

Desenvolvimento para Web 3

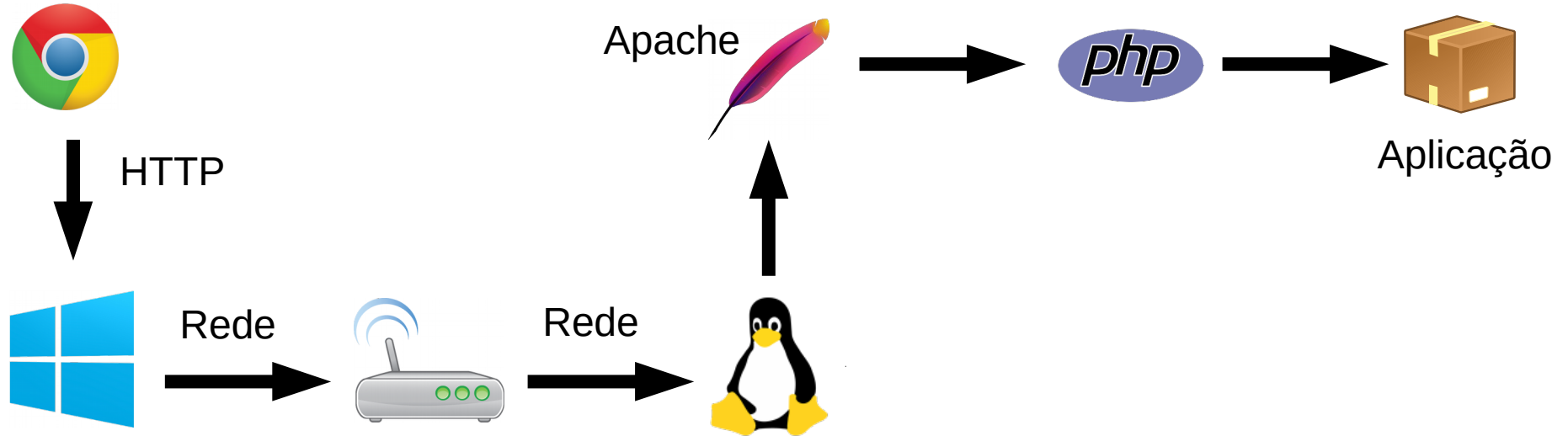
Prof. Guilherme da Costa Silva

Desenvolvimento para Web

- Vocês lembram qual é o caminho da requisição?

Desenvolvimento para Web

- Vamos rever o caminho da requisição.



Desenvolvimento para Web

- É importante saber o caminho da requisição pois podem haver bugs em qualquer local do caminho. Numa aplicação real, vocês também precisarão se preocupar com possíveis falhas de segurança nesse caminho.

Desenvolvimento para Web

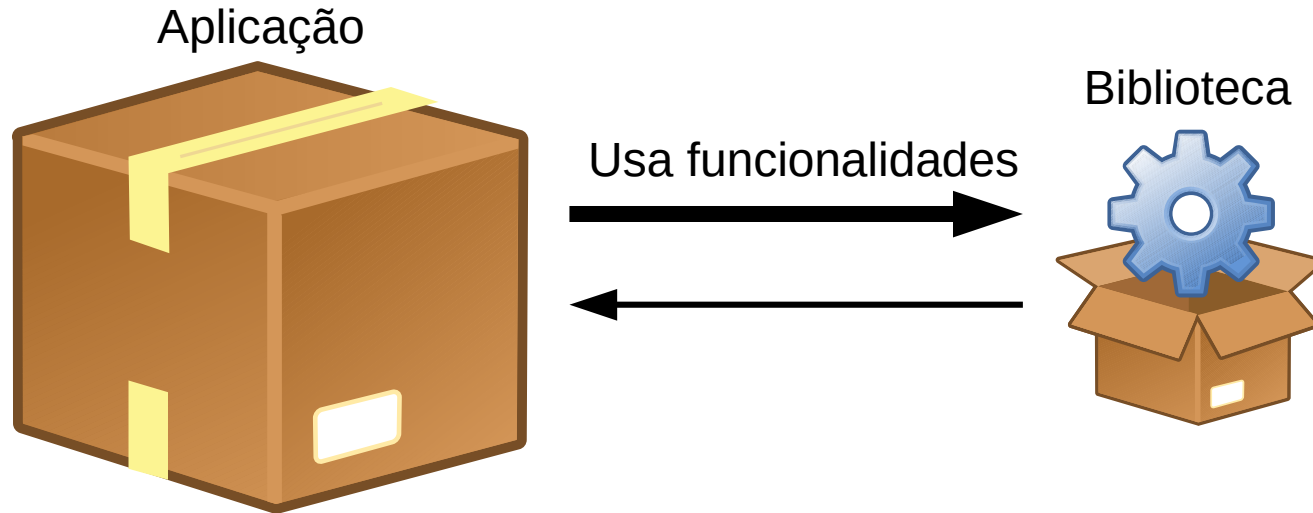
- Agora vamos estudar como criar uma aplicação mais robusta do que um simples arquivo.
- Vocês sabem o que é uma biblioteca e um *framework*?

