

Introdução ao OpenUP (Open Unified Process)

Diferentes projetos têm diferentes necessidades de processos. Fatores típicos ditam as necessidades de um processo mais formal ou ágil, como o tamanho da equipe e localização, complexidade da arquitetura, novidades tecnológicas, conformidade com padrões, dentre outros. No entanto, há boas práticas de desenvolvimento de software que beneficiam qualquer time de projeto e ajuda-os a serem mais eficientes.

Este artigo apresenta os blocos para a construção do OpenUP, um processo ágil e unificado que contém um conjunto mínimo de práticas para ajudar times a serem mais eficientes no desenvolvimento de software. OpenUP engloba uma filosofia pragmática e ágil que foca na natureza colaborativa do desenvolvimento do software. Ele é um processo independente de ferramenta e de pouca cerimônia que pode ser usado como está ou ser estendido para atender uma ampla variedade de tipos de projetos.

O que é o OpenUP

OpenUP é um processo de desenvolvimento de software minimamente suficiente – isso significa que somente o conteúdo fundamental é incluído. Assim, ele não provê orientação em alguns tópicos que o projeto possa tratar, como equipes grandes, situações contratuais, aplicações de segurança ou missão crítica, orientação de tecnologia específica, etc. Contudo, OpenUP é completo no sentido que ele pode ser apresentado como um processo completo para construir um sistema. Para atender as necessidades que não são cobertas no seu conteúdo, OpenUP é extensível para ser usado como base no qual o conteúdo do processo pode ser adicionado ou adaptado como necessário.

OpenUP é um processo ágil. Embora o OpenUP seja leve, há muito mais para ser ágil do que simplesmente ser leve. Práticas ágeis mais reconhecidas são voltadas para obter comunicação do time no qual todos compartilham o entendimento do projeto. Métodos ágeis têm chamado nossa atenção de volta à importância de coordenar o entendimento, beneficiando os stakeholders ao invés de entregas improdutivas e formalidade.

OpenUP têm uma característica essencial de um Processo Unificado magro que aplica abordagens interativas e incrementais dentro de um comprovado ciclo de vida estruturado. OpenUP é baseado em casos de uso e cenários, gerenciamento de risco e uma abordagem centrada na arquitetura orientada ao desenvolvimento.

Princípios do OpenUP

OpenUP é orientado a quatro princípios principais listados abaixo. Os princípios capturam a intenção geral por trás do processo e cria a fundação para interpretar papéis e produtos de trabalho, e realizar tarefas:

- **Colaboração para alinhar interesses e compartilhar entendimento.** Este princípio promove práticas para fomentar um ambiente saudável entre o time, permitindo a colaboração e desenvolvimento de um entendimento compartilhado do projeto.

- **Balanceamento de prioridades concorrente para maximizar o valor do cliente.** Este princípio promove práticas que permitem os participantes do projeto e stakeholders desenvolver uma solução que maximize os benefícios dos stakeholders e está de acordo com as restrições do projeto.
- **Foco antecipado na arquitetura para minimizar os riscos e organizar o desenvolvimento.** Este princípio promove práticas que permitem o time focar numa arquitetura para minimizar os riscos e organizar o desenvolvimento.
- **Envolver para continuamente obter feedback e melhorar.** Este princípio promove práticas que permitem o time obter feedback contínuo e antecipado dos stakeholders, e demonstrar valor incremental a eles.

Cada princípio do OpenUP suporta uma afirmação do Manifesto Ágil, como visto na tabela 1.

Princípio do OpenUP	Afirmação do Manifesto Ágil
Colaborar para alinhar interesses e compartilhar entendimento	Indivíduos e interações sobre processos e ferramentas
Balanceamento de prioridades concorrentes para maximizar valor para o cliente	Colaboração com o cliente mais que negociação de contratos
Foco antecipado na arquitetura para minimizar os riscos e organizar o desenvolvimento	Software em funcionamento mais que documentação abrangente
Envolvimento contínuo para obter feedback e melhoria	Responder a mudanças mais que seguir um plano

Tabela 1 – mapeamento entre os princípio do OpenUP e o Manifesto Ágil

Como o OpenUP é organizado

OpenUp é organizado em duas dimensões diferentes e correlacionadas: conteúdo de métodos e conteúdo de processos. O conteúdo de método é onde os elementos de método (papéis, tarefas, artefatos e orientação) são definidos, independentemente de como eles são usados no ciclo de vida do projeto. O conteúdo do processo é onde os elementos do método são aplicados no sentido temporal. Muitos ciclos de vida diferentes para tipos de projetos diferentes podem ser criados do mesmo conjunto de elementos de métodos (mais detalhes estão na seção Processo abaixo).

