

# Técnicas Avançadas de Programação

## Padrões de Projeto – MVC

Arthur Andrade Gonçalves  
Assuerio Batista dos Santos  
Lucas Braga Ferreira  
Marcos Pablo Souza de Almeida  
Wendrya Andrade Oliveira

# Sobre

- Nome:  
MVC (Model View Controller)
- Classificação:  
Padrão de Projeto GRASP - Controller

## Como surgiu?

Final da década de 70, o funcionário da corporação Xerox PARC, Trygve Reenskaug, descreveu no artigo *"Applications Programming in Smalltalk-80: How to use Model-View-Controller"*, sendo aí o início deste padrão de projeto.



Trygve Reenskaug

# Problema a ser Resolvido

- Como isolar as regras de negócios da lógica de apresentação, a interface com o usuário? Quem será o responsável por lidar com os eventos de entrada?

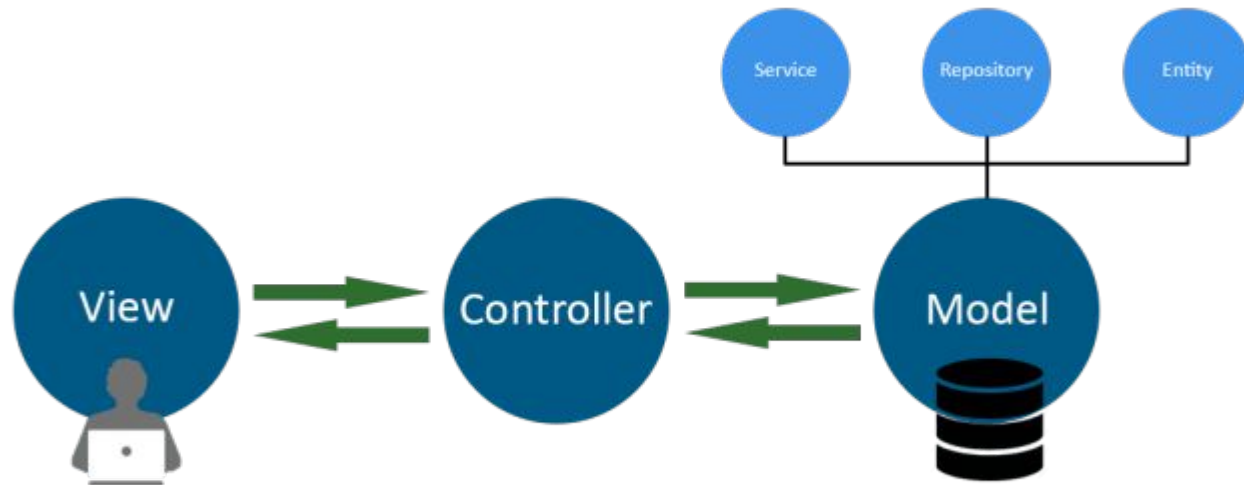
# Solução

Atribuir responsabilidades para receber ou lidar com um evento do sistema :

- Uma classe que representa todo o sistema ou subsistema
- Uma classe que representa cenário de caso de uso (controlador de caso de uso ou de sessão)

# Consequência

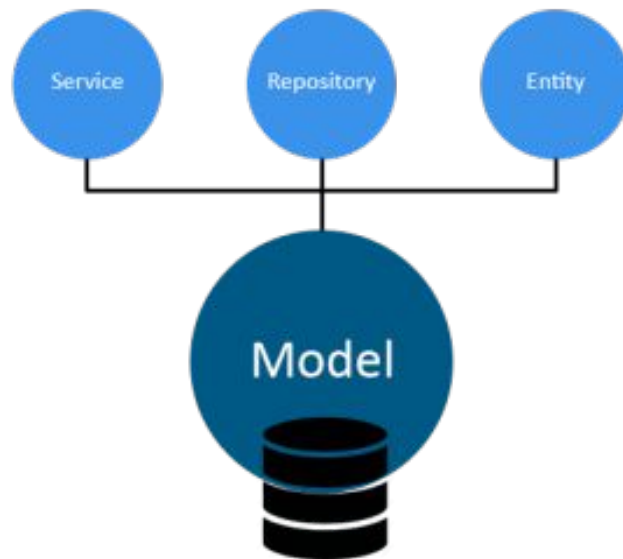
- Diminui a sensibilidade da camada de apresentação em relação à lógica de domínio
- O problema mais comum é ter poucos controladores (controladores sobrecarregados).



