

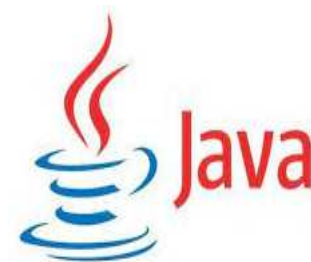


Residência em Software

Residência em Tecnologia da Informação e Comunicação SpringBoot - Introdução

Professor:

Alvaro Degas Coelho



INSTITUIÇÃO EXECUTORA



COORDENADORA

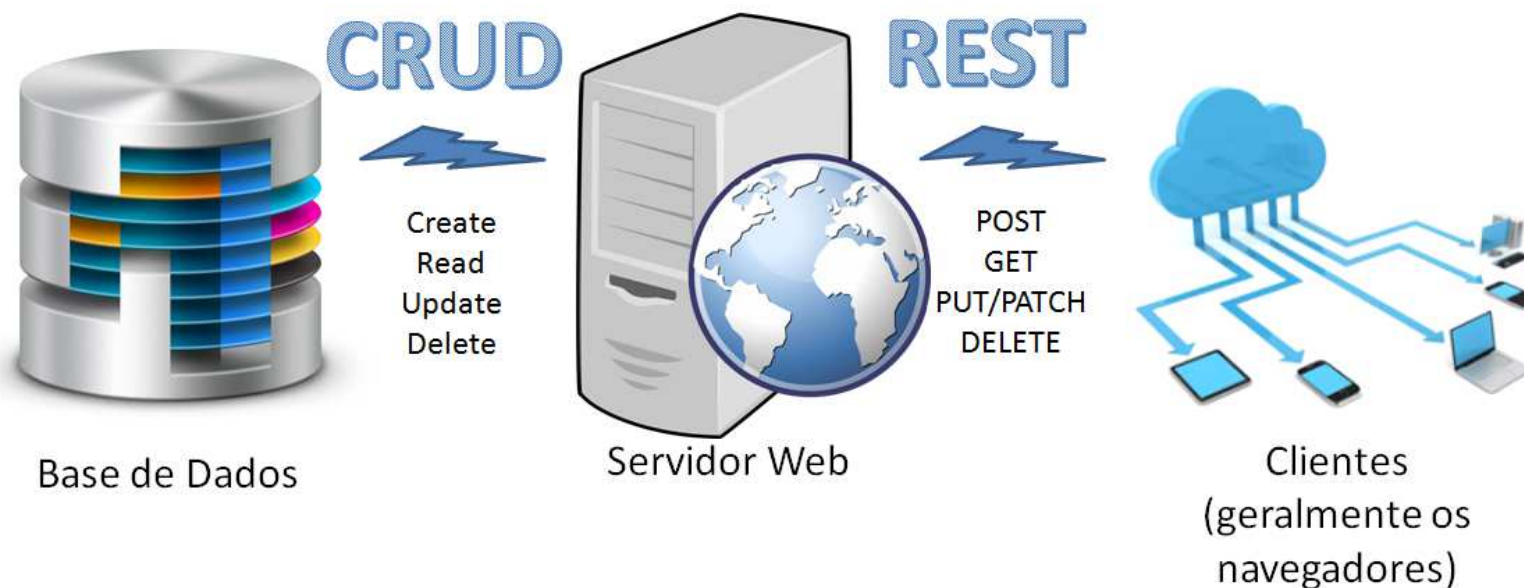


APOIO



O que é SpringBoot

- Começemos por uma imagem



SpringBoot

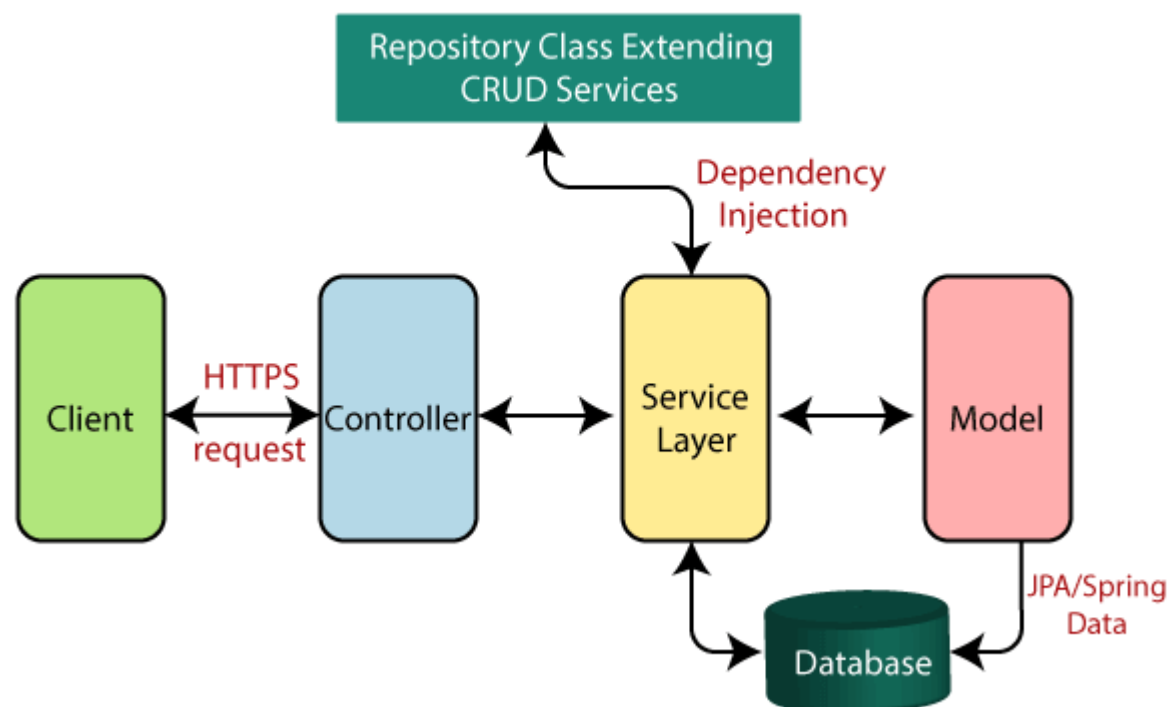
- Framework para desenvolvimento de aplicações WEB (e microserviços)
- Baseado em *dependências*
 - Um pedaço de software precisa de outro para funcionar
 - Eles se encontram e fazem acontecer

O que faz o SpringBoot

- Em linhas gerais
 - ligação de dados
 - Conectar dados desde a camada de interface com o usuário até o dado persistido
 - conversão de tipo
 - Principalmente aspectos ORM
 - Validação
 - Verificação de operações e dados
 - gerenciamento de exceções
 - Erros padrão HTTP e/ou definidos pela aplicação
 - ...

Arquitetura Típica

Spring Boot flow architecture



Por exemplo

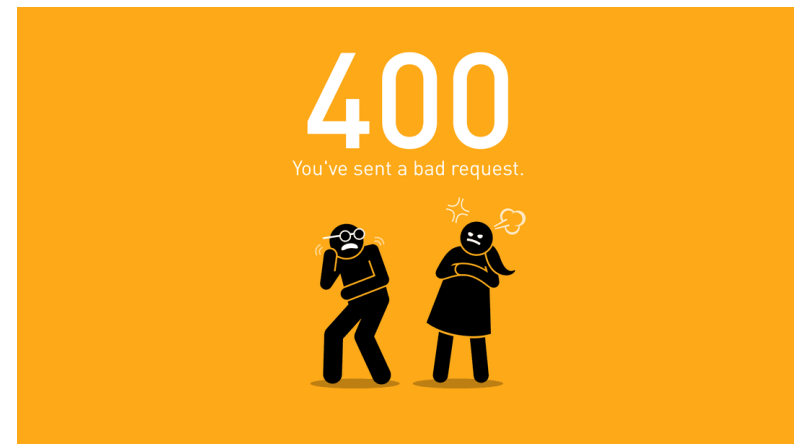
- No seu navegador, um estudante envia os dados de sua solicitação de matrícula
 - Lista de turmas de disciplinas (código)
 - Verbo POST do HTTP (veremos)
 - Endereço (Endpoint) correto
 - Algo como `sistemaacademico.matricula.solicitacao`
- O que deve acontecer?

