

Fases do RUP.

No RUP o ciclo de vida do software é dividido em quatro fases seqüenciais, cada uma concluída por um marco principal. Em cada final de fase é executado uma avaliação para determinar se os objetivos da fase foram alcançados, permitindo, ou não, que o projeto passe para próxima fase.

As quatro fases são:

Concepção ou Iniciação: O objectivo primordial da fase de iniciação é atingir o consenso entre todas as partes interessadas sobre os objectivos do ciclo de vida para o projeto. A fase de iniciação tem muita importância principalmente para os esforços de desenvolvimento de novos, em que há negócios e de requisitos riscos que devem ser abordadas antes que o projeto possa prosseguir. Para projetos que visam melhorias em um sistema existente, a fase de iniciação é mais breve, mas ainda está focada em garantir que o projeto é tanto vale a pena fazer e possível fazer.

Objetivos:

Os objetivos primários da fase de Concepção/Iniciação incluem:

- Estabelecer o escopo do software do projeto e as condições limite, incluindo uma visão operacional, critérios de aceitação e que pretende ser no produto e que não é.
- Discriminar os casos de uso críticos do sistema, os principais cenários de operação eo que direcionará as compensações maiores de design.
- Expositoras, e talvez demonstrar, pelo menos uma arquitetura candidata contra alguns dos principais cenários. Estimar o custo geral ea programação para o projeto inteiro (e estimativas mais detalhadas para a fase de elaboração)
- Estimar os riscos potenciais (as fontes de imprevisibilidade) (Veja Conceito: Risco)
- Preparando o ambiente de suporte para o projeto.

Atividades essenciais:

As atividades essenciais da fase de Iniciação incluem:

- Formulação do escopo do projeto. Isso envolve capturar o contexto e os requisitos mais importantes e as restrições a tal ponto que você pode derivar critérios de aceitação para o produto final.
- Planejamento e preparação de um business case. Avaliando alternativas para a gestão de risco, de pessoal, plano de projeto e custo/programação/rentabilidade tradeoffs.
- Sintetizar uma arquitetura candidata, avaliando vantagens e desvantagens em design, e em fazer / comprar / reutilizar, de modo que custo, cronograma e os recursos podem ser estimados. O objetivo aqui é demonstrar a viabilidade através de algum tipo de prova de conceito. Isto pode assumir a forma de um modelo que simula o que é exigido, ou um protótipo inicial que explora o que são consideradas áreas de alto risco. O esforço do protótipo durante a iniciação

deve ser limitada a ganhar confiança de que uma solução é possível - a solução é executada durante a elaboração e construção.

- Preparação do ambiente para o projeto, avaliando o projeto e a organização, selecionando ferramentas e decidindo quais partes do processo para melhorar.

Elaboração: O objetivo da fase de elaboração é criar a baseline da arquitetura do sistema para fornecer uma base estável para a maior parte do projeto e esforço de implementação na fase de construção. A arquitetura se desenvolve a partir de um exame dos requisitos mais significativos (aqueles que têm um grande impacto sobre a arquitetura do sistema) e uma avaliação de risco. A estabilidade da arquitetura é avaliada através de um ou mais protótipos de arquitetura.

Objetivos:

Os objetivos primários da fase de Elaboração incluem:

- Assegurar que a arquitetura, os requisitos e os planos sejam estáveis o suficiente e os riscos suficientemente mitigados para ser capaz de determinar o custo previsível e cronograma para a conclusão do desenvolvimento. Para a maioria dos projetos, ultrapassar essa marca também corresponde à transição de uma luz-e-rápido, operação de baixo risco a um custo elevado, a operação de alto risco com uma inércia organizacional.
- Para lidar com todos os riscos significativos para a arquitetura do projeto.
- Para estabelecer uma arquitetura da baseline derivada do tratamento dos cenários significativos para a arquitetura, que normalmente expõem os maiores riscos técnicos do projeto.
- Para produzir um protótipo evolutivo da produção de componentes de qualidade, bem como, eventualmente um ou mais exploratória, throw-away protótipos para mitigar os riscos específicos, tais como:
 - design / requisitos trade-offs.
 - reutilização de componentes
 - viabilidade do produto ou demonstrações para investidores, clientes e usuários finais.
- Para demonstrar que a arquitetura de baseline suportará os requisitos do sistema a um custo razoável e em um tempo razoável.
- Para estabelecer um ambiente de apoio.

Para atingir estes objectivos primários, é igualmente importante para configurar o ambiente de suporte para o projeto. Isto inclui a adaptação do processo para o projeto, preparando modelos, diretrizes e criação de ferramentas.

Atividades essenciais:

As atividades essenciais da fase de Elaboração incluem:

- Definir, validar e criar a baseline da arquitetura com rapidez e eficiência.