# Estrutura MVC

# Estrutura (MODEL)

- Gerencia um ou mais elementos de dados, responde sobre o seu estado e responde aos comandos para mudar de estado. Modela os dados e o comportamento por trás do processo de negócios.



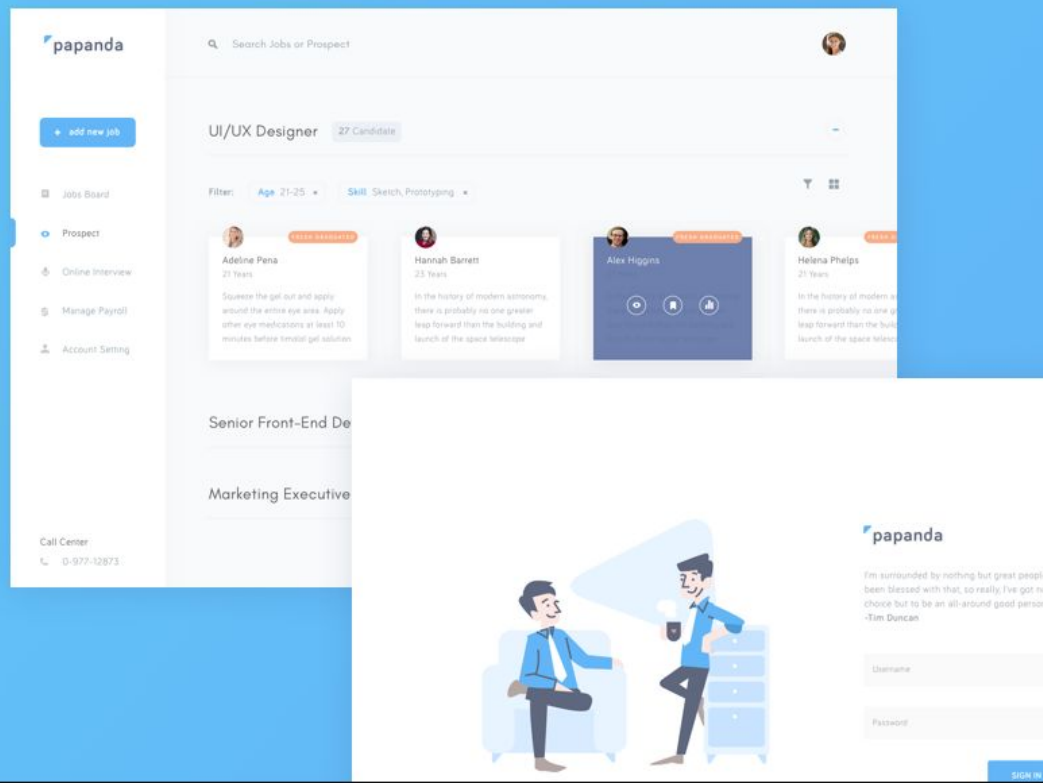
# Estrutura (VIEW)

- Gerencia a área que é responsável por apresentar as informações para o usuário. É por onde o usuário se comunica com o sistema e o que está visível para ele.
- Seu papel é receber instruções do *controller* e informações do *model* e então exibir elas.





# Estrutura (VIEW)



# Estrutura (CONTROLLER)WENDRYA DISGRASS

- Interpreta as ações (entradas do mouse ou do teclado) enviado pelo usuário, através da *view* e as mapeia em comandos que são enviados para a *model*.
- É o intermédio entre a *view* e a *model*.