Áreas de conteúdo

O conteúdo do OpenUP atende organizações de trabalho no níveis pessoal, do time e do stakeholders, como visto na Figura 1.

No nível pessoal, os membros do time no projeto OpenUP contribuem com seus trabalhos em **micro-incrementos**, que tipicamente representam o resultado de algumas horas a alguns dias de trabalho. A aplicação evolui um micro-incremento por vez e o progresso é efetivamente visto a cada dia. Os membros do time abertamente compartilham seu progresso diário nos micro-incrementos, que aumentam a visibilidade do trabalho, a confiança e o trabalho de equipe.

O projeto é dividido em interações: intervalos planejados e com tempo definidos, tipicamente medidos em semanas. OpenUP ajuda o time a concentrar apropriadamente seus esforços através de **ciclos de vida de interações** a fim de entregar valor incremental ao cliente de maneira previsível – uma versão completa testada ou um incremento ao produto ao fim de cada interação.

OpenUP estrutura o **ciclo de vida do projeto** em quatro fases: Início, Elaboração, Construção e Transição. O ciclo de vida do projeto proveem os stakeholders fiscalização, transparência e mecanismos de direção para controle dos fundos do projeto, escopo, exposição ao risco, valor fornecido e outros aspectos do processo.

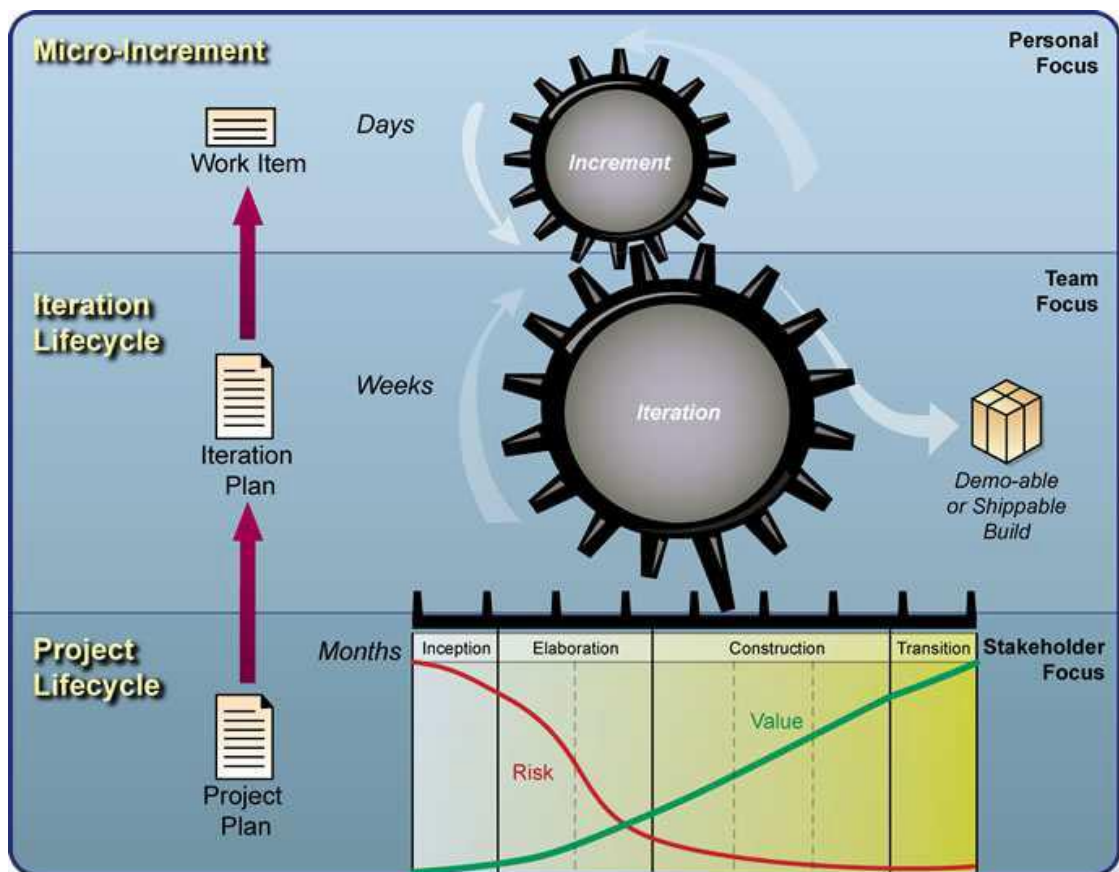


Figura 1 – Organização do trabalho e foco do conteúdo no OpenUP

Papéis

A habilidade essencial necessária aos pequenos times co-aloçados são representados pelos papéis do OpenUP:

- **Stakeholder** representa o interesse do grupo cujas necessidades devem ser satisfeitas pelo projeto. Esse é um papel que pode ser executado por qualquer um que é (ou potencialmente será) afetado materialmente pelo resultado do projeto.
- **Analista** representa o cliente e o usuário final no que diz respeito a obter a entrada dos stakeholders para entender o problema a ser resolvido e pela captura e configuração das prioridades para os requisitos.

- **Arquiteto** é responsável por projetar a arquitetura do software, que inclui tomar as decisões técnicas chave que definem todo o design e implementação do projeto.
- **Desenvolvedor** é responsável por desenvolver uma parte do sistema, inclusive projetá-lo de acordo com a arquitetura, e então implementá-lo, executar testes unitários e integrar os componentes que fazem parte da solução.
