

[◀ VOLTAR](#)

# Definição e conceituação de ferramentas CASE: ferramentas para automatizar o processo de desenvolvimento de software

Definir e conceituar ferramentas CASE.

## NESTE TÓPICO

- Introdução
- Riscos associados com a utilização de ferramentas CASE
- Categorização de ferramentas CASE
- Norma ISO/IEC 14102
- Referências



## Introdução



Ferramentas CASE (Computer Aided Software Engineering) são uma classe de software que automatiza muitas das atividades envolvidas em várias fases do ciclo de vida.

Por exemplo, quando você estiver estabelecendo os requisitos funcionais de uma aplicação, poderá fazer uso de ferramentas de prototipação que o auxiliarão a desenvolver modelos gráficos das telas que serão apresentadas aos usuários finais, os quais poderão visualizar como a aplicação ficará após o desenvolvimento.

A seguir, projetistas de sistemas poderão utilizar ferramentas de projeto para transformar os requisitos funcionais prototipados anteriormente em documentos de projeto detalhados.

Programadores poderão usar geradores de código automatizados para converter os documentos de projeto em código e assim por diante nas várias etapas de desenvolvimento.

Ferramentas automatizadas podem ser usadas **coletivamente** (em várias etapas do processo de desenvolvimento), como citado anteriormente, ou **individualmente**.

Um exemplo de utilização individual, por exemplo, seria ferramentas de prototipação serem usadas para definir requisitos de aplicação, que, posteriormente, seriam passadas aos técnicos de projeto. Estes, por sua vez, poderiam converter de uma maneira tradicional os requisitos em projeto detalhado, usando diagramas e documentos narrativos, sem o auxílio de ferramentas automatizadas.

Ferramentas automatizadas podem também facilitar a coordenação das atividades de desenvolvimento de software por meio de data warehouses ou repositórios.

Repositórios podem fornecer um meio para armazenar e acessar informações relacionadas a um projeto, tais como planos de projeto, requisitos funcionais, documentos de projeto, bibliotecas de programas, bancos de testes, etc.

Organizações geralmente utilizam ferramentas de desenvolvimento automatizado para aumentar a produtividade, diminuir custos, melhorar o controle de projetos e aumentar a qualidade dos produtos gerados.

Contudo, somente através do gerenciamento dos vários riscos associados com as tecnologias de automatização que as organizações poderão assegurar que estão desenvolvendo sistemas com funcionalidade, segurança, integridade e confiabilidade apropriadas.

## Riscos associados com a utilização de ferramentas CASE

Conforme falamos, há riscos associados com a utilização de ferramentas CASE. Podemos citar, como exemplos, os seguintes riscos:



**Padronização inadequada** – integrar ferramentas de diferentes fornecedores (por exemplo, ferramenta de design da companhia X com ferramenta de programação da companhia Y) pode se mostrar difícil se essas ferramentas não usam classificação de dados e estruturas de código padronizadas.

Formatos de arquivos podem ser convertidos, mas normalmente não de uma maneira econômica. O mais recomendado é você utilizar ferramentas de um mesmo fornecedor ou usar ferramentas baseadas em protocolos padronizados e com comprovada compatibilidade.

Além disso, se a organização for adquirir ferramentas para somente uma parte do processo de desenvolvimento, recomenda-se que ela as adquira de um fornecedor que tenha uma linha completa de produtos, para assegurar futura compatibilidade caso novas ferramentas venham a ser adquiridas.

**Expectativas irrealistas** – organizações, às vezes, implementam tecnologias CASE para reduzir os custos de desenvolvimento. Só que implantar estratégias CASE normalmente envolve custos iniciais elevados.

Geralmente, a gerência deve estar disposta a aceitar retorno de investimento em um período de médio a longo prazo.

Recomenda-se que gerentes seniores estejam envolvidos na definição dos propósitos e das estratégias de implementação de tecnologias CASE dentro de uma organização.

**Implementação rápida** – implementar tecnologias CASE envolve mudanças significativas no ambiente de desenvolvimento tradicional.

Tipicamente, as organizações não deveriam utilizar ferramentas CASE, pela primeira vez, em projetos críticos ou em projetos com tempo de entrega curto.

Essa recomendação se deve ao fato de que a utilização adequada de ferramentas CASE exige um tempo de treinamento considerável.

O ideal é, ao adquirir uma ferramenta CASE, a empresa a utilize pela primeira vez em projetos menos complexos. Recomenda-se também que sejam introduzidas gradualmente para permitir um maior tempo de treinamento.

**Controle Fraco de Repositório** – falhas para controlar adequadamente o acesso aos repositórios CASE podem resultar em violação de segurança ou danos aos documentos de trabalho, projetos do sistema ou módulos de código armazenados. Procedimentos de controle incluem proteger os repositórios com controles adequados de acesso, versão e backup.

## Categorização de ferramentas CASE

Não existe um padrão estabelecido para a categorização das ferramentas CASE. Os termos a seguir são apenas um exemplo possível dentro desse propósito.



- **Front End ou Upper CASE:** dão suporte às fases iniciais de criação dos sistemas (fases de planejamento, análise e projeto da aplicação).
- **Back End ou Lower CASE:** dão suporte às fases de codificação testes e manutenção da aplicação.
- **I-CASE ou Integrated CASE:** engloba todos os produtos que cobrem todo o ciclo de vida do software, desde os requisitos do sistema até o controle final da qualidade.

## Norma ISO/IEC 14102

Esta norma aborda a seleção e avaliação de ferramentas CASE. Ela considera uma parte ou todo o ciclo de vida da engenharia de software.

Estabelece processos e atividades a serem aplicadas na avaliação de ferramentas e na seleção da ferramenta mais apropriada dentre possíveis candidatas.

Os processos considerados são genéricos e as organizações devem adaptá-los de acordo com suas necessidades.

Agora que você já estudou esta aula, resolva os exercícios e verifique seu conhecimento. Caso fique alguma dúvida, leve a questão ao Fórum e divida com seus colegas e professor.

## Referências

PRESSMAN, R. S. *Engenharia de software*. 7. ed. São Paulo: McGraw-Hill, 2010.

SOMMERVILLE, Ian. *Engenharia de software*. São Paulo: Addison-Wesley, 2007.



Avalie este tópico



ANTERIOR

Modelagem de processos: elementos de um modelo de processo (atividade, processo, entrega, condição, papel, exceção, comunicação)

Biblioteca  
(https://www.uninove.br/conheca-uninove/biblioteca/sobre-a-biblioteca/apresentacao/) Portal Uninove  
(http://www.uninove.br) Mapa do Site



Índice

Desenvolvimento de sistemas críticos: implicações e cuidados no processo de desenvolvimento de sistemas críticos

© Todos os direitos reservados

Ajuda?  
(https://ava.uninove.br/curso=)

