

# Padrão MVC

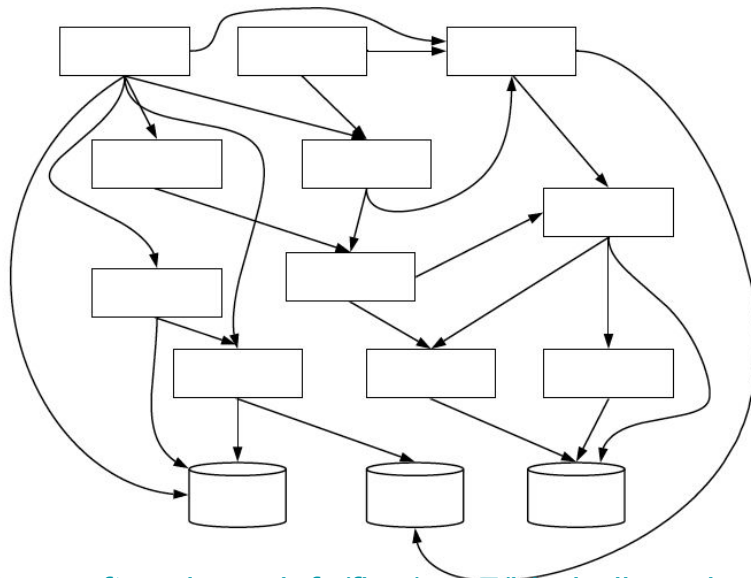
Lilian Passos Scatalon  
[lpscatalon2@uem.br](mailto:lpscatalon2@uem.br)

# Arquitetura de Software

- Consiste em identificar subsistemas
- Considera os elementos do projeto em uma visão mais abrangente
- Determina como as classes ficam organizadas em unidades maiores (p. ex. pacotes, módulos)

# Big Ball of Mud

- Um módulo pode usar praticamente qualquer outro módulo do sistema; ou seja, sistema é uma "bagunça"



<https://engsoftmoderna.info/figs/cap7/big-ball-mud.svg>

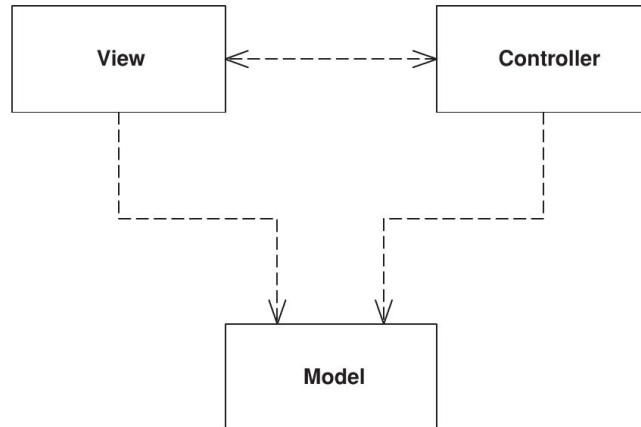
# Estilos/Padrões Arquiteturais

- Definem um modelo de organização estrutural para um sistema de software
- Consistem na descrição de componentes e de como estão conectados
- Exemplos:
  - Camadas, MVC, Cliente/Servidor, Microserviços, etc

# Padrão MVC (*Model View Controller*)

Organiza as classes em 3 grupos: modelo, visão, controlador

Surgiu no final da década de 70 com Smalltalk para o projeto de interface gráfica