- **Testador** é responsável pelas principais atividades de testes, como identificar, definir, implementar e conduzir os testes necessários, bem como registrar as saídas dos testes e analisar os resultados.
- **Gerente de projetos** lidera o planejamento do projeto em colaboração com os stakeholders e o time, coordenando interações com os stakeholders, e mantendo o time do projeto focado em atingir os objetivos do projeto.
- **Qualquer papel** representa alguém no time que pode realizar tarefas gerais.

Disciplinas

O conteúdo de método do OpenUP é focado nas seguintes disciplinas: Requisitos, Arquitetura, Desenvolvimento, Teste, Gerenciamento de Projeto e Gerenciamento de Configuração e Mudança.

Outras disciplinas e áreas de preocupação foram omitidas, como Modelagem de Negócios, Ambiente, Gerenciamento de Requisitos avançado e configuração de ferramentas de Gerenciamento de Configuração. Esses assuntos ou são considerados desnecessários para um projeto pequeno ou são tratados por outras áreas da organização, fora do time do projeto.

Tarefas

Uma tarefa é uma unidade de trabalho que um papel pode realizar. No OpenUP, há 18 tarefas que os papéis realizam como executante primário (o responsável por executar a tarefa) ou executante adicional (apoiando e fornecendo informação usada na execução da tarefa). A natureza colaborativa do OpenUP é manifestado ao ter os executantes primários trabalhando com uma gama de outros indivíduos quando realizam uma tarefa.

Artefatos

Um artefato é qualquer coisa que é produzida, modificada ou usada em uma tarefa. Os papéis são responsáveis pela criação e atualização dos artefatos. Artefatos estão sujeitos a controle de versão durante o ciclo de vida do projeto.

Os 17 artefatos do OpenUP são considerados os artefatos essenciais que um projeto deve usar para capturar informações sobre o produto e relacionadas ao projeto. Não há obrigação em capturar informação em artefato formal. A informação pode ser informalmente capturada numa lousa (por exemplo para design e arquitetura), reunir notas (por exemplo para avaliações de status), etc. Modelos que oferecem uma forma padrão e sem mistérios para capturar informações. Projetos podem usar artefatos OpenUP ou substituí-los pelos seus próprios.

