

# Uma breve comparação entre o Adianti Framework e o ScriptCase para geração de código PHP

Ivan Augusto Alves da Rocha<sup>1</sup>, Tales Nereu Bogoni<sup>1</sup>

<sup>1</sup>Sistemas de Informação – Universidade do Estado de Mato Grosso – Campus Sinop  
Sinop – MT – Brasil

ivan.augusto@unemat.br, tales@unemat.br

**Abstract.** *The main purpose of this article is to make a comparison between two PHP code generators, to analyze the advantages and disadvantages of each one. Code generators are powerful tools used to increase the productivity of application development. The analyzes were made through the development of a small product registration and inventory system.*

**Resumo.** *Este artigo tem como objetivo principal fazer uma comparação entre dois geradores de códigos PHP, a fim de analisar as vantagens e desvantagens de cada um. Geradores de códigos são poderosas ferramentas utilizadas para aumentar a produtividade do desenvolvimento de aplicações. As análises foram feitas através do desenvolvimento de um pequeno sistema de cadastro de produtos e estoque.*

## 1. Introdução:

Geradores de códigos são poderosas ferramentas utilizadas para aumentar a produtividade do desenvolvimento de aplicações, economizando tempo e aumentando o desempenho dos desenvolvedores. Esses geradores podem trabalhar por linhas de comandos e códigos ou por meio de interfaces gráficas de modo interativo.

Segundo Herrington (2003), a utilização de geradores de códigos para o desenvolvimento de softwares traz enormes vantagens aos códigos gerados como qualidade, agilidade e consistência.

Este artigo tem como objetivo fazer uma breve comparação entre dois geradores de códigos PHP para desenvolvimento WEB e analisar suas respectivas vantagens e desvantagens. Neste estudo de caso foram comparados o Adianti Framework ©<sup>1</sup> e o ScriptCase ©<sup>2</sup>.

Para a comparação dos dois sistemas foi desenvolvido um pequeno sistema de cadastro de produtos para que fosse possível analisar dados objetivos e subjetivos sobre os dois ambientes, como por exemplo, quantidade de linhas de código geradas, tempo necessário para o desenvolvimento, facilidade de uso, entre outras métricas.

## 2. Adianti Framework

---

<sup>1</sup> <https://www.adianti.com.br/framework>

<sup>2</sup> <https://www.scriptcase.com.br/>

O Adianti Framework é um framework de arquitetura *open source* para criação de sistemas em PHP, criado em 2006, focado no desenvolvimento de aplicações de negócios. Desde o seu lançamento ele vem se consolidando como um framework especializado no desenvolvimento de sistemas de gestão (ERP), com diversas funcionalidades. (Dall'Oglio, 2018).

O Framework possui diversas características para facilitar o desenvolvimento, sendo as principais: pequeno; muito simples de utilizar; baseado em componentes de alto nível para construção de interfaces; oferece uma camada simples e ágil para acesso a dados; implementa padrões de projeto. Dificilmente vai ver código HTML ou SQL dentro de uma aplicação escrita com o Adianti Framework. (Dall'Oglio, 2018).

Segundo Dall'Oglio (2018), criador do Adianti, não houve a intenção de criar um Framework generalista, possibilitando o desenvolvimento de qualquer aplicação com ele, como um site por exemplo. Apesar de ser possível realizar tal atividade com o Adianti Framework, ele nasceu dentro de um contexto de desenvolvimento de sistemas de gestão, sistemas de informação, conhecidos como ERP's.

Diferente de outros Frameworks que levam o desenvolvedor a mesclar HTML e PHP em templates, no Adianti Framework a ideia é construir interfaces por meio de componentes. Caso você precise de uma funcionalidade visual muito específica, então poderá usar o mecanismo de templates fornecido pelo Framework, que permite utilizar HTML para compor uma interface.