O caminho a ser seguido

- Assumindo que os aspectos de segurança (autenticação) estão cumpridos
- A solicitação de matrícula precisa ser avaliada
 - Há alguma sobreposição de horário?
 - Deve ser recusada
 - Há alguma disciplina (note que temos apenas o código d turma) cujo pré-requisito não está cumprido?
 - Deve ser recusada

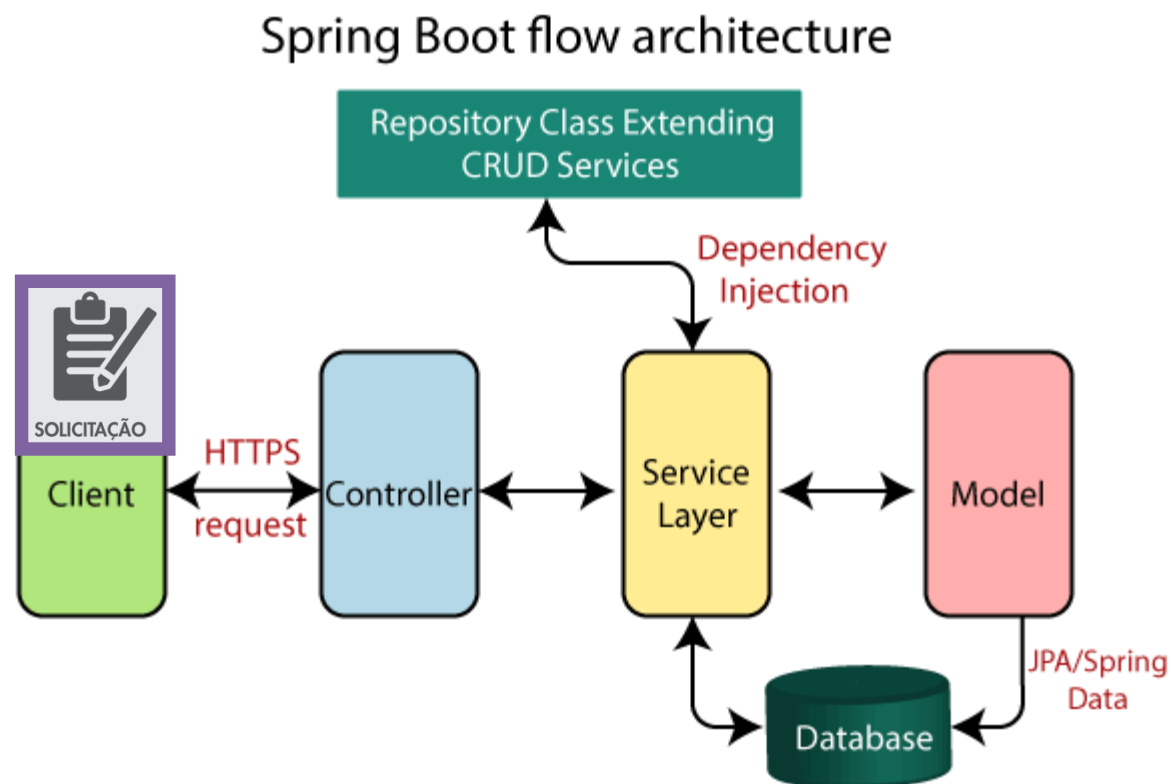
Primeira etapa

- A solicitação chega a algum ponto (listener) do sistema
 - EndPoint
- Por hipótese: arquivo JSON
- Se o endpoint solicitado não for o correto, volta um erro http
 - 400 (provavelmente)