# O diálogo das camadas WENDRYA DISGRASS

**View:** Fala Controller! O usuário acabou de pedir para acessar o Facebook! Pega os dados de login dele aí.

**Controller:** Beleza. Já te mando a resposta. Aí Model, meu parceiro, toma esses dados de login e verifica se ele loga.

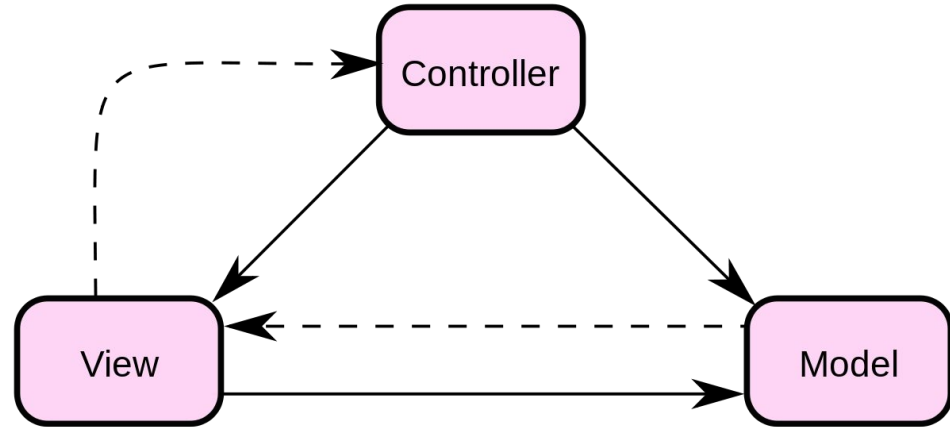
**Model:** Os dados são válidos. Mandando a resposta de login.

**Controller:** Beleza. View, o usuário informou os dados corretos. Vou mandar pra você os dados dele e você carrega a página de perfil.

**View:** Valeu. Mostrando ao usuário...

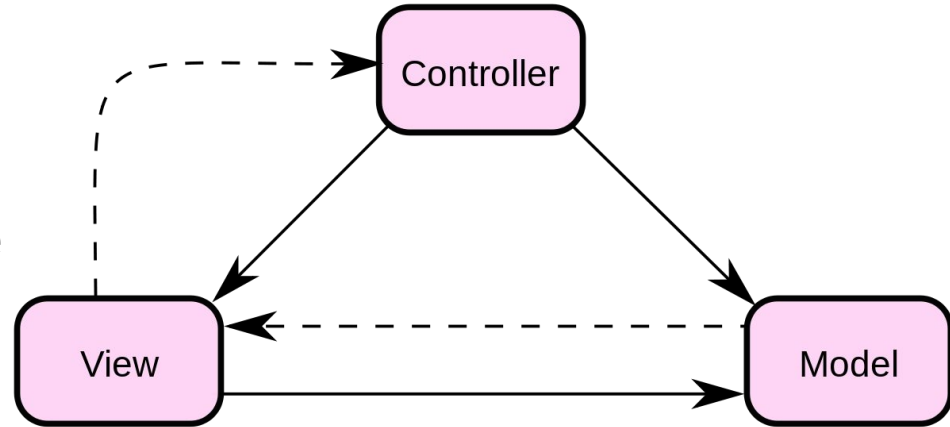
# Vantagens

- Desacoplamento e redução da interdependência entre as camadas
- Aumento de produtividade pois permite desenvolvimento em paralelo;
- Uniformidade na estrutura do software;
- Redução de complexidade no código;



