



Instituto Federal de Santa Catarina – IFSC
Campus São José

Processo de Desenvolvimento de Software

Programação Orientada a Objetos

Prof. Francisco de Assis S. Santos, Dr.

São José, 2015.

Processo de Desenvolvimento de Software

O desenvolvimento de software é uma atividade complexa

Grande número de projetos que não chegam ao fim, extrapolam recursos financeiros e de tempo.

Processo de desenvolvimento de software

Compreende todas as atividades necessárias para definir, desenvolver, testar e manter um software.

- Define quais atividades serão executadas ao longo do projeto, quando, como e por quem tais atividades serão executadas.

Componentes Humanos no Desenvolvimento

Gerente de projeto

- Responsável pela gerência e coordenação das atividades necessárias para a construção do sistema, além de estimar tempo e custo.

Analista

- Possui conhecimento sobre o domínio do negocio para que possa levantar os requisitos.

Projetista

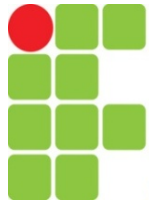
- Avalia as alternativas de solução e gera uma especificação detalhada da solução computacional (Ex: projetista de rede, de banco de dados, etc.).

Programador

- Responsável pela implementação do sistema.

Cliente

- O cliente usuário e especialista no domínio do negocio e interage diretamente com o Analista para levantar os requisitos do sistema.



Interação entre as Partes



Como o cliente explicou...



Como o líder de projeto entendeu...



Como o analista projetou...



Como o programador construiu...



Como o Consultor de Negócios descreveu...

Interação entre as Partes



Modelos de Desenvolvimento de Software

- **Modelo Cascata**
- **Modelo de Ciclo de Vida Iterativo e Incremental**
- **Prototipação**
- **Modelo Espiral**

Atividades Típicas de um Processo de Desenvolvimento

1 – Levantamento dos Requisitos

Compreensão do problema, visando permitir que usuários e desenvolvedores tenham a mesma visão do problema a ser resolvido.

2 - Análise de requisitos (ou Especificação de requisitos)

Estudo detalhado dos requisitos levantados e a construção de modelos para representar o sistema a ser construído.

Atividades Típicas de um Processo de Desenvolvimento

3 – Projeto

Determina como o sistema funcionará para atender os requisitos, de acordo com os recursos tecnológicos existentes. **A modelagem do software pode ser realizada por um conjunto de diagramas, por exemplo, pela UML.**

4 - Implementação

Ocorre a tradução da descrição computacional da fase de projeto em código executável através do uso de linguagens de programação. **É nessa fase que os diagramas/modelos criados “ganham vida”.**

Atividades Típicas de um Processo de Desenvolvimento

5 – Testes

Para verificar a corretude do sistema, levando-se em conta a especificação feita na fase de projeto.

6 - Implantação

O sistema é empacotado, distribuído e instalado no ambiente do usuário. São entregues os manuais do sistema e os usuários são treinados para utilizar o sistema.

Levantamento de Requisitos

Os requisitos são apresentados em um documento com as seguintes secções:

- **Requisitos Funcionais:** Definem as funcionalidades do sistema
 - Ex: O sistema deve permitir que cada professor realize o lançamento de notas das turmas nas quais lecionou.

À medida que o levantamento de requisitos é realizado, pode-se fazer uma modelagem das atividades encontradas, por exemplo, desenvolver diagramas de Casos de Uso pertencentes a UML.

Levantamento de Requisitos

- **Requisitos não-funcionais:** Declaram características de qualidade que o sistema deve possuir
 - Desempenho (tempo de resposta esperado);
 - Portabilidade (plataformas de hardware e software nas quais o sistema será implantado)
- **Restrições:** Impostas sobre o desenvolvimento do sistema
 - Adequação a custos e prazos, aspectos legais (licenciamento), etc.

Projeto

O projeto consiste de duas atividades principais:

Projeto de arquitetura (alto nível)

- Consiste em distribuir as classes de objetos relacionadas ao sistema em subsistemas e seus componentes.
- Diagramas UML de implementação são normalmente usados nesta fase.

Projeto detalhado (baixo nível)

- São modeladas as colaborações entre objetos, projeto da interface com o usuário e o projeto do banco de dados.
- Os diagramas UML usados nessa fase são:
 - Diagrama de classes, de caso de uso, de interação, de estados e de atividades.

Linguagem de Modelagem Unificada (UML)

A Linguagem de Modelagem Unificada (**UML**) NÃO é um método de desenvolvimento de sistema, mas sim uma linguagem de **modelagem gráfica** que pode ser aplicada para descrever e documentar um projeto de software (Tonsig, 2008).

A UML tem o propósito de fornecer aos desenvolvedores de softwares uma linguagem de modelagem visual completa, de modo a alcançar os seguintes aspectos (Tonsig, 2008):

- Disponibilização de mecanismos de especificação que possam expressar os níveis conceituais;
- Independência de processos de desenvolvimento e linguagens de programação;
- Incentivo ao crescimento das aplicações desenvolvidas no conceito da orientação a objetos.

O processo de desenvolvimento de software não está previsto na UML, o que torna uma linguagem de modelagem e não um método, mas podem-se eleger as seis fases apresentadas para o desenvolvimento de software em que a UML pode ser aplicada:

1- Levantamento dos Requisitos;

2 – Análise dos requisitos;

3 – Projeto;

4 – Implementação;

5- Testes;

6- Implantação.

Notações da UML

Em razão de não ser possível representar um sistema em sua totalidade por meio de um único diagrama, é necessário um conjunto de recursos que expressem os diversos aspectos que constituem o sistema.

Diagramas UML que serão abordados:

- Diagrama de casos de uso;
- Diagrama de classes;
- Diagrama de sequência.

Modelagem de Casos de Uso

Referências

Notas de aula do Prof. Emerson Ribeiro de Mello.

Tonsig, S. L. Engenharia de Software – Análise e Projeto de Sistemas – 2ª ed. Rio de Janeiro, Editora Ciência Moderna, 2008.