Processos

Conteúdo de método reusável é criado separadamente da sua aplicação em processo. Conteúdo de método fornece explicações passo a passo, descrevendo como os objetivos específicos de desenvolvimento são alcançados, independente da ordem dos elementos do método dentro do ciclo de vida de desenvolvimento.

Processos pegam esses elementos dos métodos e relaciona-os em sequências semi-ordenadas que são customizadas para especificar o tipo de projeto. Elementos de métodos são organizados em pedaços de processos reusáveis chamados *padrões de capacidade*, fornecendo uma abordagem de desenvolvimento consistente para as necessidades comuns de projeto. Esses padrões são feitos de tarefas organizadas (do conteúdo do método) em atividades, agrupando-os em uma sequência que faça sentido para uma área particular onde o padrão é aplicado.

Padrões podem ser pequenos e focados em áreas particulares como gerenciamento de interação, iniciação do projeto, definição da arquitetura e assim por diante. Esses são considerados os blocos de construção básicos para criar padrões maiores ou processos de entrega (definidos abaixo).

Um exemplo de bloco de construção básico no OpenUp é o padrão de Incremento de Solução de Desenvolvimento, como representado na figura 2.

Esta atividade fornece uma forma de realizar um planejamento baseado em metas e execução de um trabalho. Trabalho é assumido por desenvolvedores e o progresso do trabalho é medido baseado no atingimento das metas usando um código-fonte projetado, testado pelo desenvolvedor e integrado.

O item de trabalho pode ser um caso de uso, um cenário, um requisito de apoio ou uma requisição de mudança. Um contexto pode ser especificado quando um item de trabalho está para ser desenvolvido em um incremento. Desenvolvimento pode ser focado, por exemplo, em uma camada (por exemplo interface com o usuário, lógica de negócio ou acesso ao banco de dados), ou em um componente. Se um contexto é especificado ou não, a responsabilidade do desenvolvedor é criar um design e implementação para aquele item de trabalho, e escrever e executar os testes do desenvolvedor na implementação para garantir que a implementação funciona como projetado, tanto os unitários como os de integração no código base.

O padrão de *Incremento de Solução de Desenvolvimento* ocorre quantas vezes houver item de trabalho a desenvolver na interação dada.