Biblioteca

- Quando queremos reutilizar o código-fonte de alguém, usando apenas algumas funcionalidades, estamos usando uma biblioteca.
- Então biblioteca é um conjunto de funcionalidades para serem reaproveitadas.

Biblioteca

- Exemplo: biblioteca de manipulação de imagens.



Biblioteca

- É importante notar que a aplicação tem o controle do fluxo de execução do programa. A biblioteca possui apenas uma parte da execução, quando é chamada.

Framework

- Para entendermos o que é um *framework*, pense num prédio. Ele possui uma estrutura com entrada, escadas, elevadores, estacionamento, a parte elétrica e hidráulica.

Framework

- O prédio possui toda a estrutura e dentro dele estão os apartamentos. Cada apartamento pode ser personalizado, mas todos usam a estrutura do prédio.
- O ***framework*** é o **prédio** e a aplicação, o apartamento.

Framework

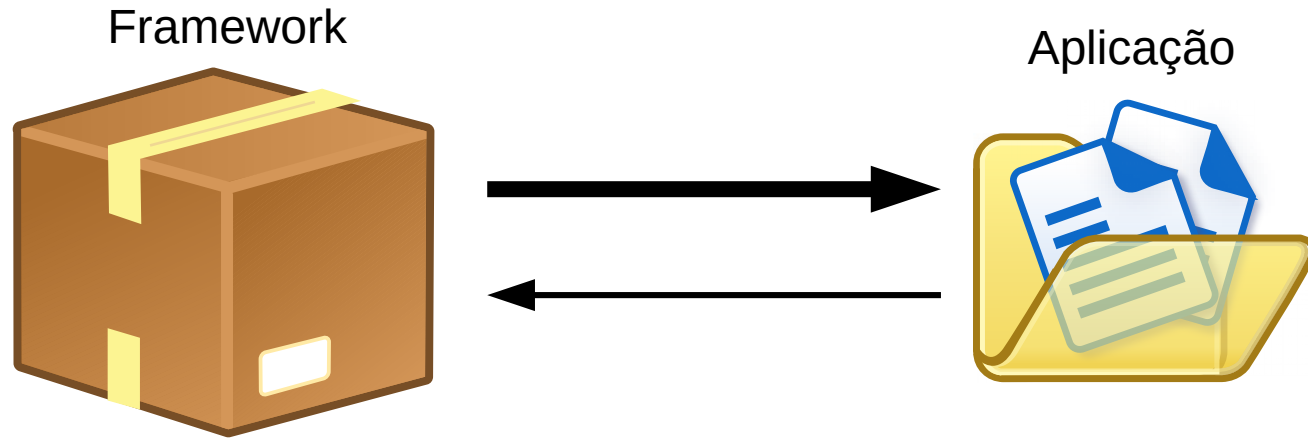
- *Framework* significa **arcabouço**. É como uma base estrutural para a aplicação. Nesse caso, o *framework* possuirá o controle do fluxo de informações. Ele é quem decide quais métodos serão chamados e qual a sua ordem.
- Sua aplicação será “apenas” uma parte da execução.