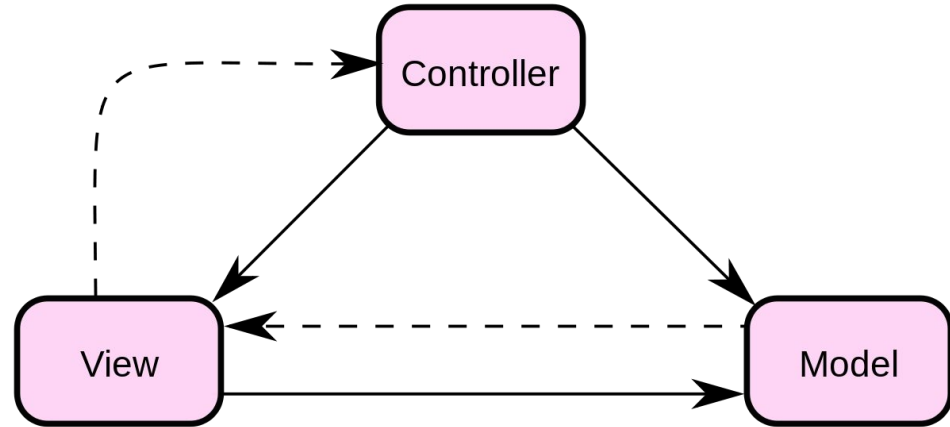
# Vantagens

- Separação entre interfaces e regras de negócios
- Permite a reutilização de código e módulos
- Ajuda a construir softwares confiáveis e seguros



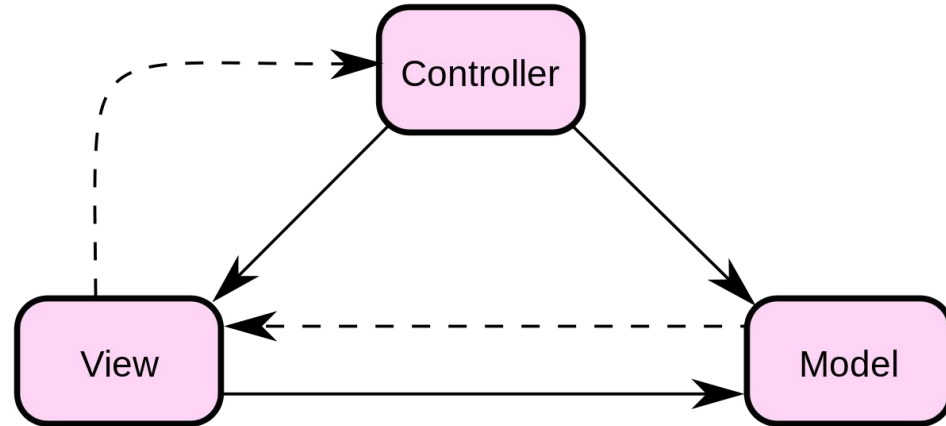
# Desvantagens

- Requer mão de Obra especializada
- Demanda um maior tempo para analisar o sistema e modelá-lo com o padrão



# Desvantagens

- A medida que o tamanho e a complexidade do projeto crescem, a quantidade de arquivos e pastas cresce também
- Não é recomendado em sistemas de pequeno porte



# Frameworks Baseados em MVC



CODEIGNITER



Laravel



CakePHP



# Frameworks Baseados em MVC



Symfony



# Referências

MACORATTI, José. **Padrões de Projeto: O modelo MVC - Model View Controller**. 2020. Disponível em: [http://www.macoratti.net/vbn\\_mvc.htm](http://www.macoratti.net/vbn_mvc.htm). Acesso em: 15 maio 2020.

SOUZA, Marcio. **Padrão MVC - Java Magazine**. 2020. Disponível em: <https://www.devmedia.com.br/padrao-mvc-java-magazine/21995>. Acesso em: 14 maio 2020.

WIKIPEDIA. **Padrões de projeto de Software**. 2020. Disponível em: [https://pt.wikipedia.org/wiki/Padr%C3%A3o\\_de\\_projeto\\_de\\_software](https://pt.wikipedia.org/wiki/Padr%C3%A3o_de_projeto_de_software). Acesso em: 16 maio 2020

MASSARI, Jorge. **Padrão MVC | Arquitetura Model-View-Controller**. Disponível em: <https://www.portalgsti.com.br/2017/08/padrao-mvc-arquitetura-model-view-controller.html>. Acesso em: 17 maio 2020.