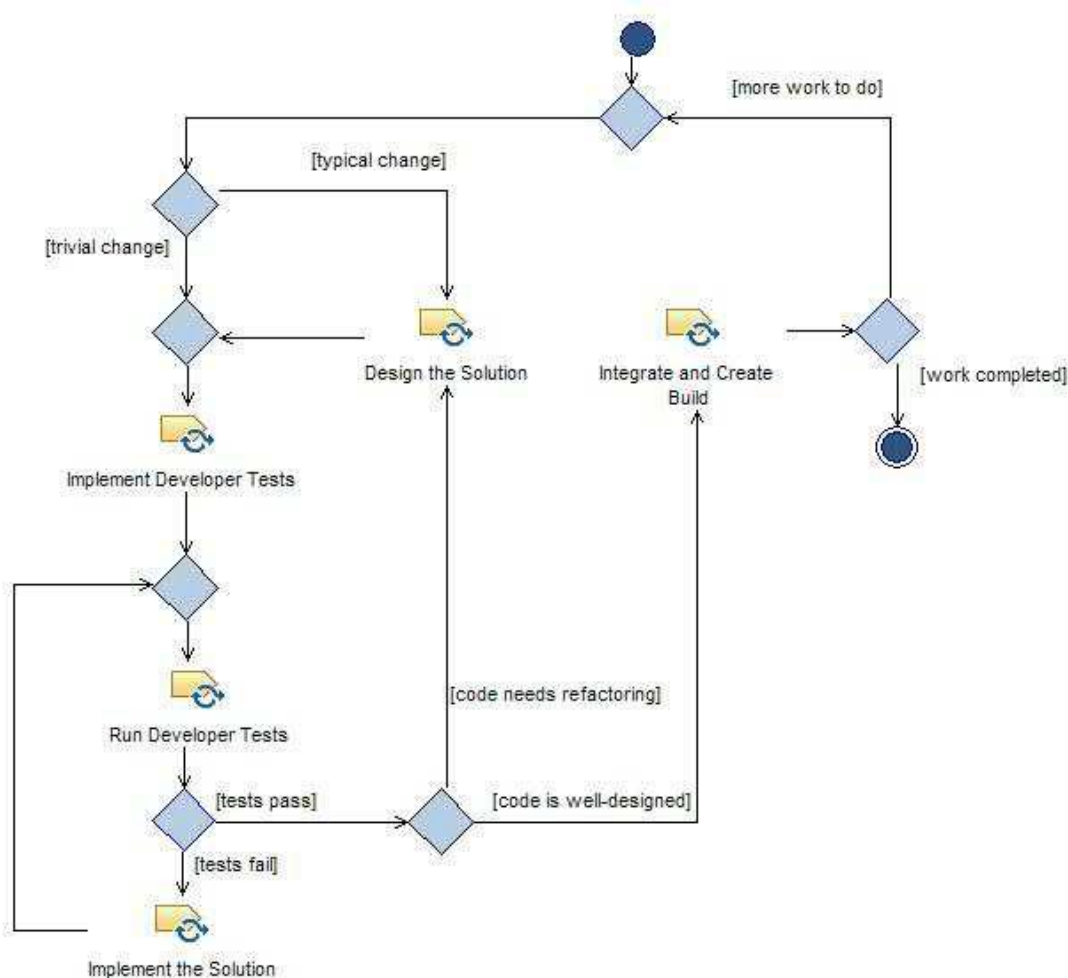


Figura 2 – Incremento de Solução de Desenvolvimento

Como mencionado anteriormente, blocos de construção básica são usados para a criação de padrões maiores, por exemplo, padrões que podem ser usados como modelos para interações – padrões contendo todas as atividades necessárias para uma interação particular dentro da fase de projeto.

OpenUP aplica as fases do Processo Unificado: Iniciação, Elaboração, Construção e Transição. Quando tomados juntos, esses blocos de construção básicos são também usados para atingir o objetivo para cada fase (vide Tabela 2 para o mapeamento desse padrões e objetivos das fases).

Padrão de modelo de interação	Objetivos das fases
<ul style="list-style-type: none"> Interação da Fase de Iniciação <ul style="list-style-type: none"> Iniciação do Projeto Interação de Planejamento e Gerenciamento Identificação e refinamento dos requisitos Acordo sobre a Abordagem Técnica 	<ul style="list-style-type: none"> Entendimento do que construir Identificação das funcionalidades chave do sistema Determinar ao menos uma solução possível Entender o custo, o prazo e os riscos associados ao projeto
<ul style="list-style-type: none"> Interação da Fase de Elaboração 	<ul style="list-style-type: none"> Obter um entendimento mais

<ul style="list-style-type: none"> • Interação de Gerenciamento e Planejamento • Identificação e Refinamento dos Requisitos • Definição da Arquitetura • Incremento da Solução de Desenvolvimento • Solução de Testes • Tarefas em Andamento 	<ul style="list-style-type: none"> • detalhado dos requisitos • Projetar, implementar, validar uma arquitetura • Mitigar os principais riscos e produzir uma agenda precisa e estimativas de custo
<ul style="list-style-type: none"> • Interação da Fase de Construção • Interação de Planejamento e Gerenciamento • Identificação e Refinamento dos Requisitos • Solução de Testes • Tarefas em Andamento 	<ul style="list-style-type: none"> • Desenvolver interativamente um produto completo que esteja pronto para transição para sua comunidade de usuários • Minimizar os custos de desenvolvimento e alcançar algum grau de paralelismo
<ul style="list-style-type: none"> • Interação da Fase de Transição • Interação de Planejamento e Gerenciamento • Incremento da Solução de Desenvolvimento • Solução de Testes • Tarefas em andamento 	<ul style="list-style-type: none"> • Realizar testes <i>beta</i> para validar as expectativas do usuário • Atingir um consenso dos stakeholders quando a implantação está completa.