Framework

- Além disso, o *framework* também oferece muitas funcionalidades para a aplicação. Portanto ele pode ter várias bibliotecas.
- Tudo isso é importante para que o desenvolvimento seja mais produtivo, ou seja, reuse muitas funcionalidades prontas.

Framework

- Exemplo de uma aplicação web com um *framework*.



Framework

- Vantagens de um *framework*:
 - Produtividade.
 - Padronização.
 - Reuso de funcionalidades.
 - Segurança.
 - Desempenho.

Framework

- Cada linguagem de programação possui vários *frameworks* diferentes. Quanto maior o *framework*, mais funcionalidades ele terá, porém será mais engessado e demorará mais para aprendê-lo.

Framework

- Nas nossas aulas, usaremos o *framework* DW3. Ele foi feito justamente para ensinar o funcionamento de um *framework* web em PHP.
- Vários conceitos que vocês aprenderão servirá para todos os *frameworks* que vocês usarem.

DW3

- Baixe o DW3 em <https://github.com/guilhermedacsilva/web3>
- Coloque-o na pasta raiz do Apache. O projeto possui várias aplicações e o *framework* separado. É necessário o MySQL e o módulo MySQL do Apache.

DW3

- Programas necessários para executar o DW3:
 - Apache com módulos: PHP, MySQL e rewrite.
 - PHP (versão 7).
 - MySQL / MariaDB 10.

DW3

- Instalação:
 - Instale o MySQL
 - Instale o Apache2 e o módulo do PHP
 - Instale o PHP7 e o módulo para se conectar no MySQL.
 - Habilite o módulo rewrite: `sudo a2enmod rewrite`
 - Altere o `apache.conf` em `/etc/apache2` para permitir “AllowOverride All”
 - Verifique as permissões de `/var/www/html`
 - Reinicie o apache

DW3

- Configuração.

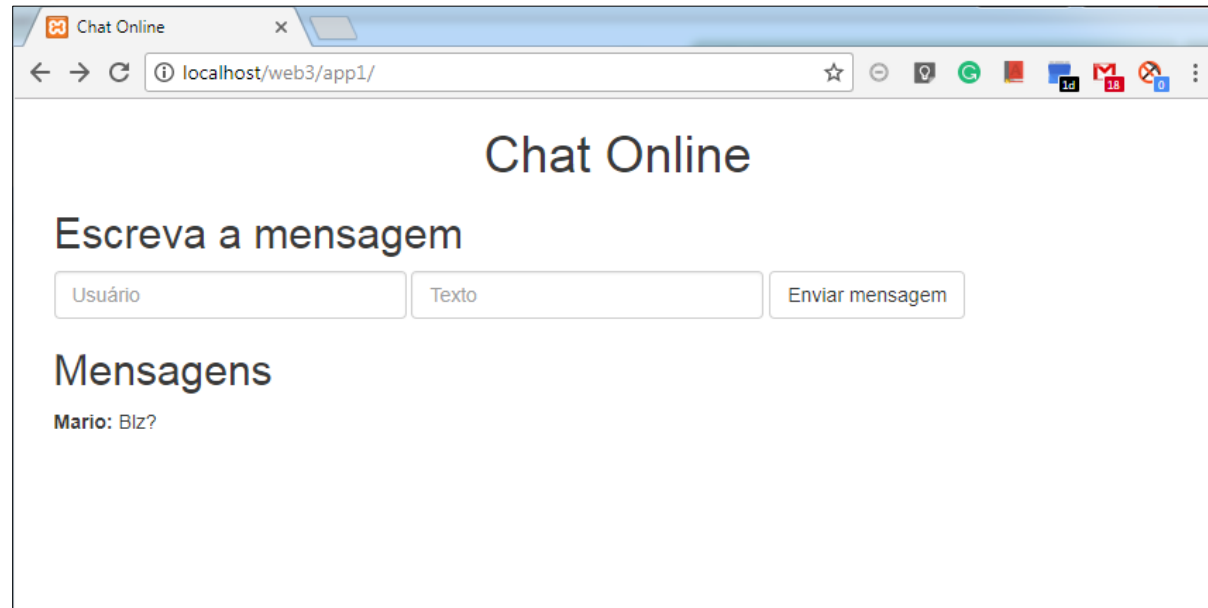
Atualize os seguintes arquivos da app1/cfg/:

- banco.php: acesso ao banco de dados;
- geral.php: verifique a URL_RAIZ.