O Adianti Framework privilegia a independência da aplicação em relação ao banco de dados, que permite desenvolver aplicações complexas sem escrever nenhuma linha de SQL, deixando esta tarefa para o framework. A mesma aplicação poderá rodar sobre diferentes bancos de dados sem necessidade de reescrita.

O Adianti Framework conta com diversas funcionalidades já prontas para o uso, funcionalidades essas que são comuns em todos os sistemas, como: Controle de login, controle de permissões, armazenamento de logs, alteração e exclusão de registros, bem como a comunicação entre usuários, e o compartilhamento de documentos e também o controle de permissão de acesso. Tais funcionalidades não estão implementadas diretamente no Framework, mas estão presentes nos templates, já prontas para o uso.

Em relação aos Bancos de dados, o Adianti Framework suporta PostgreSQL, MySQL, SQLite, Oracle e SQLServer.

### **3. Scriptcase**

O ScriptCase é uma plataforma de desenvolvimento de aplicações em PHP, com usuários do mundo todo e bem conhecida nos meios de desenvolvimento de sistemas. Com ele é possível fazer formulários de cadastro, gráficos customizáveis em tempo de execução, relatórios de gerenciamento, menus, autenticação de usuários, dashboards, calendários e muito mais. ScriptCase funciona através de um sistema de *Drag and Drop* (Arrastar e Soltar), que permite o usuário inserir os componentes apenas clicando e arrastando.

A plataforma suporta praticamente todos os tipos de bancos de dados: Oracle, SQL Server, MySQL, PostgreSQL, Access, Interbase/Firebird, Sybase, Informix e DB2, construindo aplicações utilizando bancos de dados múltiplos.

Todas as aplicações criadas no Scriptcase já possuem uma estrutura visual predefinida. Além da geração automática das telas, o Scriptcase permite que você personalize todos os objetos e estilos do HTML modificando o visual de seus sistemas. Scriptcase gera códigos-fonte (PHP, JavaScript, HTML e AJAX). O código gerado é totalmente independente da ferramenta e pode ser publicado em um servidor Web que possua o PHP habilitado. Da mesma forma do Adianti, o ScriptCase esconde todo seu código deixando apenas os componentes e configurações para a visualização.

A instalação é muito rápida e simples, já que o ScriptCase roda dentro de um navegador, isso possibilita que mais de uma pessoa possa trabalhar ao mesmo tempo com o mesmo ambiente de desenvolvimento através de redes locais ou na nuvem.

Uma grande vantagem é que possui diversas funcionalidades já prontas, com diferentes *templates* para usos gerais e específicos. Funcionalidades essas como: Sistema de login, menus, gráficos, geradores de PDF, grids, etc. Uma outra vantagem é a facilidade para a publicação do projeto ao usuário final, para a publicação é necessário apenas um servidor WEB com suporte a PHP para executar as aplicações.

