

Versão POO vs Versão Monolítica

A programação orientada a objetos sinaliza para uma maneira mais apropriada de realizar a correta divisão de um programa em conceitos (classes), permitindo que cada classe centralize em si todos os dados relativos ao conceito que ela representa, além de reunir todas as funcionalidades que pertencem a ela. Dado essa introdução de POO, é possível notar que esse paradigma é altamente recomendável quando há um time de desenvolvedores no mesmo código, com isso, ao comparar a versão monolítica com a versão POO implementado pela Professora da Universidade da Califórnia, é visível a facilidade que um time terá ao se deparar com o código modularizado em diferentes mini tasks, assim, um dev não se preocupa com o “Como essa tarefa será realizada”, e sim com “O que essa tarefa retornará para eu continuar o processo”. Como por exemplo, uma regra crítica desse algoritmo é desconsiderar artigos e preposições, apesar dessa regra influenciar diretamente no output do código, apenas um desenvolvedor (considerando que cada desenvolvedor é responsável por um objeto) terá de se preocupar com essa regra importante. Além disso, vale destacar que na versão monolítica a complexidade do algoritmo é algo a ser observado mais de perto, visto que existe uma série de laços encadeados no código.