- Refinar a Visão, com base em novas informações obtidas durante a fase, estabelecendo uma compreensão sólida dos casos de uso mais críticos que conduzem as decisões de arquitetura e planejamento.
- Criação e planos detalhados baselining iteração para a fase de construção.
- Refinamento do processo de desenvolvimento e criação de ambiente de desenvolvimento, incluindo o processo, ferramentas e suporte de automação necessária para apoiar a equipe de construção.
- Refinamento da arquitetura e selecionar componentes. Componentes potenciais são avaliados e fazer / comprar / reutilizar decisões suficientemente compreendidas para determinar o custo da fase de construção e programar com confiança. Os componentes de arquitetura selecionados são integrados e avaliados em comparação com os cenários principais. As lições aprendidas dessas atividades podem resultar em um redesenho da arquitetura, levando em consideração designs alternativos ou reconsiderando os requisitos.

Construção: O objetivo da fase de construção é esclarecer os requisitos restantes e concluir o desenvolvimento do sistema baseado na arquitetura da baseline. A fase de construção é de certa forma um processo de fabricação, onde a ênfase é colocada no gerenciamento de recursos e controle de operações para otimizar custos, cronogramas e qualidade. Neste sentido, a mentalidade do gerenciamento passa por uma transição a partir do desenvolvimento de propriedade intelectual durante a iniciação e elaboração, para o desenvolvimento de produtos implantados durante a construção e transição.

Objetivos:

Os objetivos primários da fase de construção incluem:

- Minimizando os custos de desenvolvimento, otimizando recursos e evitando desperdícios desnecessários e retrabalho.
- Atingir a qualidade adequada com rapidez e eficiência
- Atingir as versões úteis (alfa, beta e outros releases de teste) com rapidez e eficiência
- Concluindo a análise, concepção, desenvolvimento e teste de todas as funcionalidades necessárias.
- Iterativa e incremental para desenvolver um produto completo que esteja pronto para a transição para a sua comunidade de usuários. Isso implica descrever os casos de uso restantes e outros requisitos, incrementar o design, completando a implementação e testar o software.
- Para decidir se o software, os sites e os usuários estão todos prontos para a aplicação a ser implantado.
- Para alcançar algum grau de paralelismo entre o trabalho das equipes de desenvolvimento. Mesmo em projetos menores, normalmente há componentes que podem ser desenvolvidos de forma independente um do outro, permitindo o paralelismo natural entre as equipes (recursos permitirem). O paralelismo pode acelerar as atividades de desenvolvimento significativamente, mas também

aumenta a complexidade da gestão de recursos e sincronização de fluxo de trabalho. A arquitetura robusta é essencial se um paralelismo significativo é o de ser alcançado.

Atividade essenciais:

As atividades essenciais da fase de construção incluem:

- Otimização da gestão de recursos, controle e processo.
- Desenvolvimento completo do componente e testes contra os critérios de avaliação definidos.
- Avaliação de lançamentos de produtos com base em critérios de aceitação para a visão.

Transição: O foco da Fase de Transição é assegurar que o software está disponível para seus usuários. A fase de transição pode atravessar várias iterações e inclui testar o produto em preparação para release e ajustes pequenos com base no feedback do usuário. Neste ponto do ciclo de vida, o feedback do usuário deve priorizar o ajuste fino do produto, a configuração, instalação e problemas de usabilidade, todos os problemas estruturais mais graves devem ter sido trabalhado muito antes no ciclo de vida do projeto.

Objetivos:

Até o final da Transição objetivos do ciclo de vida Fase deveriam ter sido atendidos eo projeto deve estar em uma posição para ser encerrada. Em alguns casos, o fim do ciclo de vida atual pode coincidir com o início de outro ciclo de vida no mesmo produto, conduzindo à nova geração ou versão do produto. Para outros projetos, o fim da transição pode coincidir com uma liberação total dos artefatos a terceiros que podem ser responsáveis por operações de manutenção e melhorias do sistema de entrega.

Esta Fase de Transição pode ser muito fácil ou muito complexa, dependendo do tipo de produto. A nova versão de um produto de mesa existente pode ser muito simples, enquanto que a substituição do sistema de uma nação controle de tráfego aéreo pode ser extremamente complexo.

Atividades realizadas durante uma iteração na Fase de Transição dependem do objetivo. Por exemplo, ao corrigir erros de implementação e de teste são geralmente suficiente. Se, no entanto, novas características tiverem de ser adicionadas, a iteração é semelhante a um na fase de construção que exigem análise e design, etc.

A Fase de Transição entra quando uma linha de base é maduro o suficiente para ser implantado no domínio do usuário final. Isso normalmente requer que algum subconjunto usável do sistema tenha sido concluído com nível de qualidade aceitável e documentação do usuário para que a transição para o usuário forneça resultados positivos para todas as partes.

