

# Arquitetura de aplicações web com AngularJS

João M. Fachinetto

# Sobre mim.

- Analista de sistemas na Westsoft;
- Microsoft Certified Specialist C#

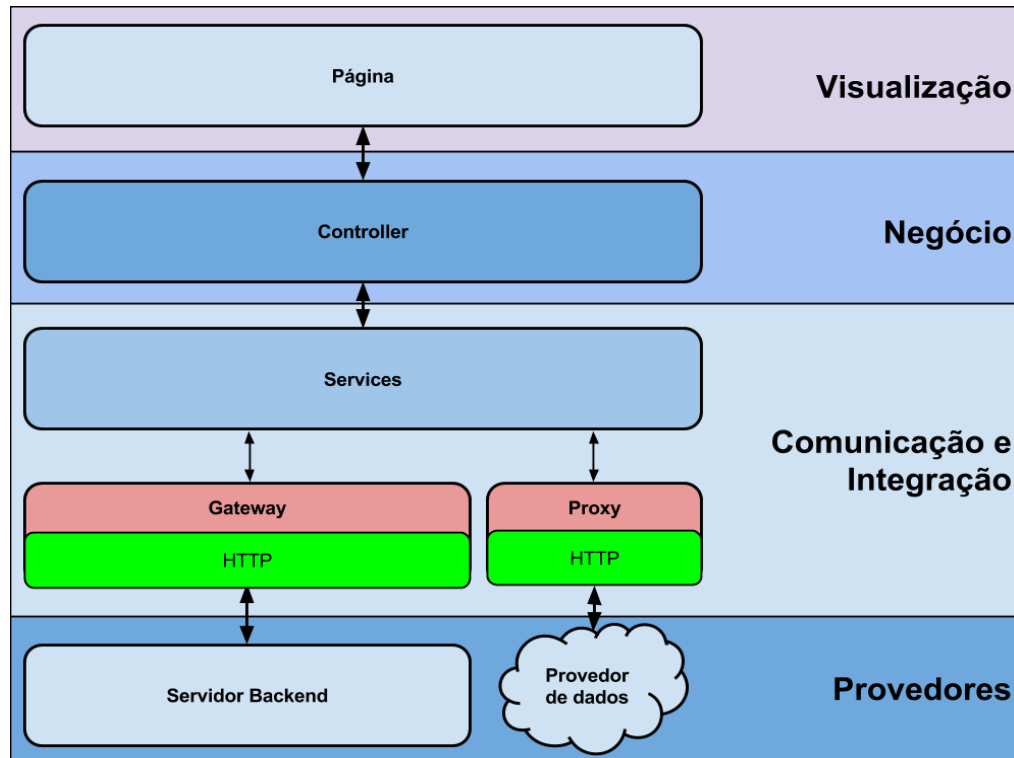
# Origem do modelo

- Precisa funcionar com o Sharepoint Online;
- Utilizar os Webservices do Sharepoint Online;
- Utilizando apenas Javascript.

# Como o modelo deve ser?

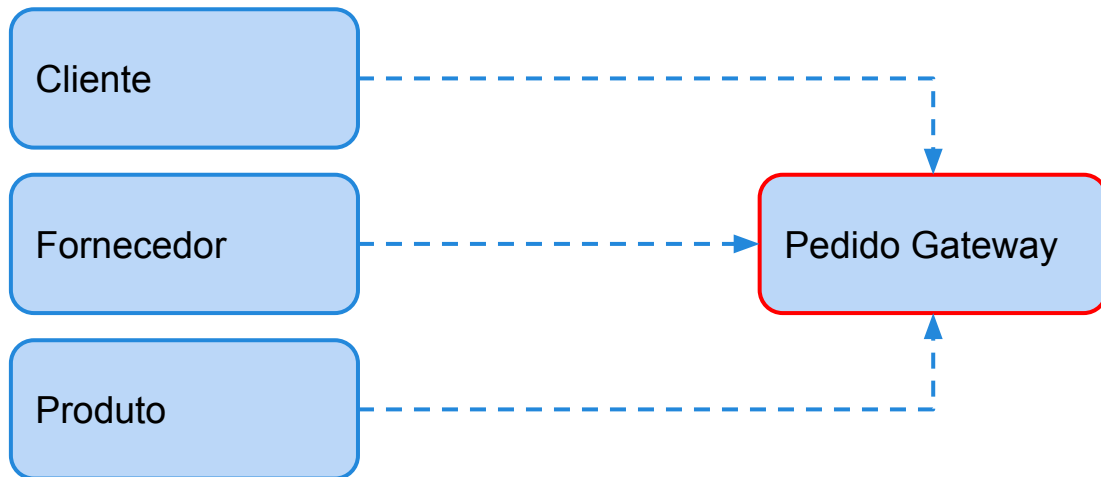
- Escalável;
- Adaptável;
- Reutilizável.