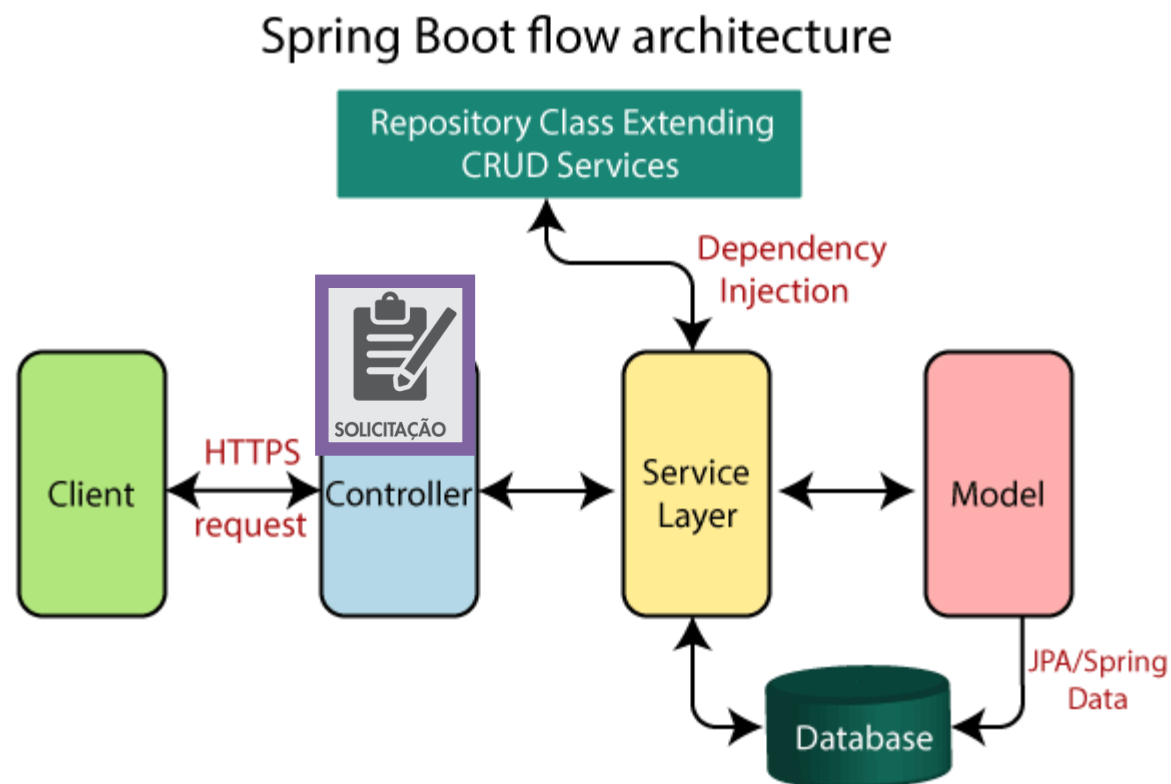
Solicitação recebida

- Supondo que a solicitação alcançou o endpoint correto



Solicitação recebida

- Supondo que a solicitação alcançou o endpoint correto

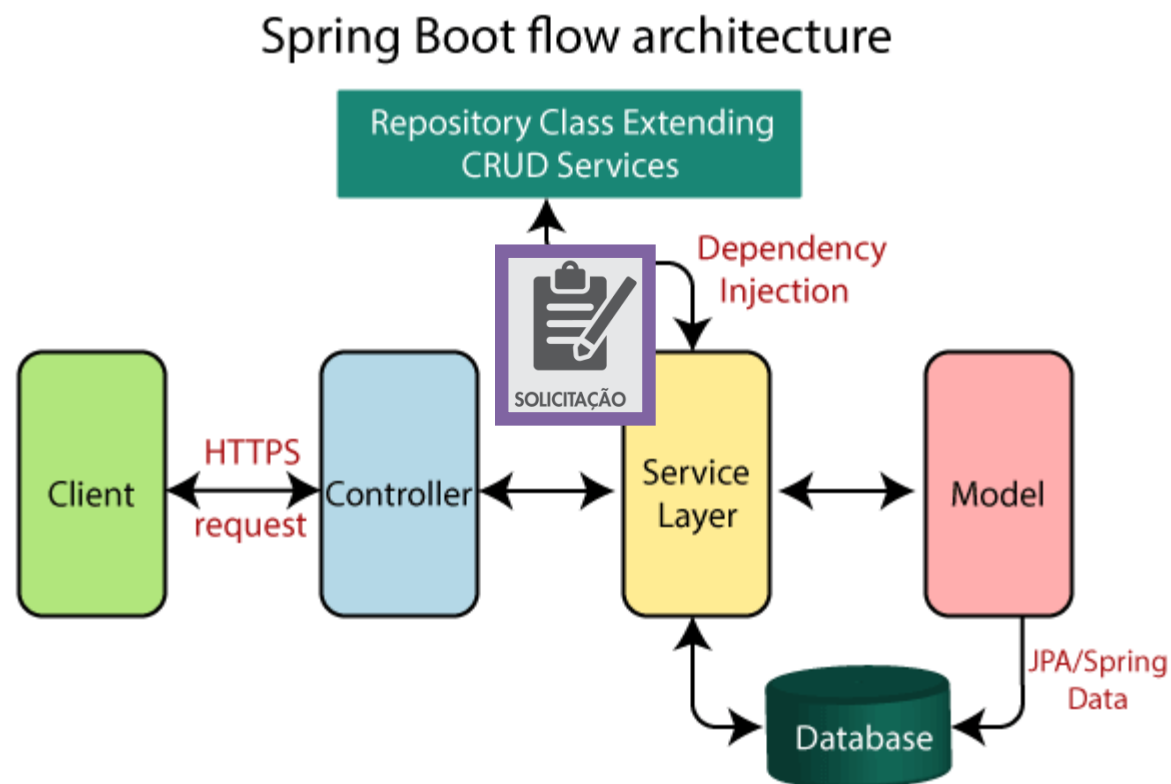


É necessário iniciar as verificações

- Verificação de choque de horários
- Comparar
 - Horários das disciplinas solicitadas

