



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO
TOCANTINS
CAMPUS PARAÍSO DO TOCANTINS
CURSO SUPERIOR BACHARELADO SISTEMAS DE INFORMAÇÃO**

Gabriel Pereira Veras

Atividade 08

Disciplina: Linguagens e técnicas de programação III

Aluno: Gabriel Pereira Veras

PARAÍSO DO TOCANTINS

2019

O que é PSR?

PSR é a recomendação de padrões PHP(PHP Standards Recommendation), PSR é um padrão que possui um conjunto de regras que devem ser seguidos por seus adeptos, ele nasceu com a idéia de criar um padrão universal de desenvolvimento, ele é o principal padrão usado atualmente. Esse documento engloba muitas regras, tem várias versões e por isso pode ser complicado de se colocar em prática com perfeição.

Ele é considerado um padrão a ser usado para ter um código 'de qualidade' digamos assim, os padrões PSR são 5, PSR-0, PSR-1, ..., PSR-4, esses padrões facilitam a reutilização do código bem como a manutenção dos sistemas que se utilizam do mesmo.

Vantagens:

Uma das vantagens da utilização de um padrão PSR é que em boa parte dos frameworks de PHP seguem o padrão, logo, entender o PSR ajudará o programador quando ele for iniciar em um framework, além de que, se a maioria dos programadores de PHP que desenvolvem códigos 'de qualidade' utilizam esse padrão, logo, se um programador compreende o padrão será muito mais fácil para este entender os códigos de outros profissionais.

Como descrito até aqui, o uso de um padrão como o PSR é muito útil para um desenvolvedor php, esse padrão é hoje muito utilizado e sempre que considerado necessário é criada uma nova versão dele, geralmente essa nova versão não substitui a versão anterior, como é visto em muitos programas, ao invés disso cada versão que é criada lhe é atribuído um número seguindo a sequência e o programador decide qual(is) dele(s) vai utilizar.

Particularidades e usabilidades:

PSR-0: namespaces devem ter a estrutura (*), cada namespace possui um namespace de nível superior, cada namespace pode ter quantos namespaces desejar, cada separador de namespaces é convertido para um separador de diretórios durante o carregamento dos arquivos, cada _ é convertido para um separador de diretórios durante o carregamento de arquivos, durante o carregamento o namespace passa a ter a extensão '.php', nome de vendors, namespaces e classes podem ter qualquer combinação de

letras maiúsculas e minúsculas.

PSR-1: Os arquivos devem usar apenas tags `<?php` e `<?='`, ou seja, todo código php deve estar escrito entre essas tags bem como ter seu fechamento, Os arquivos devem usar apenas UTF-8 para arquivos PHP sem BOM(Byte Order Mark), Os arquivos deveriam ou declarar símbolos(Classess, Funções, Constantes, etc.) ou causar outros efeitos, mas nunca fazer as duas coisas, Os namespaces e as classes devem seguir a PSR-0, As constantes devem ser declaradas sempre com maiúsculas, este padrão também é amplamente usado em diversas outras linguagens de programação e deve ser seguido por aqui também, Os métodos devem ser declarados em camelCase, padrão também utilizado em diversas linguagens de programação.

PSR-2: Arquivos somente PHP deve-se omitir a tag de fechamento, o código deve seguir a PSR-1, ao invés de usar tabs(tabulação) deve-se usar 4 espaços(usado normalmente por padrão na maioria das IDEs), não deve possuir excesso de linhas, por arquivo cerca de 120 linhas com até 80 caracteres cada, deve-se haver uma linha vazia após a declaração do namespace e uma após o bloco de declarações do use, as aberturas de chaves devem estar na próxima linha e o fechamento na próxima linha após o corpo, a visibilidade deve ser declarada em todos os atributos e métodos, palavras chave de controle devem haver dois espaços entre elas, a abertura de estruturas de controle devem estar na mesma linha que elas e o fechamento na próxima linha após o corpo, etc.

PSR-3: Trata-se da interface, trabalha com a idéia de cliente, esse não entrarei em detalhes porque é realmente muito complicado para encontrar material sobre esse padrão em português, então vou me limitar a isso e passar para o próximo padrão PSR.

PSR-4: É o padrão mais utilizado atualmente, permite a utilização do autoloader juntamente com o composer, autoloader permite que as bibliotecas e classes necessárias sejam importadas automaticamente, sem a necessidade de ficar fazendo include/require a cada classe que for necessária, caracteres alfabéticos podem ter qualquer combinação de maiúsculas e minúsculas em

quelquer namespace, etc.