

Técnica de análise estática de código

A análise estática de código é uma técnica de análise de software que é realizada sem a necessidade de executar o programa. Em vez disso, a análise é feita diretamente no código-fonte do software, com o objetivo de detectar potenciais erros, bugs, vulnerabilidades de segurança, ou qualquer outro problema que possa afetar a qualidade do software.

Durante a análise estática de código, são utilizadas ferramentas de análise de código-fonte para analisar a estrutura, a sintaxe e a semântica do código. Essas ferramentas realizam uma análise completa do código, identificando possíveis problemas, como variáveis não inicializadas, loops infinitos, chamadas de função incorretas, erros de sintaxe, problemas de desempenho, entre outros.

A análise estática de código é uma parte importante do processo de garantia de qualidade do software, pois permite que os desenvolvedores identifiquem e corrijam problemas no código antes que o software seja lançado ou implantado em um ambiente de produção. Além disso, a análise estática de código também ajuda a melhorar a manutenibilidade e a escalabilidade do software, tornando-o mais fácil de entender e modificar.

Software:	Converter estrutura JSON para arquivos .csv
Componente avaliado:	Foi testado a leitura da estrutura JSON
Responsável pelo componente:	DIOGO GHIZONI DE ALMEIDA
Revisor:	DIOGO GHIZONI DE ALMEIDA – UC 22101390 EMANUELLY DE SOUZA PORTELA – UC 22200313 GABRIELLY DE SOUZA PORTELA – UC 2220274 LUCAS BONIFÁCIO FIGUEREDO DOS SANTOS – UC 22421211 SEITI KLEFFER DE FREITAS ONO – UC 21426710 VÍTOR DOUGLAS DE BRITO SANTANA – UC 22102739
Data da Revisão:	29/03/2023

#	Critério	Atende? (Sim / Não)	Item com erro	Correções a Serem Realizadas
1	O código está documentado conforme os padrões da empresa ou grupo?	NÃO		Faltou documentar método de comunicação com o cliente. Não há documentação de instrução de como funciona o código e de como se deve rodar o código.
2	Os arquivos e classes estão com nomenclatura significativa?	SIM		
3	Os métodos estão com nomenclatura significativa?	SIM		
4	As variáveis e constantes estão com nomenclatura significativa?	SIM		
5	Todos os loop estão com condição de parada?	SIM		
6	Existe algum ponto de NullPointerException não tratado?	SIM	Class News	Validar parâmetro de entrada. O código só se preocupa com o Status de retorno da consulta e não se o retorno é nulo o que pode gerar erros nos loops de tratamento do Json.
7	Todas as variáveis estão inicializadas?	NÃO	Classe de serviço	Validar variáveis antes de chamar a requisição com cliente.
8	O código obedece à arquitetura definida para o software?	SIM		
9	Foi realizado o tratamento de exceções?	NÃO	Classe de serviço	Não em todos os casos possíveis. Somente no retorno negativo da consulta a API.

10	O software é fácil de usar e intuitivo?	SIM		
11	O software é testado em vários ambientes para garantir a compatibilidade?	SIM		
12	O software é compatível com outros softwares ou aplicativos?	SIM		