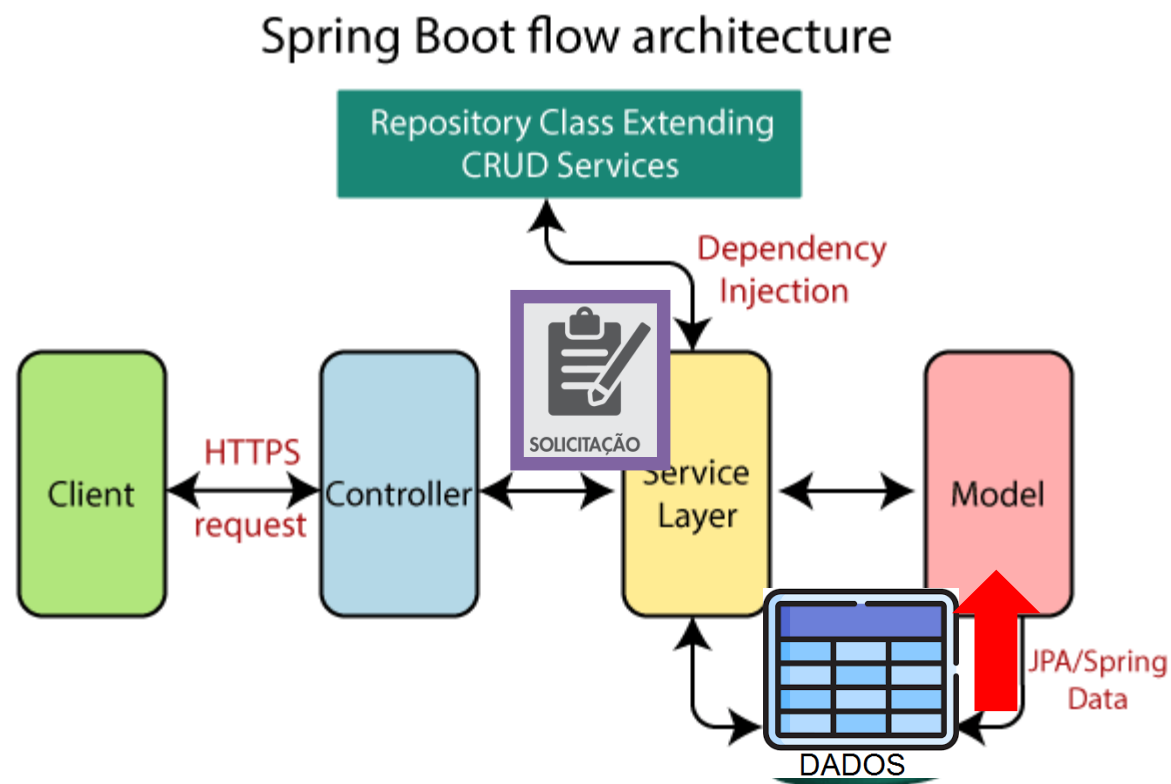
Solicitação recebida

- Supondo que a solicitação alcançou o endpoint correto



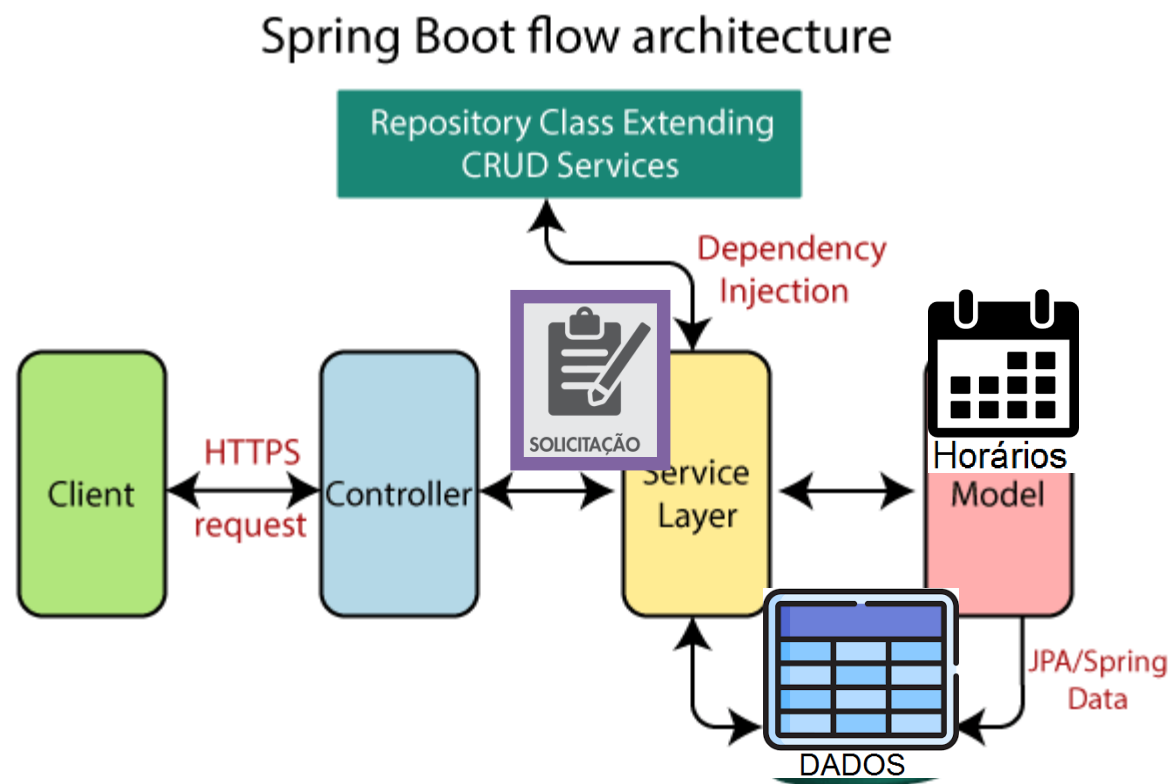
Solicitação recebida

- Verificação de choque de horário



Solicitação recebida

- Verificação de choque de horário

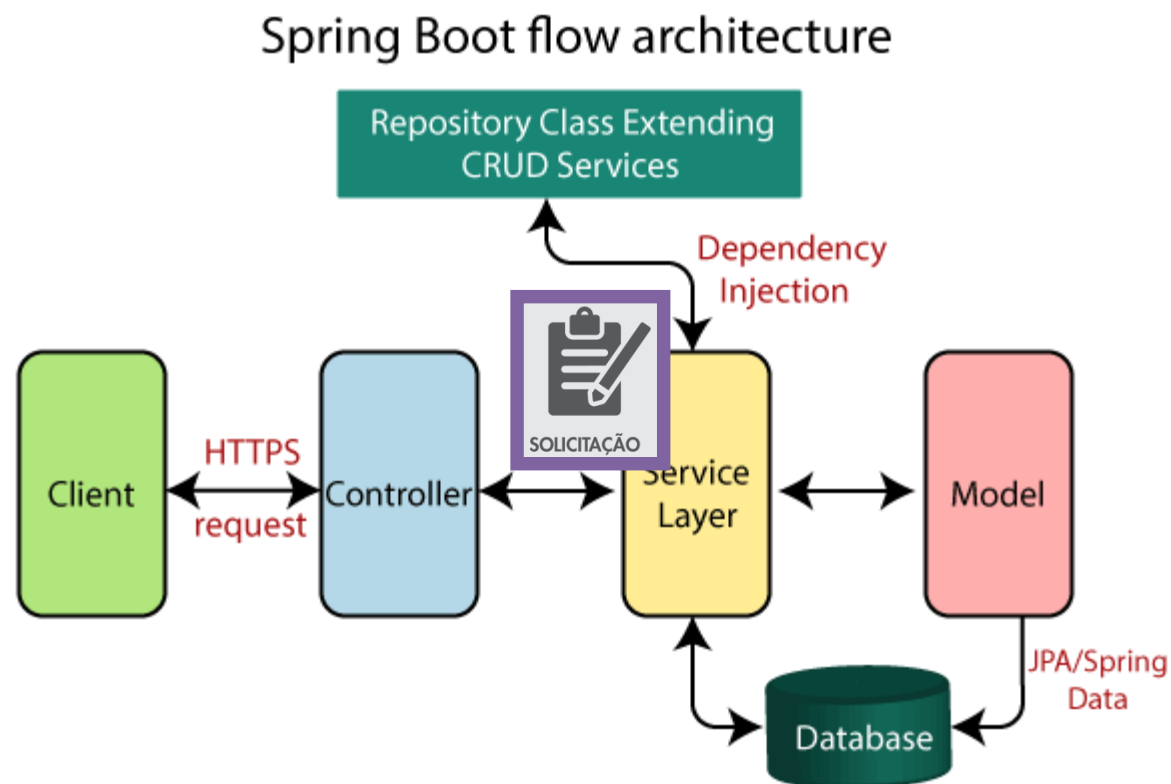


Continuando

- É necessário checar pré-requisitos
- Comparar
 - Lista de pré-requisitos das disciplinas solicitadas
 - Lista de disciplinas já cursadas pelo estudante