- Crie o banco de dados usando as sqls da pasta app1/sqls.

DW3

- Acesse a aplicação pelo navegador e veja se tudo está funcionando.

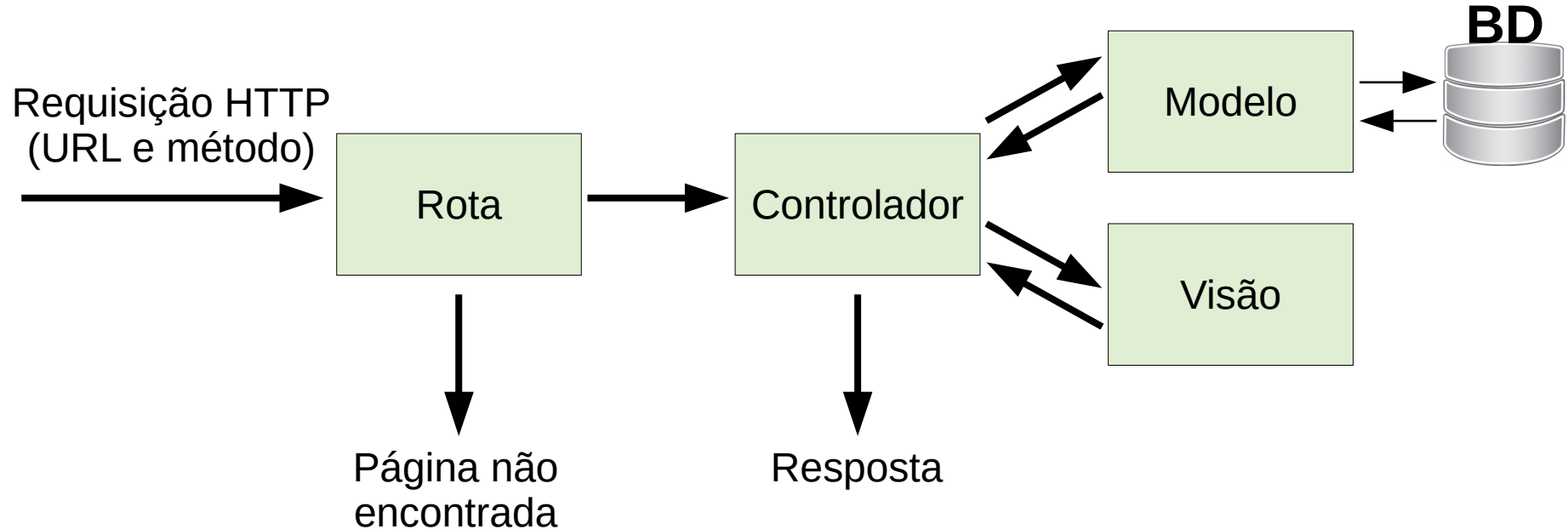


DW3

- Dê uma olhada na estrutura da diretórios da aplicação. Cada pasta possui uma finalidade.

DW3

- Ciclo de vida aplicação (MVC).



DW3

- Perceba que para eu criar uma nova página, eu vou precisar de uma rota, controlador e visão (3 arquivos). Se página acessar o banco de dados, vou precisar de um modelo também.

DW3

- Vamos ver os arquivos das aplicações.
 - Aplicação 0: só possui uma página estática.
 - Aplicação 1: possui um Chat Online com banco de dados.

DW3

- Na Aplicação 1, não foi usado o padrão de projeto Data Access Object (DAO), mas sim o **Active Record**. Nesse caso, o acesso ao banco e a lógica de negócios ficam juntos numa única classe.

DW3

- As aplicações também possuem testes automatizados. Toda aplicação deve possuir testes, para certificar de que tudo (aparentemente) funciona.