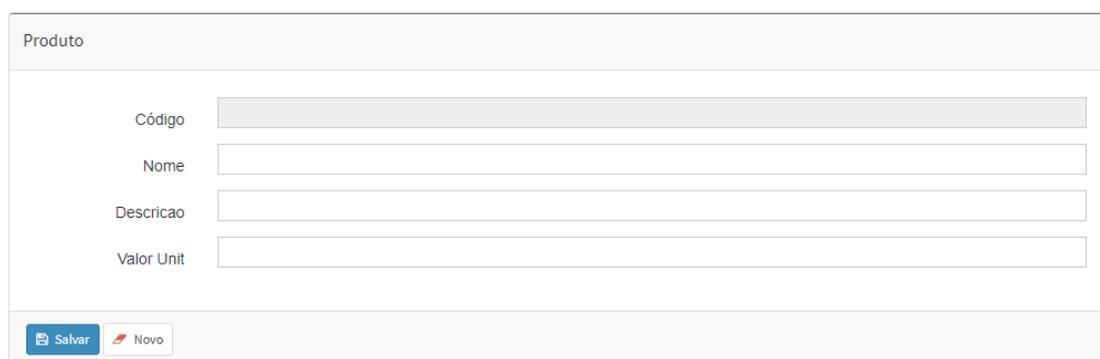
#### **4. Metodologia e Resultados**

Para fazer uma comparação entre os frameworks foi criado o mesmo sistema WEB de cadastros de produtos e estoque. Com isso foram coletados alguns dados objetivos, como quantidade de linhas de código geradas, tamanho do projeto, tamanho do framework e tempo gasto de desenvolvimento. E alguns dados subjetivos como facilidade de instalação, facilidade na conexão com o banco de dados e facilidade na criação e customização de formulários.

A instalação dos dois Frameworks foi muito rápida e simples. O ScriptCase ocupou cerca de 2GB de tamanho de disco na instalação e o Adianti Framework ocupou aproximadamente 170MB de tamanho de disco.

Com relação à conectividade com o banco de dados, ambos fazem de forma automática, é necessário apenas selecionar o banco de dados desejado e seu tipo que eles realizam a conexão com as tabelas com muita facilidade.

Para criar um formulário de cadastro de produtos no Adianti, apenas selecionamos a tabela do banco de dados referente a tela e selecionamos algumas opções desejadas de tipos de entradas (*Entry, Combobox, ...*) e então é gerado o código PHP e já é mostrado direto na tela do framework. O código é muito simples, bem otimizado e de fácil leitura, permitindo atualizações diretamente no arquivo gerado. A tela de cadastro de produtos com quatro campos de entrada gerou um código de 148 linhas todo comentado com explicações e instruções de como funciona, e com os métodos para salvar no Banco de dados e todos os componentes gráficos. A Figura 1 mostra o formulário gerado para o cadastro dos produtos.



**Figura 1. Formulário de cadastro de produtos do Adianti Framework**

Observou-se que nesta etapa não há muitas opções de modificações e personalização da tela além do *template* escolhido. Para ter uma maior personalização e inserção de mais componentes pode-se modificar diretamente o arquivo com o código PHP, permitindo maior controle do desenvolvedor sobre a aplicação gerada. A Figura 2 mostra um trecho de código gerado pelo Adianti Framework, que mostra um exemplo de código para validar e salvar o cadastro feito.

```

84
85     $this->form->validate(); // validate form data
86     $data = $this->form->getData(); // get form data as array
87
88     $object = new Estoque; // create an empty object
89     $object->fromArray( (array) $data); // load the object with data
90     $object->store(); // save the object
91

```

**Figura 2. Parte do código PHP gerado da tela de cadastro de produtos do Adianti Framework**

O mesmo formulário foi criado no ScriptCase afim de obter os resultados para a análise. Para criar o formulário bastam poucos cliques. Da mesma forma que com o Adianti Framework, selecionamos a tabela do banco de dados a qual queremos criar a conexão para persistência de dados e personalizamos os dados da conexão. Diferente do Adianti, com o Scriptcase o código PHP não é mostrado para edição, mas sim um conjunto de componentes e opções para a customização do formulário. Para alguém que esteja no primeiro contato com o ScriptCase sem um treinamento adequado pode parecer um pouco confuso, pela quantidade de opções jogadas na tela, o que torna difícil saber como lidar com a personalização da tela de cadastro. Ao final da montagem da tela o código PHP gerado chegou a 2934 linhas. A Figura 3 mostra o formulário de cadastro de produtos feito utilizando o Scriptcase.