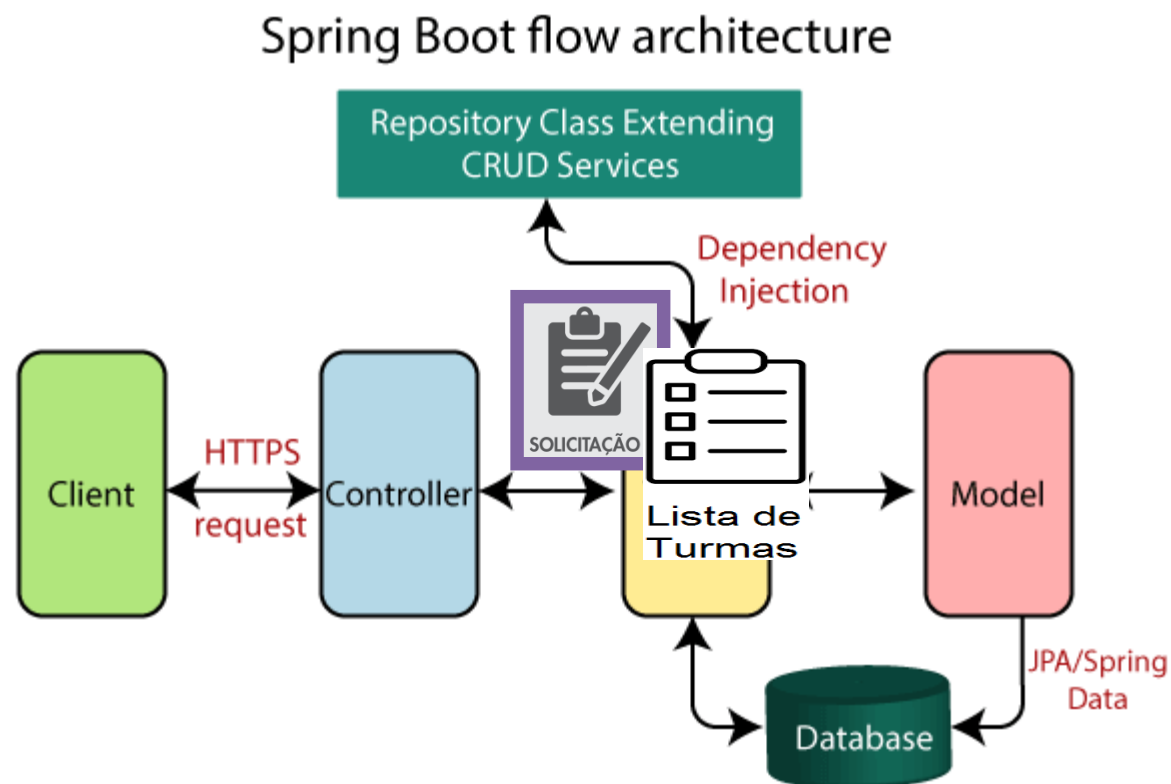
Solicitação recebida

- Verificação de Pre-requisitos



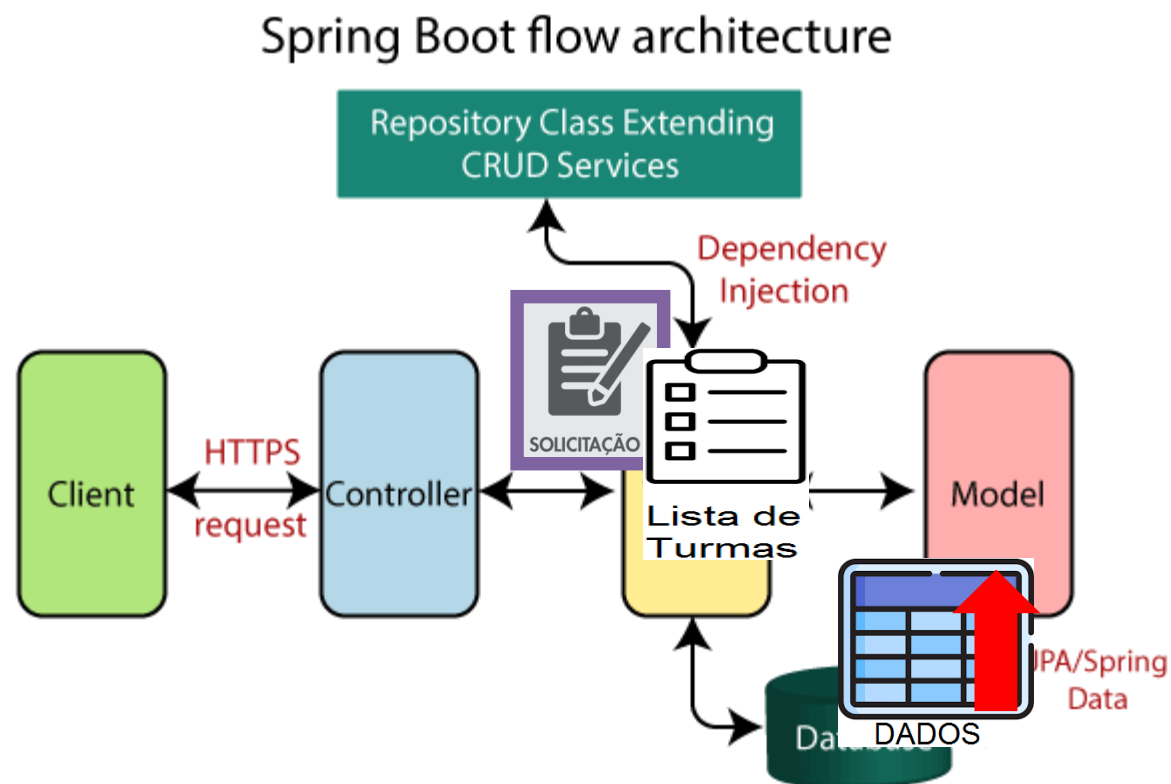
Solicitação recebida

- Verificação de Pre-requisitos



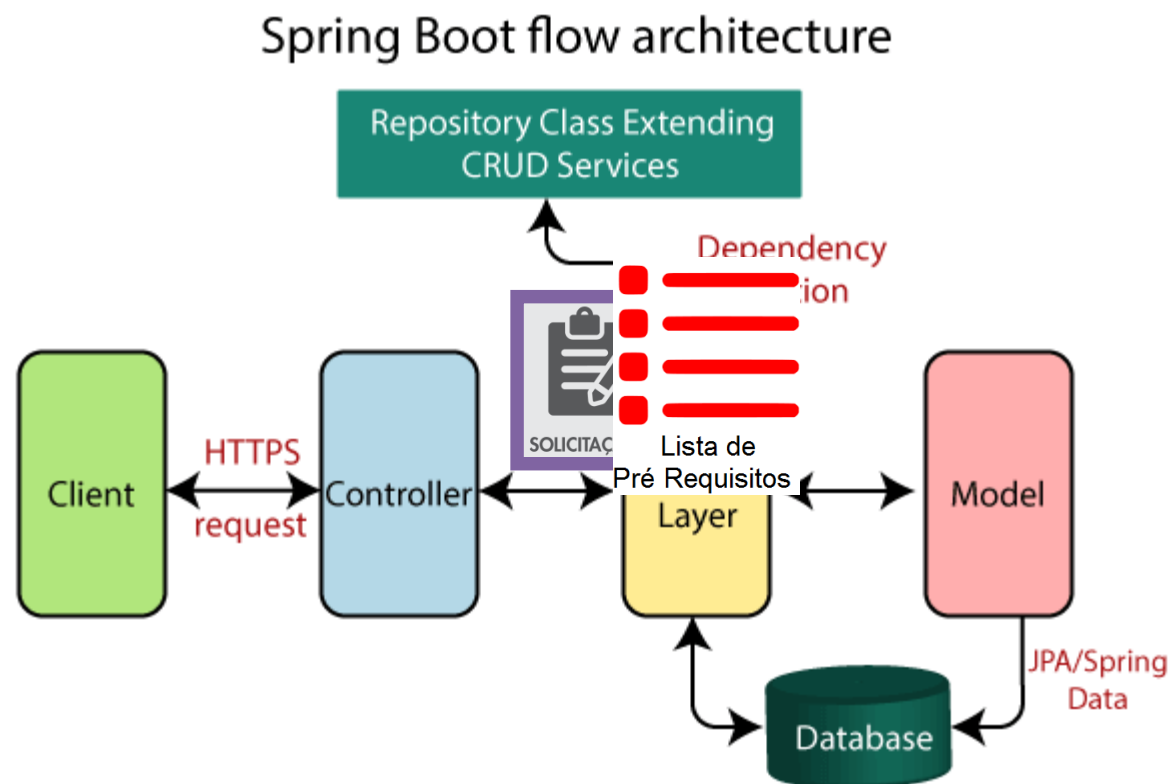
Solicitação recebida

- Verificação de Pre-requisitos



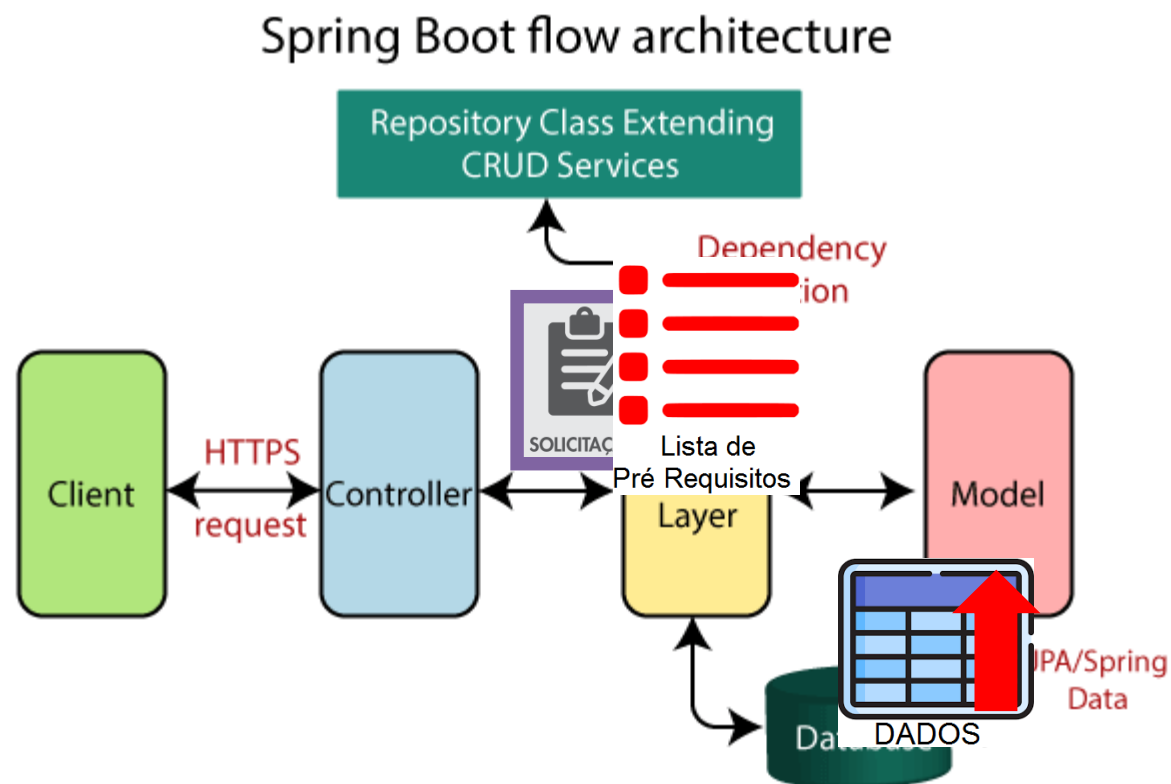
Solicitação recebida

- Verificação de Pre-requisitos



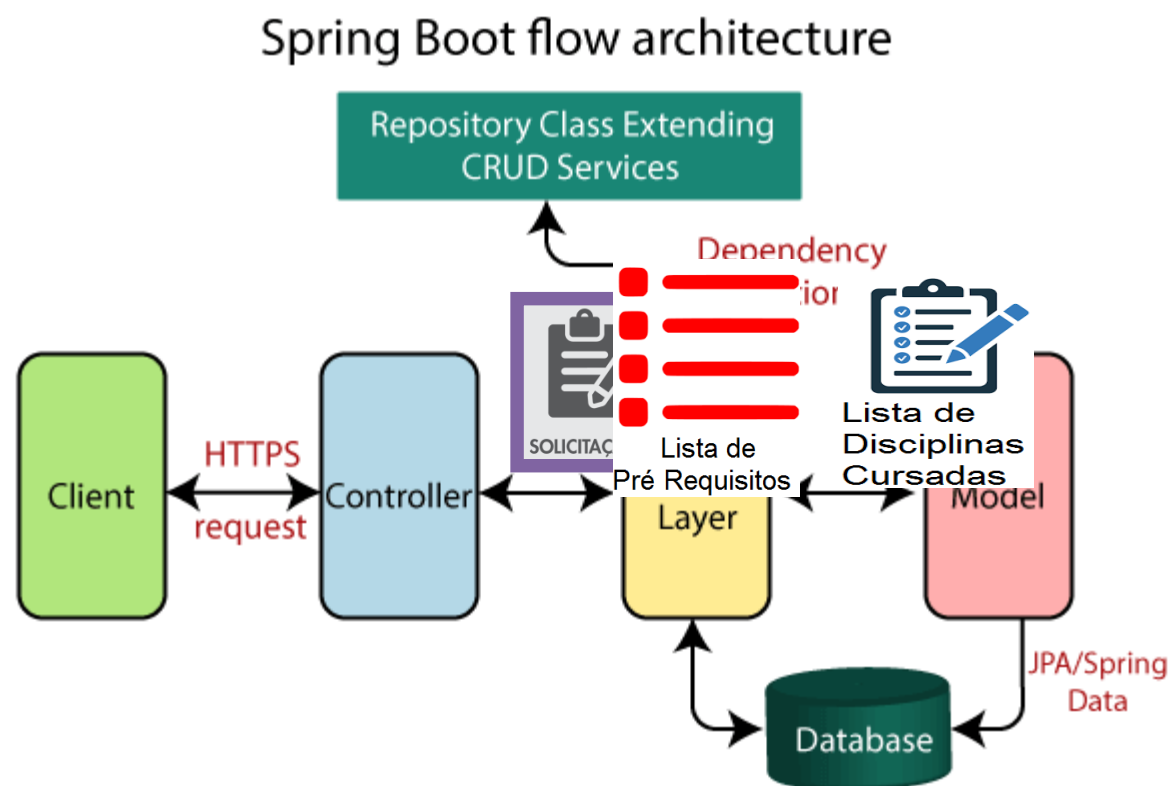
Solicitação recebida

- Verificação de Pre-requisitos



Solicitação recebida

- Verificação de Pre-requisitos

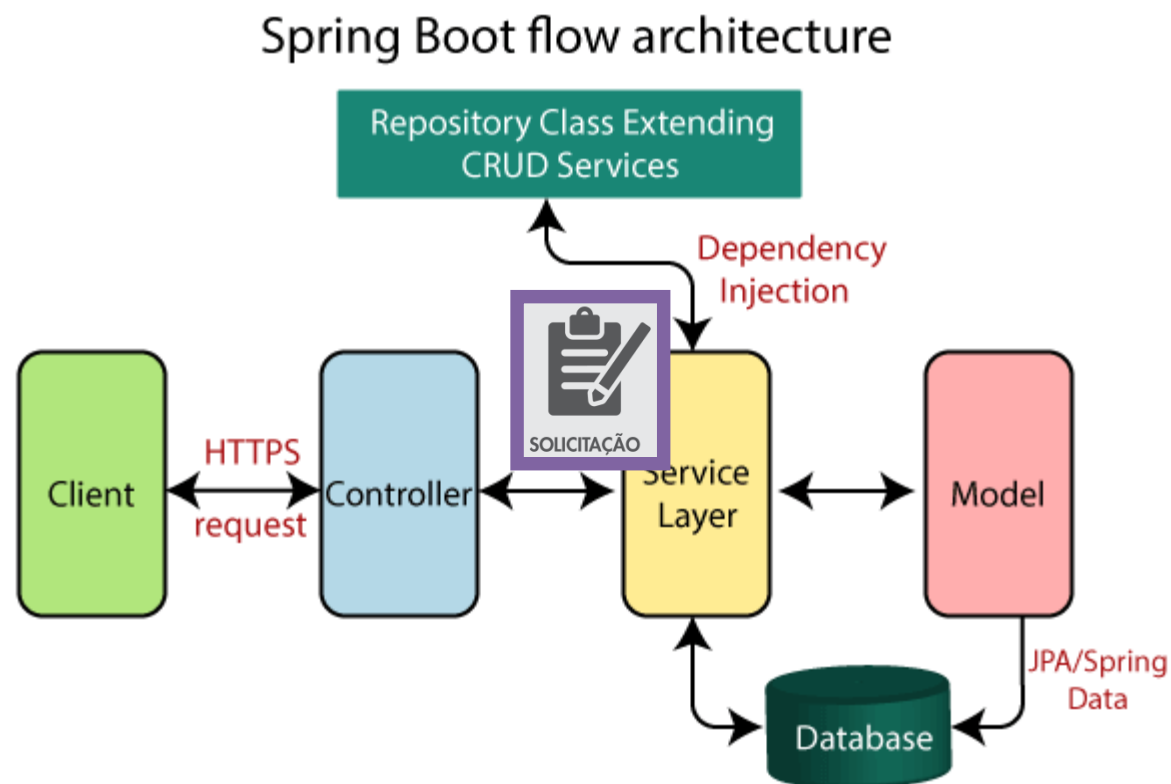


