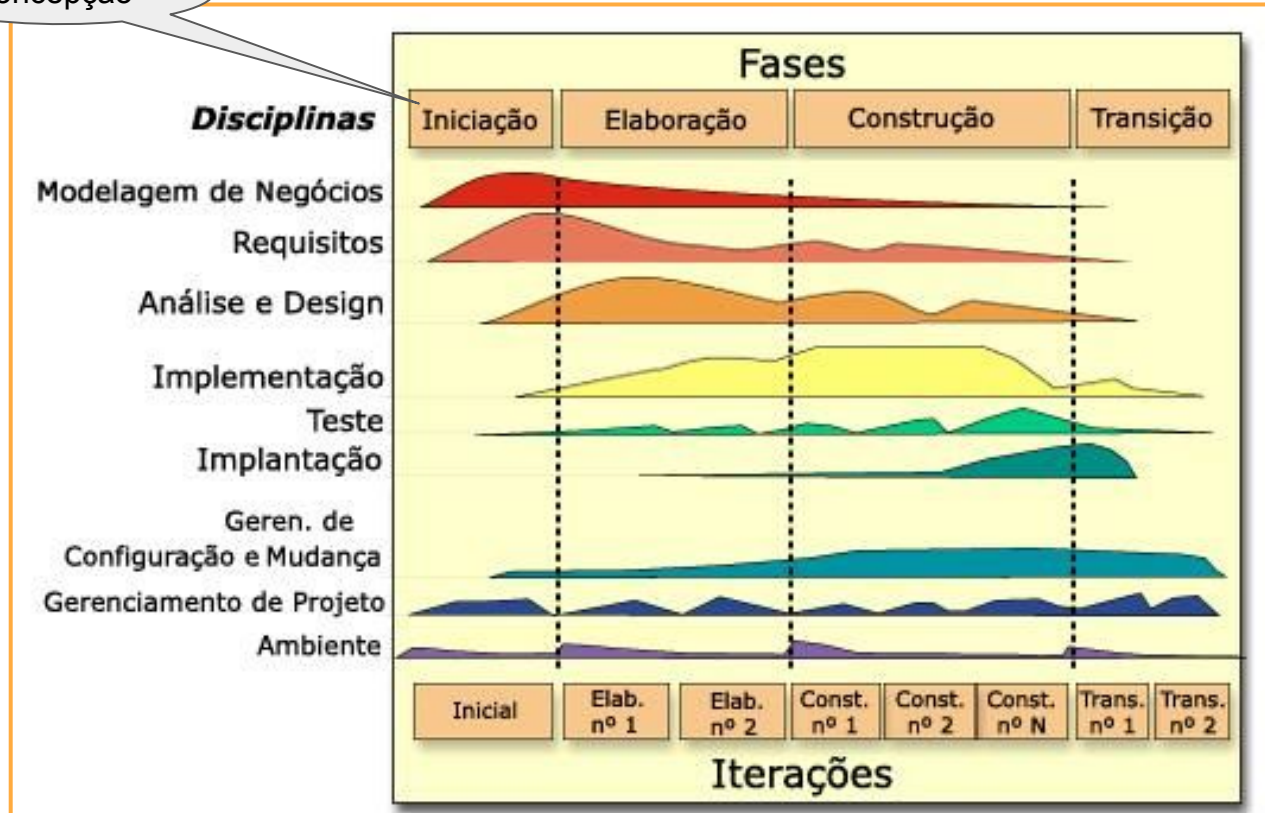
Ferramentas



Processo Unificado

Fases do Processo Unificado

Concepção



Fase de Concepção

- A fase de concepção do Processo Unificado está relacionada com as etapas **comunicação com o cliente** e de **planejamento**. Nesta fase, os requisitos de negócio devem ser identificados.
- Os **requisitos fundamentais** serão descritos com a utilização de casos de uso preliminares que apresentam as características desejáveis para cada grupo de usuários.
- Outro documento desta fase é o rascunho da arquitetura do sistema, que deve conter apenas os principais subsistemas e suas características.
- Este rascunho deve ser expandido em um conjunto de modelos que representarão diferentes visões do sistema.