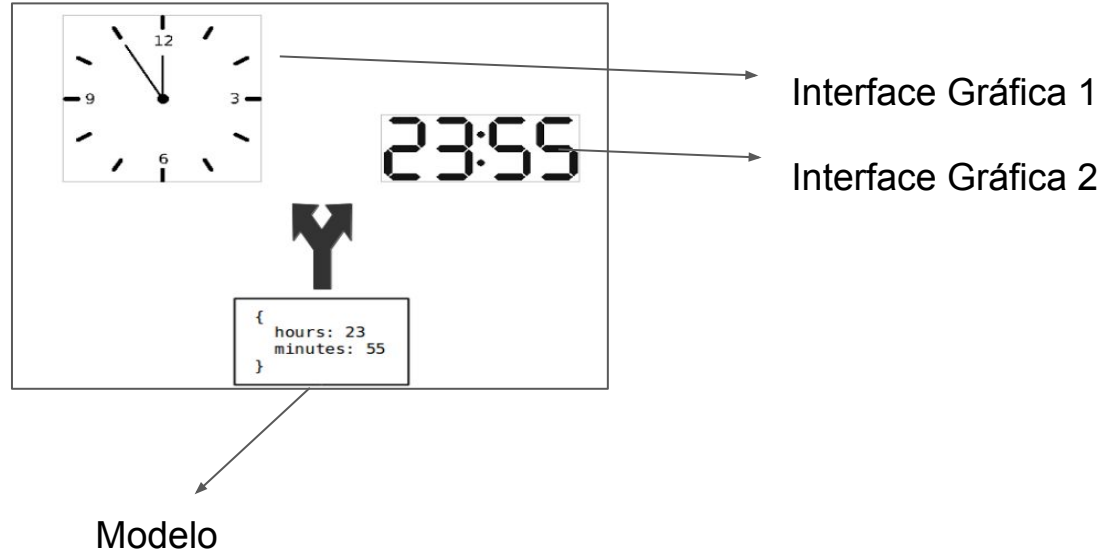


# Padrão MVC

- **Visão:** classes dos componentes GUI
- **Controlador:** classes que fazem a mediação entre visão e modelo
- **Modelo:** classes que representam os conceitos do domínio, derivados do processo de análise

# Padrão MVC

## Separação entre visão e modelo



<https://engsoftmoderna.info/figs/cap7/clock.png>

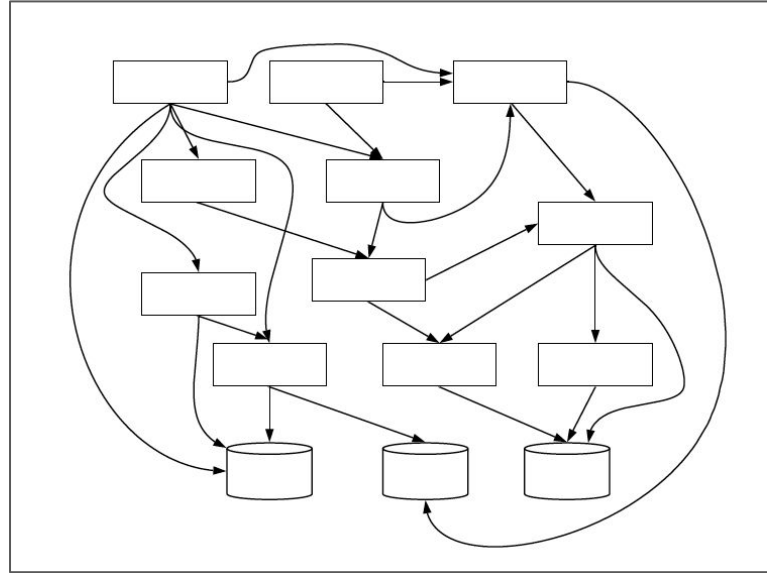
# Vantagens

Organização das classes de acordo com as responsabilidades

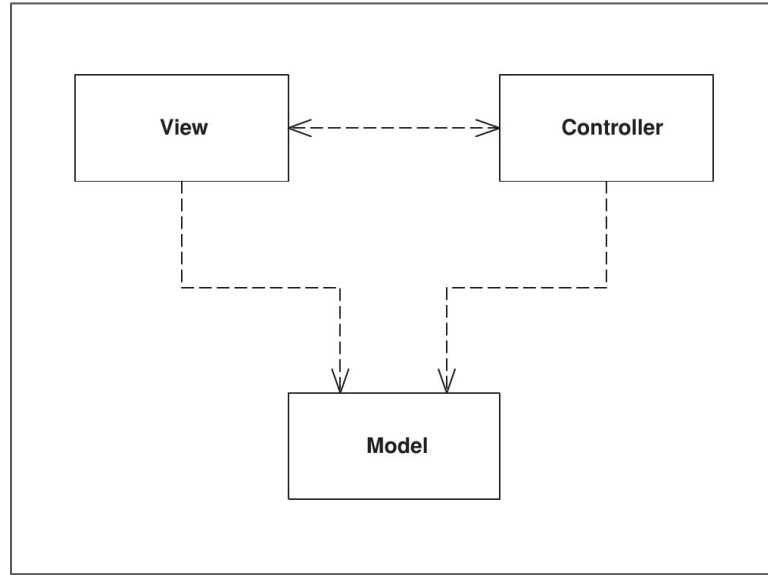
Reuso dos elementos de modelo

Facilidade para substituir elementos de interface





Sistema



Sistema

# Padrão MVC

Envolve a combinação de dois padrões de projeto:

- Controlador (GRASP)
- Observador (GoF)

# Bibliografia

Martin Fowler. **Patterns of Enterprise Application Architecture**, Addison-Wesley, 2002.

Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. **Design Patterns: Elements of Reusable Object-Oriented Software**. Addison-Wesley, 1995.

Marco Tulio Valente. **Engenharia de Software Moderna**. 1ª edição, 2022.