Finalmente

- A solicitação está perfeita
- Pode ser registrada
 - Ou seja: persistida no BD

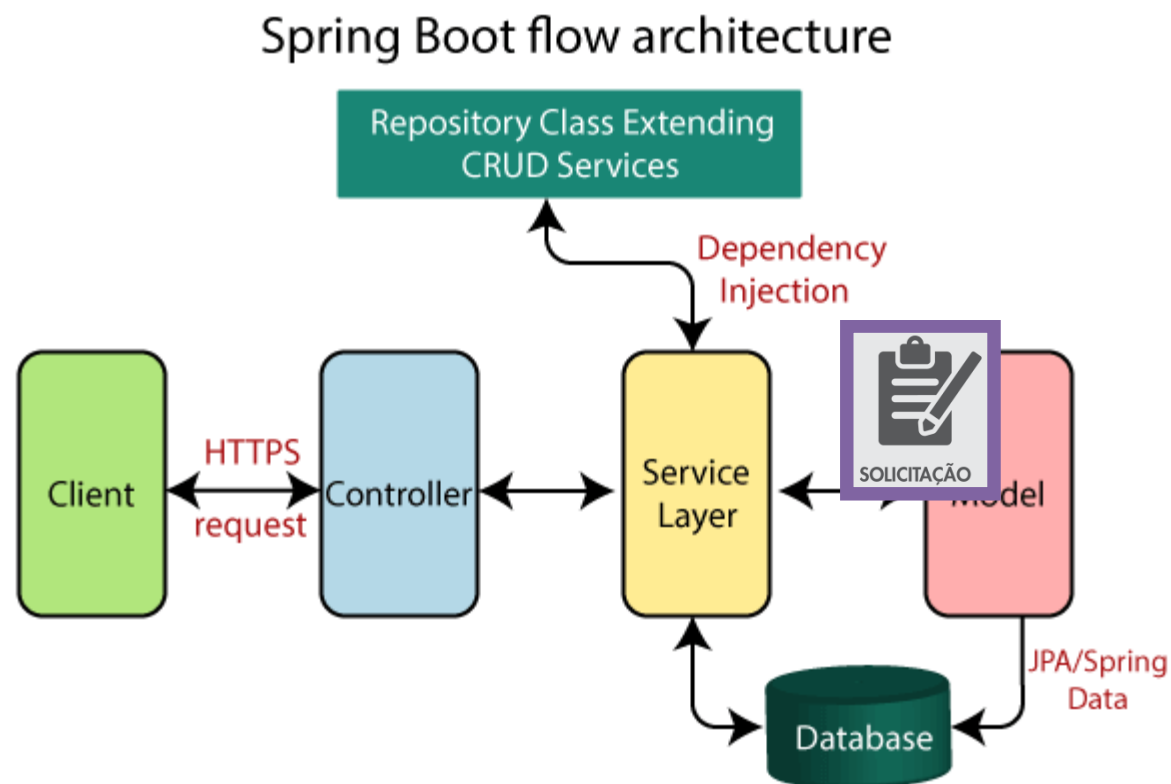
Solicitação recebida

- Verificação de Pre-requisitos



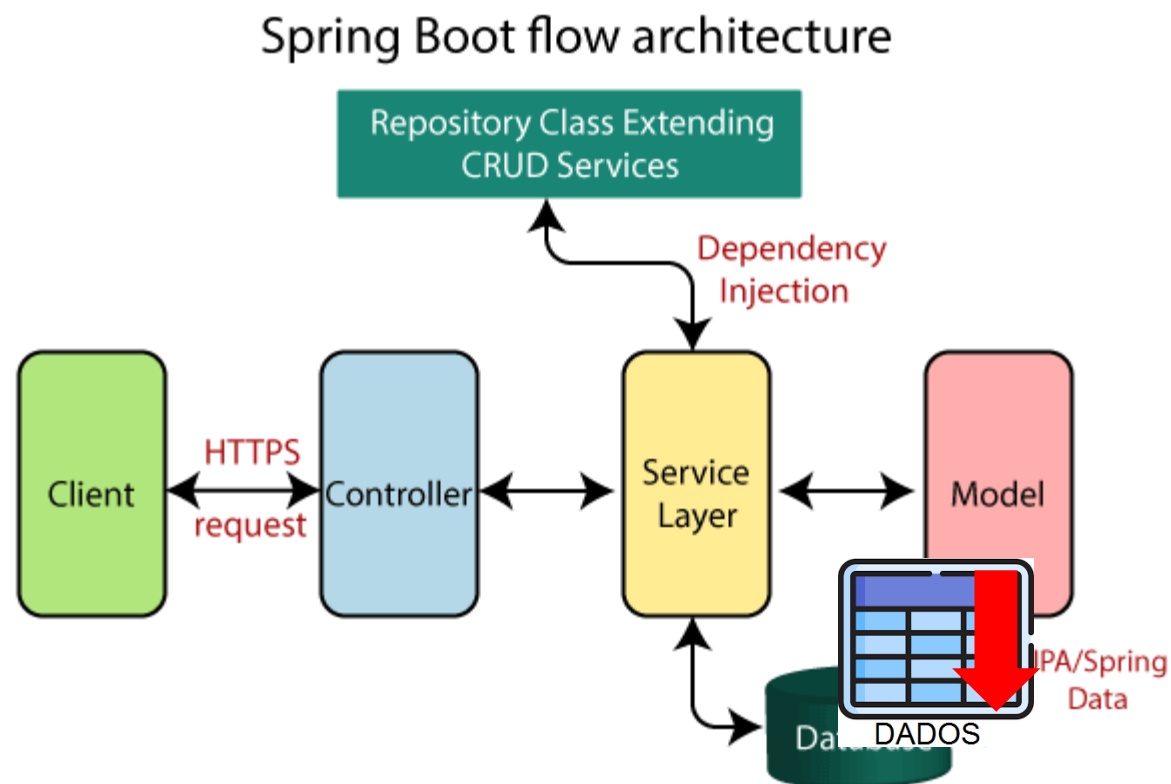
Solicitação recebida

- Verificação de Pre-requisitos



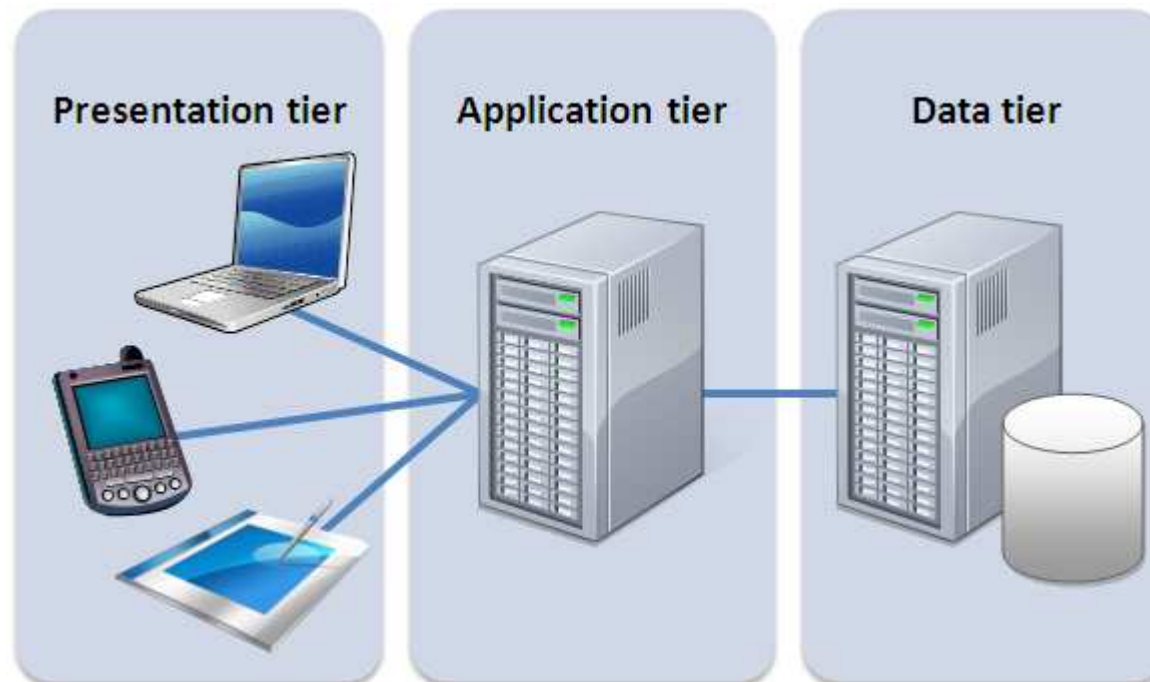
Solicitação recebida

- Verificação de Pre-requisitos



Segue-se um padrão

- Que se adere aos sistemas de três camadas



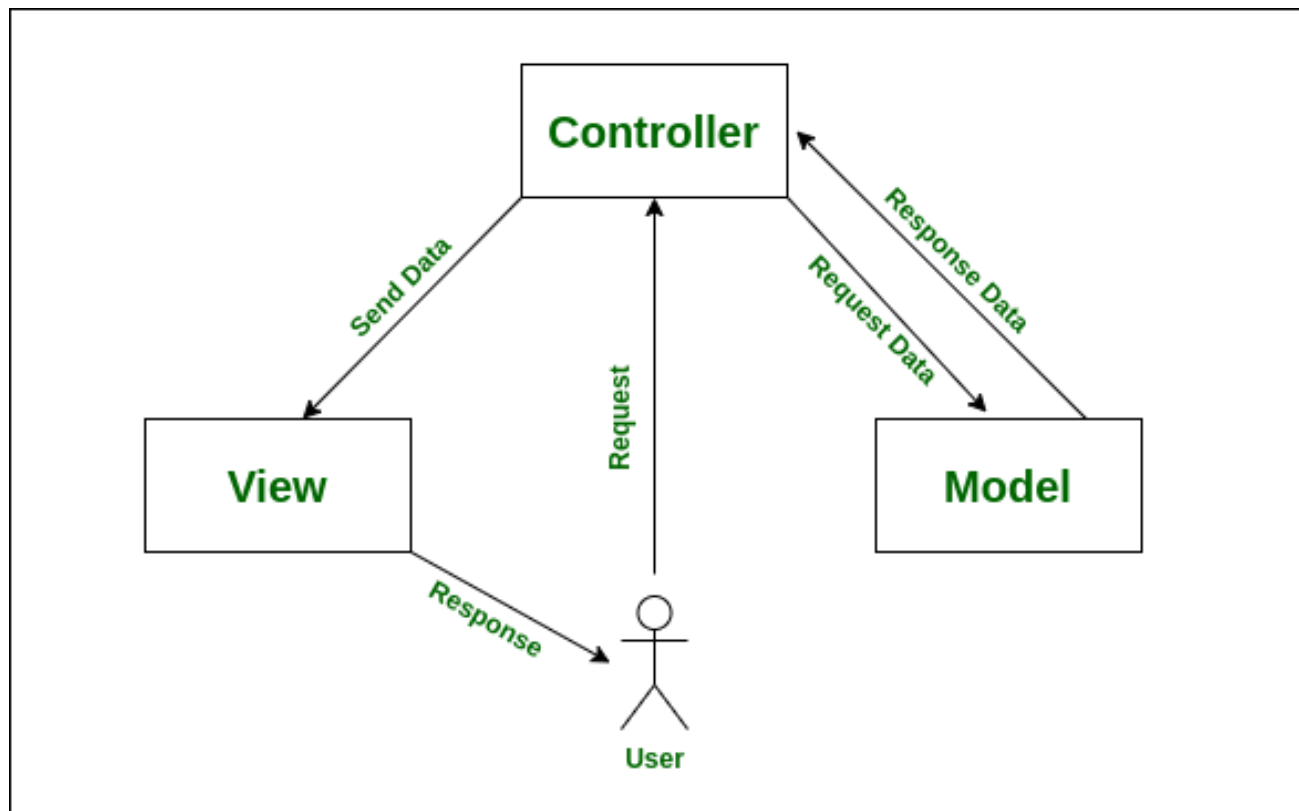
MVC

- Model View Controller
- Model (Modelo)