Fase de Elaboração

- Na fase de elaboração os casos de uso preliminares desenvolvidos na concepção devem ser refinados e expandidos.
- A representação arquitetural também deve ser expandida para abranger as cinco visões do sistema que você observou na figura anterior.
- Nesta fase também pode ser criada uma versão inicial do sistema executável. Além disso, o planejamento deve ser revisto para garantir que o escopo, os riscos e o cronograma permaneçam razoáveis.

Fase de Construção

- Com base no modelo arquitetural, os componentes de software são desenvolvidos ou adquiridos. Esses componentes devem tornar cada caso de uso operacional para os usuários do software.
- As atividades de montagem dos componentes e testes unitários e de integração devem ser realizados também.
- Os testes de aceitação devem ser obtidos a partir dos casos de uso para que sejam executados nesta etapa.

Fase de Transição

- Na fase de transição, as últimas tarefas de construção e as primeiras atividades de implantação são realizadas. O incremento de software é entregue aos usuários finais, que devem testá-lo e enviar relatórios de feedback sobre defeitos e modificações que devem ser feitas no software que foi entregue.
- Os manuais de usuário, guias de solução de problemas e procedimentos de instalação devem ser criados pela equipe de desenvolvimento. É no final da fase de transição que o incremento de software torna-se uma versão utilizável do sistema completo.
- Na fase de produção o uso do software é monitorado com suporte ao ambiente de operação. Os relatórios de defeitos e modificações são avaliados.

Artefatos do Processo Unificado

Fase de concepção

- Documento de visão
- Modelo inicial de casos de uso
- Glossário inicial do projeto
- Caso de negócio inicial
- Avaliação inicial de risco
- Plano de Projeto, fases e iterações
- Modelo de negócio
- Um ou mais protótipos

Fase de elaboração

- Modelo de casos de uso
- Requisitos funcionais e não funcionais
- Modelo de análise
- Descrição da arquitetura de software
- Protótipo arquitetural executável
- Modelo de projeto preliminar
- Lista de riscos revisada
- Plano de projeto (planos de iteração, fluxo de trabalho, marcos, produtos técnicos) adaptados
- Manual preliminar do usuário

Fase de construção

- Modelo de Projeto
- Componentes de software
- Incremento
- Plano de procedimento e testes
- Caso de teste
- Documentação de apoio

Fase de transição

- Incremento do software entregue
- Relatório de teste beta
- Realimentação geral do usuário

Disciplined Agile Delivery

História

Scott Ambler e Mark Lines inicialmente lideraram o desenvolvimento do DAD.