# Modelo proposto



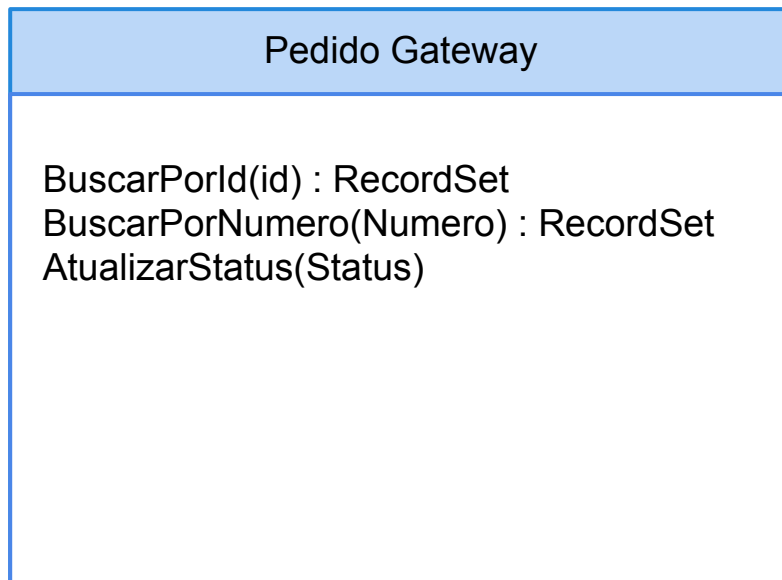
# Gateway

Segundo Martin Fowler um Gateway é um objeto que encapsula o acesso para um recurso externo.



# Table Data Gateway

Martin Fowler define um Table Data Gateway como um objeto Gateway que expõem métodos para manipular tabelas em um banco de dados.



# Proxy

O proxy é um recurso criado para buscar dados fora do domínio da aplicação. Por exemplo dados de endereçamento postal.