Os objetivos primários da fase de transição incluem:

- Teste beta para validar o novo sistema contra as expectativas do usuário.
- Teste beta e operação paralela relativa a um sistema legado que é a substituição.
- Conversão de bases de dados operacionais.
- Treinamento dos usuários e mantenedores.
- Roll-out para as forças de distribuição, marketing e vendas.
- Implantação específicos de engenharia, tais como embalagens de transição, comerciais e de produção, as vendas de roll-out, campo de treinamento de pessoal.
- Atividades de ajuste, como correção de bugs, melhoria no desempenho e usabilidade.
- Avaliação das baselines de implantação contra a visão completa e os critérios de aceitação para o produto.
- Alcançar usuário auto-suporte.
- Obtenção do consentimento dos envolvidos de que as baselines de implantação estão completas.
- Obtenção do consentimento dos envolvidos de que as baselines de implantação são consistentes com os critérios de avaliação da visão.

Atividades essenciais:

As atividades essenciais da fase de Transição incluem o seguinte:

- Execução de planos de implantação.
- Material de apoio finalizar usuário final.
- Testar o produto liberado no local de desenvolvimento.
- Criação de um lançamento de produto.
- Obter feedback do usuário.
- O ajuste fino do produto com base no feedback.
- Tornando o produto disponível para os usuários

Disciplinas.

As disciplinas do RUP estão divididas em disciplinas de processo e de suporte:

- Disciplinas de processo:
 - Modelagem de Negócios:
 - Requisitos:
 - Análise e Projeto:
 - Implementação:
 - Teste:
 - Implantação:
- Disciplinas de suporte:

- Configuração e Gerenciamento de Mudanças:
- Gerenciamento de projeto:
- Ambiente:

Principais Papéis.

Papel do Analista: Este conjunto de funções é principalmente envolvido em resgatar requisitos.

- Especificar Requisitos: Tem a função de especificar e manter os requisitos do sistema detalhados ao longo do projeto.
 - Atividades: detalhar casos de uso e detalhar requisitos de software.
- Stakeholder: Este papel representa um grupo de interesse, cujas necessidades devem ser satisfeitas pelo projeto. Stakeholder é qualquer indivíduo materialmente afetados pelo resultado do projeto.
- Analista de Sistemas: Este papel lidera e coordena a elicitação de requisitos, destacando as funcionalidades e delimitações do sistema.
 - Atividades: captura de um vocabulário comum, Desenvolver especificações suplementares, desenvolver a visão, requisitar as partes interessadas, encontrar atores e casos de uso e estruturar o modelo de caso de uso.

Papel de desenvolvedor: Tem a função de concepção e implementação do software.

- Designer de banco de dados: Este papel leva o desenho da estrutura persistente de armazenamento de dados a ser utilizado pelo sistema.
 - Atividade: designer do banco de dados.
- Designer : Este papel leva o desenho de uma parte do sistema, dentro das limitações dos requisitos, arquitetura e processo de desenvolvimento para o projeto.
 - Atividades: design de class, design subsistemas, design e análise de casos de uso.
- Implementador: Esse papel desenvolve componentes de software e realiza testes de desenvolvimento para integração em subsistemas maiores, de acordo com os padrões adotados para o projeto.
 - Atividades: analisar o comportamento e o tempo de execução, implementar, executar e desenvolver testes.
- Integrator : Este papel leva o planejamento e execução de integração de elementos de implementação para produzir builds.
 - Atividades: plano de integração do subsistema, integração do subsistema, plano de integração do sistema e integração do sistema.
- Arquiteto de Software: Este papel conduz o desenvolvimento da arquitetura do sistema de software, que inclui a promoção e criação de suporte para as principais decisões técnicas que condicionam a concepção global e implementação do projeto.
 - Atividades: análise arquitetural, estruturar um modelo implementação, identificar, incorporar e excluir elementos de design.

- Designer de Interface de Usuário: Este papel coordena o projeto de interface do usuário. Isto inclui a coleta de requisitos de usabilidade e uma prototipagem da interface do usuário para atender a esses requisitos.
 - Atividades: design e prototipagem da interface de usuário.

Papel de testador: Tem como função testar o software.

- Analista de Teste: Este papel identifica e define os testes necessários, monitora o progresso testes detalhados e os resultados em cada ciclo de teste e avalia a qualidade geral. O papel também representa os interessados que não têm representação direta ou regular sobre o projeto.
 - Atividades: identifica idéias de teste e determina os resultados dos testes.
- Designer de test: Este papel leva a definir uma abordagem de teste e garantir que sua implementação seja bem sucedida. Isto inclui identificar as técnicas apropriadas, ferramentas e diretrizes para implementar os testes necessários, e fornecer orientações para esforço de teste nos requisitos correspondentes.
 - Atividades: definir abordagem de teste e a estrutura de implementação.
- Testador: Esse papel realiza testes e registra os resultados destes.
 - Atividades: implementa e executa os testes.
- Gerente de testes: Este papel lidera o esforço de teste em geral. Isto inclui a qualidade e teste de advocacia, planejamento de recursos e gestão e resolução de questões que impedem o esforço de teste.
 - Atividade: avaliar e monitorar a qualidade dos testes.

Links:

http://www.wthreex.com/rup/process/artifact/ar_itpln.htm