

FRAMEWORK

“Framework é um conjunto de classes que colaboram para realizar uma responsabilidade para um domínio de um subsistema da aplicação.”

Fayad e Schmidt

Um **framework**, em desenvolvimento de software, é uma **abstração** que une **códigos** comuns entre vários projetos de software provendo uma funcionalidade genérica. Um framework pode atingir uma funcionalidade específica, por configuração, durante a programação de uma aplicação. Ao contrário das **bibliotecas**, é o framework quem dita o fluxo de controle da aplicação, chamado de **Inversão de Controle**.

Framework conceitual é um conjunto de conceitos usado para resolver um problema de um domínio específico, não se trata de um software executável, mas sim de um **modelo de dados** para um domínio.

Framework de **software** compreende um conjunto de **classes** implementadas em uma **linguagem de programação** específica, usadas para auxiliar o desenvolvimento de software, atua onde há funcionalidades em comum a várias aplicações, porém para isso as aplicações devem ter algo razoavelmente grande em comum para que o mesmo possa ser utilizado em várias aplicações.

Um framework inclui código, diferentemente de um padrão de projeto, pode ser modelado com vários padrões de projeto, e sempre possuem um domínio de uma aplicação particular, algo que não ocorre nos padrões e projeto de software.

Frameworks possuem vantagens, tais como: maior facilidade para a detecção de erros, por serem peças mais concisas de **software**; concentração na abstração de soluções do problema que estamos tratando; eficiência na resolução dos problemas e otimização de recursos.

Tipos

Frameworks verticais são confeccionados através da experiência obtida em um determinado contexto específico. Exemplos: framework financeiro, recursos humanos. Frameworks horizontais não dependem do domínio da aplicação e podem ser usados em diferentes domínios. Exemplos: Interfaces gráficas, persistência, transação.

Orientação a objetos

Especificamente em **orientação a objetos**, framework é um conjunto de **classes** com objetivo de reutilização de **arquitetura de software**, provendo um guia para uma solução em um domínio específico de software. Framework se diferencia de uma simples biblioteca, pois esta se concentra apenas em oferecer implementação de funcionalidades, sem definir a reutilização de uma solução de arquitetura.

Framework em administração de empresas

É uma estrutura conceitual básica que permite o manuseio homogêneo de diferentes objetos de negócio. Serve para incrementar a disciplina de gestão e predefinir entregáveis comuns de e para cada objeto de negócio.

Um framework deve prover sugestões de solução para uma família de problemas semelhantes.

SISTEMAS CIBERNÉTICOS - A cibernética foi fundada por Norbert Wiener durante o período de 1946 – 1953. Surgiu como uma ciência interdisciplinar para relacionar as várias ciências, preencher os espaços vazios não pesquisados por nenhuma delas e permitir que cada ciência utilizasse os conhecimentos desenvolvidos pelas outras, de acordo com Wiener, os dispositivos automáticos e as criaturas vivas apresentam fortes semelhanças na sua estrutura e funcionamento.

Os sistemas cibernéticos são repetitivos por natureza, o que significa que eles tendem a repetir infinitamente seus sucessos e suas falhas. Isso é exatamente o que acontece às equipes. Sucesso aos bem sucedidos e fracasso aqueles que falham, e isso pode tomar a forma de um círculo virtuoso ou vicioso. Os sistemas cibernéticos possuem características de automação.

SISTEMAS MECANISMO DE RELÓGIO (CLOCKWORK SYSTEMS) - São deterministas e causais. Nosso sistema solar é um bom exemplo.

SISTEMAS DE CRESCIMENTO PLANEJADO (BLUEPRINTED GROWTH SYSTEMS)
São sistemas que se reproduzem não por duplicação, mas por projeto ou planejamento.

SISTEMAS COM IMAGEM INTERNA - A informação é recebida e organizada em uma imagem interna ou estrutura de conhecimento, do ambiente. Exemplos de sistemas com imagem interna são os animais, em geral.

SISTEMAS COM PROCESSAMENTO SIMBÓLICO - Sistemas que possuem capacidade de usar linguagem, como os seres vivos.

- **Conceito** - Possuem auto-consciência e, assim, são capazes de usar linguagem à nível de funcionamento humano. Em outras palavras, resultam automaticamente em ações baseadas em regras e decisões que o homem tomaria conforme certa situação ou contexto.

- **Características**

Apresentam um sistema de inteligência artificial que:

- armazena e manipula dados;
- faz a aquisição, representação e manipulação do conhecimento;
- é capaz de deduzir ou inferir novos conhecimentos, novas relações sobre fatos e conceitos, com base no conhecimento existente.

A partir daí, o sistema utiliza métodos de representação e manipulação para resolver problemas complexos, executando uma determinada ação ou função desejada.

O processo de desenvolvimento de um Sistema com Processamento Simbólico é um processo exploratório, que é feito a partir de uma série de aproximações que podemos chamar de "protótipo rápido", enquanto nos programas convencionais, o processo de desenvolvimento amplamente conhecido e utilizado é o "modelo de cascata".

A execução dos Sistemas com Processamento Simbólico é altamente interativa, enquanto nos programas convencionais é ou sequencial ou em "batch".

SISTEMAS SOCIAIS - Sistemas cujos atores compartilham uma ordem social.

SISTEMAS TRANSCENDENTAIS - Nível dos **sistemas transcendentais** - que completa a classificação dos níveis de sistemas. São os sistemas superiores, absolutos, inevitáveis, mas ignorados ou conhecidos apenas parcialmente em face da sua excessiva complexidade, e que também obedecem a uma estrutura sistemática lógica.