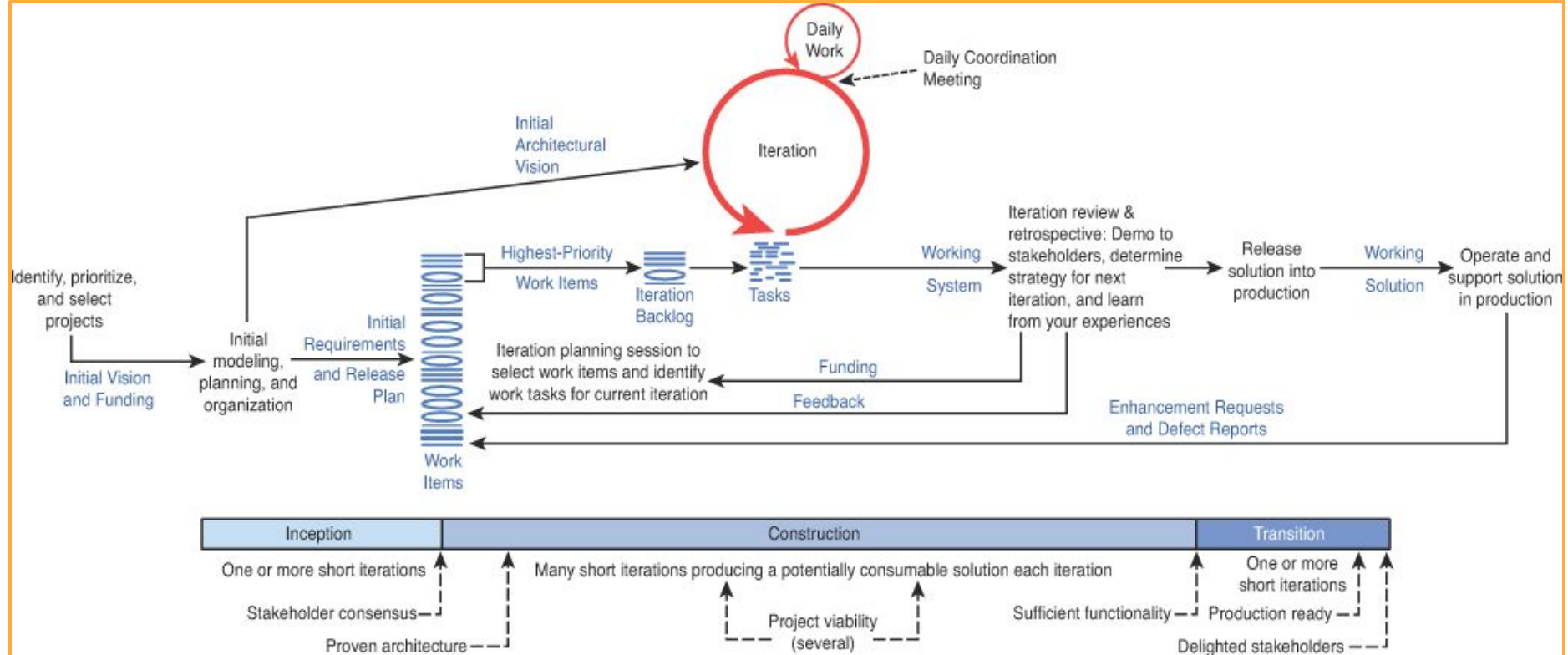
O DAD foi desenvolvido para fornecer uma abordagem mais coesa ao desenvolvimento de software ágil; **aquele que tenta preencher as lacunas do processo** que são (intencionalmente) ignoradas pelo Scrum e que tem escala no nível corporativo.

De acordo com Ambler, "Muitas metodologias ágeis - incluindo Scrum, XP, AM, Agile Data, Kanban etc - concentram-se em um subconjunto das atividades necessárias para fornecer uma solução do início ao projeto.

Ciclos de Vida

1. **Ágil** → Um ciclo de vida de projeto trifásico baseado no Scrum. As fases são Iniciação (o que às vezes é chamado "Sprint 0"), Construção e Transição (o que às vezes é chamado de Sprint de liberação).
2. **Magra** → Um ciclo de vida de projeto trifásico baseado no Kanban.
3. **Entrega Contínua Ágil** → Um ciclo de vida do produto baseado em Agile que suporta um fluxo contínuo de trabalho, resultando em liberações incrementais (geralmente uma vez por semana).
4. **Entrega Contínua: Lean** → Um ciclo de vida do produto baseado em lean que suporta um fluxo contínuo de trabalho.
5. **Exploratório** → Um ciclo de vida baseado em experimentação, baseado na inicialização enxuta, que foi estendido para abordar o desenvolvimento paralelo de produtos mínimos viáveis , de acordo com as recomendações da cynefin .
6. **Programa** → Um ciclo de vida para coordenar uma equipe de equipes.