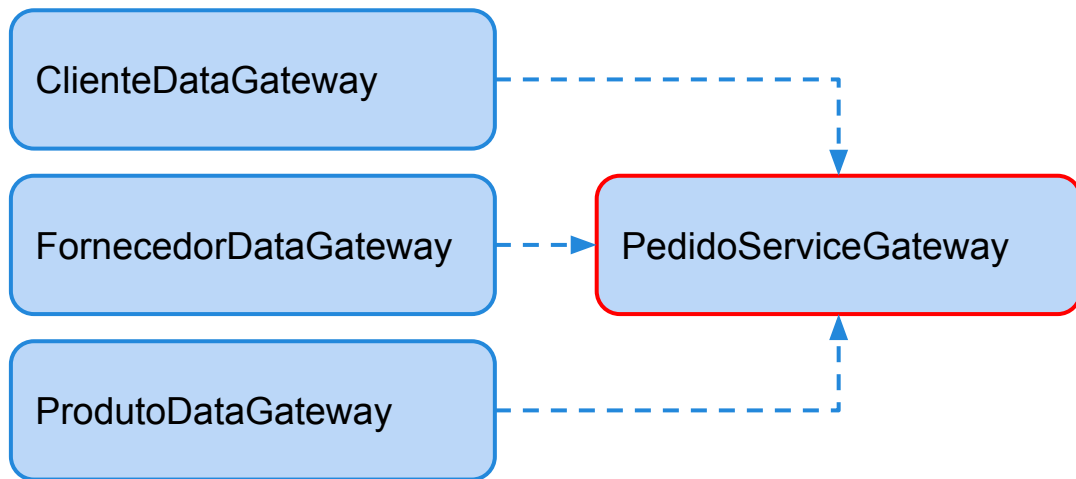




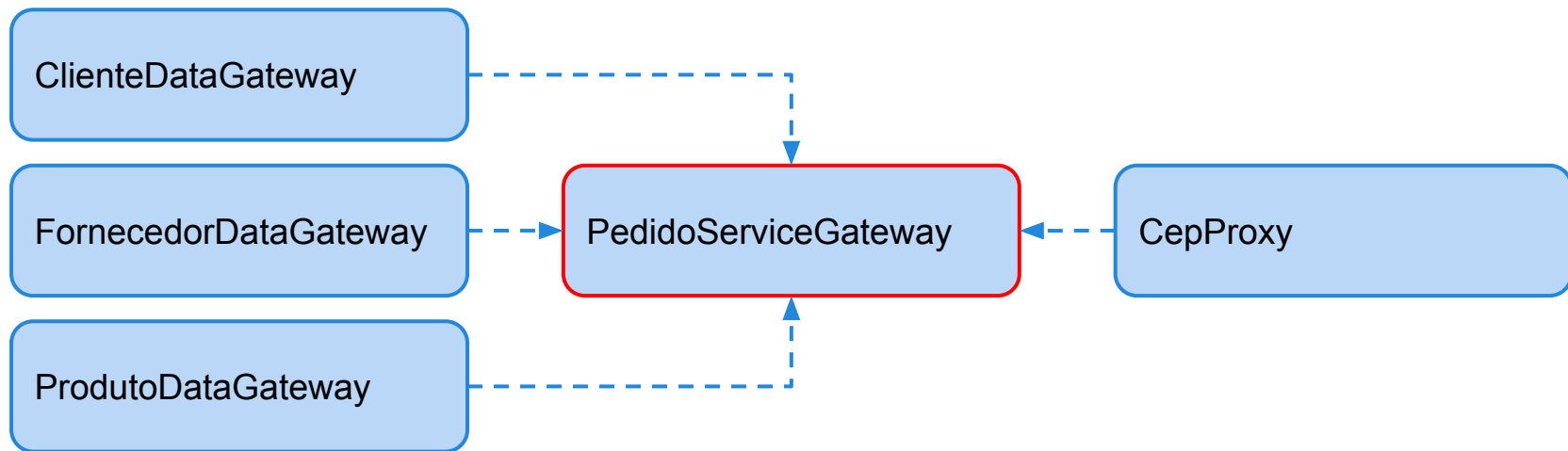
# Service Gateway

Principal função é fornecer recursos para o controller.

Vantagem:  
Singleton.



# Juntando tudo - Service Gateway



# Controller

O objetivo do controller é fornecer regras para a view e fazer o acesso ao service gateway.

# Controller

Um Controller não deve ser utilizado para:

- Manipular DOM;
- Formatar campos Input;
- Filtrar saídas;
- Compartilhar código com outros controllers;
- Gerenciar ciclo de vida de outros componentes.  
(Exemplo: Criar serviços)

# Controller

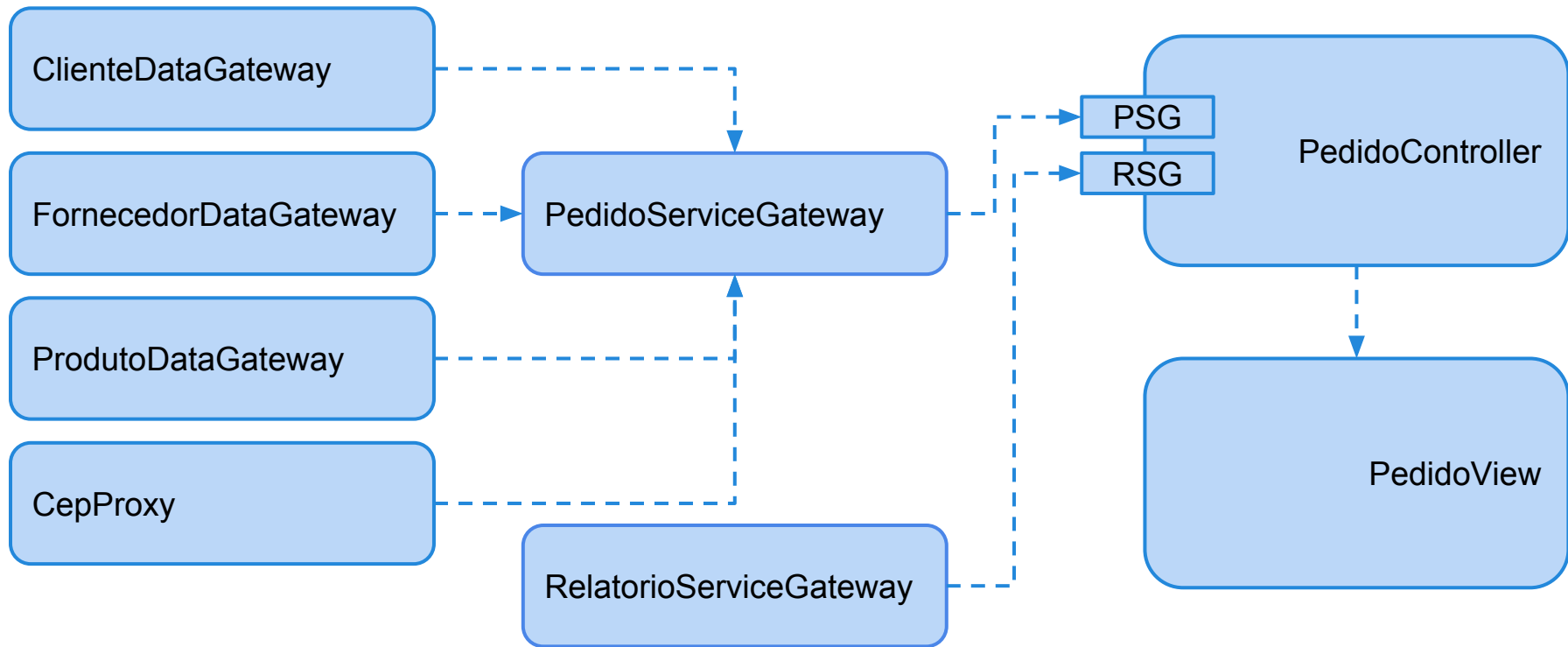
O ServiceGateway é adicionado ao controller através de injeção de dependência do AngularJs.

```
App.controller('CadastroDePedidoController', ['$scope', '$log', 'PSG',  
function($scope, $log, PSG) {...}
```

PedidoServiceGateway

\$log

# Juntando tudo



DEMO

# Obrigado

João Marcelo Fachinetto

Github: <https://github.com/expandit>

Twitter: @JoaoMFach