 - Parte do dado que está persistida ou enviada para outro ambiente/sistema
 - Persistir é enviar para outro ambiente: o BD!
- View (Visualização)
 - Devolve alguma representação para fora do sistema
 - Após processamento ou consulta
 - Normalmente para um usuário externo

MVC

- Model View Controller
- Controller
 - Responsável por fazer as coisas acontecerem.
 - Receber uma solicitação (View) e acionar as classes para realizá-la (Model)
 - Receber uma atualização ou resultado de consulta (Model) e construir os objetos utilizáveis por quem solicitou (View)
 - Manipulador de Requisições (Requests)

MVC

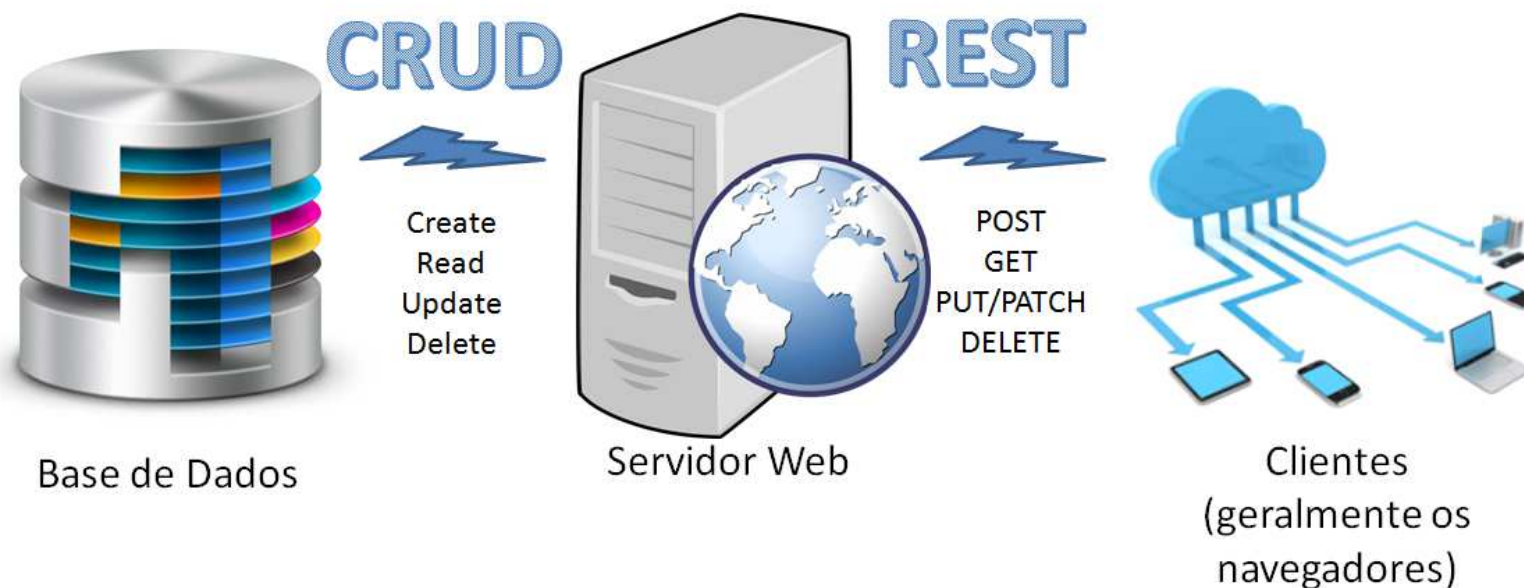


Porque MVC

- Organização de sistemas de grande porte
 - Quebrado em partes menores
 - Reutilizáveis (baixo acoplamento, alta coesão)
 - Integradas via MVC
- Distribuir
 - Componentes podem executar remotamente
 - Cada um com seu MVC
- Evolução
 - Acrescentar componentes (integrando com os demais)
 - Modificar componentes

SpringBoot é MVC

- Você não é obrigado a usar MVC
 - Mas é fortemente encorajado



SpringBoot é MVC

- Você não é obrigado a usar MVC
 - Mas é fortemente encorajado