Tabela 2 – mapeamento entre padrões e objetivos das fases

Quando seguimos os padrões de modelo de interação (que ocorrem quantas vezes forem necessários), nós temos um *processo de entrega* – uma abordagem completa e integrada para realizar um tipo de projeto específico (como visto na Figura 3). Um processo de entrega descreve um ciclo de vida do projeto completo e é usado como referência para executar projetos similares.

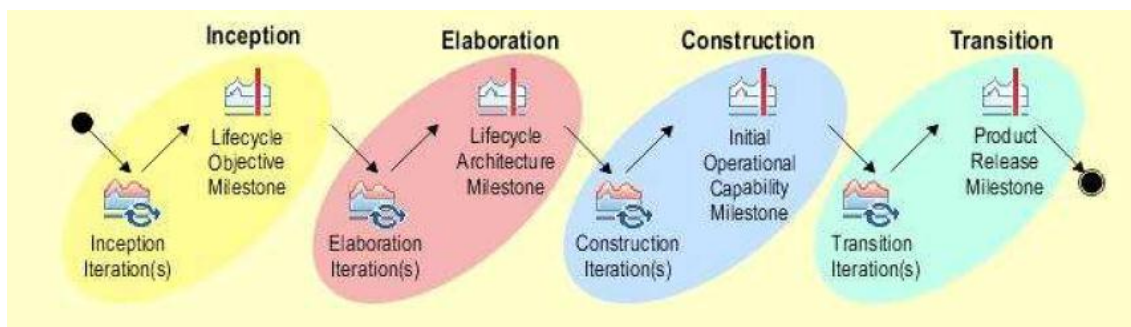


Figura 3 – Processo de Entrega do OpenUP

OpenUP tem um processo de entrega para desenvolvimento iterativo em quatro fases. Os padrões de modelo de interação são aplicados quantas vezes forem necessários, dependendo de quantas interações o time decidir executar em cada fase. Projetos com diferentes necessidades podem instanciar padrões de modelo de interação diferentemente. Por exemplo, um projeto que lida com uma tecnologia desconhecida ou uma arquitetura complexa pode precisar de mais interações na fase de Elaboração do que um projeto que lida com uma tecnologia conhecida e uma arquitetura pré-existente.

Um processo base

O processo OpenUP, embora completo no seu escopo e contexto alvo, também serve como um processo base no qual conteúdo de processo adicional pode ser construído. Plug-ins podem estender o OpenUP para incluir orientação para técnicas em grande escala (tal como Desenvolvimento Orientado ao Modelo) ou técnicas mais leves e ágeis (como Técnicas de Banco de Dados Ágeis). Vendedores de ferramentas podem criar plug-ins que anexam ferramentas de mentoria para tarefas dando instruções passo a passo de como a ferramenta pode ser usada dentro do processo.

Em adição a essa comunidade de plug-ins que podem ser misturados e combinados para construir um processo adaptado – e possivelmente mais importante para o objetivo de ter um processo que atenda as necessidades específicas de um projeto em particular – a ferramenta Eclipse Process Framework Composer que pode ser usado por autor de conteúdo in-house. Modelos específicos da organização podem ser integrados ao conteúdo do processo e novos elementos do processo de checklists e orientações à novos papéis, tarefas e produtos de trabalho podem ser introduzidos.

Conclusão

OpenUP é um processo mínimo, completo e extensível. Ele fomenta técnicas e princípios ágeis, enquanto tem um ciclo de vida estruturado comprovado que enfatiza a entrega contínua de software de qualidade que leva valor aos stakeholders.

Referências

- Manifesto Ágil: <http://agilemanifesto.org/iso/ptbr>
- Projeto Eclipse Process Framework: <http://www.eclipse.org/epf>

Autor

Ricardo Balduino é engenheiro de software sênior na IBM Rational e committer no projeto Eclipse Process Framework. É também arquiteto de conteúdo e desenvolvedor líder de conteúdo para o OpenUP. Seus 13 anos de experiência na indústria de software incluem desenvolvimento de software, treinamento e consultoria ajudando organizações a customizar e adotar as melhores práticas de desenvolvimento de software. Possui bacharelado em Ciência da Computação pela Universidade Estadual de São Paulo, Brasil.

Tradutor

Daniel Ferreira é um cara que um dia conheceu o Google Translator e agora “se acha”.