Ciclos de Vida



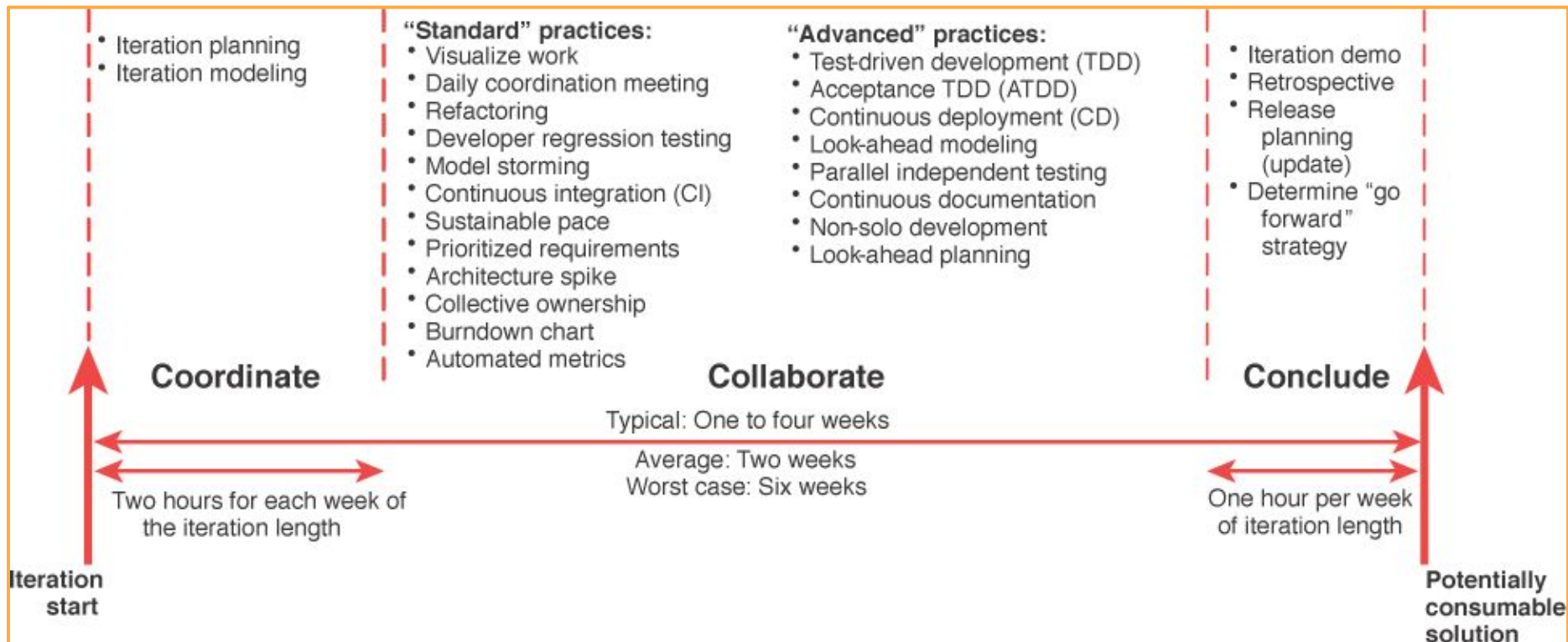
Etapas

Goals for the Inception Phase	Goals for Construction Phase Iterations	Goals for the Transition Phase
<ul style="list-style-type: none"> - Form initial team - Identify the vision for the project - Bring stakeholders to agreement around the vision - Align with enterprise direction - Identify initial technical strategy, initial requirements, and initial release plan - Set up the work environment - Secure funding - Identify risks 	<ul style="list-style-type: none"> - Produce a potentially consumable solution - Address changing stakeholder needs - Move closer to deployable release - Maintain or improve upon existing levels of quality - Prove architecture early 	<ul style="list-style-type: none"> - Ensure the solution is production ready - Ensure the stakeholders are prepared to receive the solution - Deploy the solution into production
Ongoing Goals <ul style="list-style-type: none"> - Fulfill the project mission - Grow team members' skills - Enhance existing infrastructure - Improve team process and environment - Leverage existing infrastructure - Address risk 		

Concepção



Construção



Transição



Tarefas para Próxima Aula - 06/08/2020

1. Elaborar o TP1 a ser entregue no dia 03/08/2020.
 - Mande um rascunho do TP1 para o professor dar uma olhada antes de entregar → `armenio.cardoso@prof.infnet.edu.br`
2. Estudar o capítulo 2 do livro-texto do Projeto de Bloco.