**Figura 3. Tela de cadastro de produtos do ScriptCase**

O código gerado para a tela de cadastro de produtos do ScriptCase teve uma mistura de PHP, AJAX e HTML, embora seja um código grande e de difícil compreensão à primeira vista, ele funciona muito bem de uma forma organizada e sequencial, possui poucos comentários explicativos do código em comparação com o Adianti, porém com as funções em português, ler e entender o código se torna simples. A Figura 4 mostra um trecho de código gerado pelo ScriptCase, que mostra um exemplo de código para acessar o banco de dados.

```
2413 function nm_acessa_banco()  
2414 {  
2415     global $nm_form_submit, $teste_validade, $sc_where;  
2416  
2417     $NM_val_null = array();  
2418     $NM_val_form = array();  
2419     $this->sc_erro_insert = "";  
2420     $this->sc_erro_update = "";  
2421     $this->sc_erro_delete = "";  
2422     if (!empty($this->sc_force_zero))
```

**Figura 4. Parte do código gerado no ScriptCase**

A grande vantagem observada é que o usuário não precisa ser conhecedor de PHP para criar um sistema inteiro, já que o código só é gerado na hora de fazer a publicação do projeto. A quantidade de opções de personalização é grande, porém se for necessário fazer algo muito específico fica mais difícil.

Em relação aos outros formulários de cadastro e de consulta criados para o sistema de testes foi tudo bem parecido como descrito acima. Com alguns poucos cliques e as interfaces já eram geradas com a conexão do banco de dados. O tempo total para o desenvolvimento dos formulários durou cerca de uma hora em cada em framework, porém na hora da publicação final o ScriptCase foi mais lento levando aproximadamente 9 minutos para fazer a publicação local, enquanto no Adianti Framework foi instantâneo, uma vez que o código já é gerado quando cada formulário é gerado.

Com relação ao tamanho do código gerado, no Adianti Framework a aplicação final, com todos os formulários de cadastros e consultas ocupou um espaço em disco de

28MB, enquanto a aplicação final feita pelo ScriptCase ocupou cerca de 770MB, como mostra a Figura 5

Tamanho:	24,8 MB (26.034.559 bytes)	Tamanho:	739 MB (775.089.328 bytes)
Tamanho em disco:	28,0 MB (29.446.144 bytes)	Tamanho em disco:	770 MB (808.103.936 bytes)

**Figura 5. Tamanho em disco Adianti x ScriptCase**

## 5. Conclusões

Os dois geradores de códigos PHP são sem dúvidas excelentes para o desenvolvimento de sistemas, são ágeis, de fácil instalação e muito práticos. Mas com os testes efetuados e as análises feitas pudemos observar que é notável a diferença de desempenho e otimização de código entre os dois geradores de códigos. ScriptCase gerou muitos arquivos e muitas linhas de código para uma aplicação simples, ocupou muito mais espaço em disco. Além do código ser muito grande ele gera muitas linhas “inúteis”, o que tornam o código de difícil compreensão, impossibilitando a manutenção do sistema diretamente pelo arquivo de código, forçando o usuário a possuir licença de uso Scriptcase durante toda a vida útil do sistema.

O Adianti Framework gerou uma aplicação muito leve, bem otimizada e com poucos arquivos, sendo uma ótima opção para quem deseja ter uma aplicação fluída e que possa fazer modificações depois de publicada. Como os arquivos são gerados em PHP e totalmente orientados a objeto é necessário que o programador domine a linguagem para poder fazer manutenção, mas é totalmente factível diretamente nos arquivos PHP.

O ScriptCase não deixa de ser uma ótima ferramenta para desenvolvimento, mas é perceptível como o Adianti se sobressai em relação de otimização e desempenho, possuindo mais vantagens do que o ScriptCase. Como possíveis trabalhos futuros é desejado realizar comparações mais afundo sobre a codificação de cada um deles e também comparar com outros geradores de códigos PHP e fazer documentações já que quase não há nenhuma documentação sobre este tema disponível na *internet*.

## 6. Referências

DALL'OGGIO, Pablo, Adianti Framework para PHP / Pablo Dall'Oglio. Lajeado: Edição do autor, 2018. ISBN 978-85-914354-8-7

HERRINGTON, Jack. Code generation in action. Greenwich: Manning, 2003.