

Crud com PHP

Prof. Daniel Di Domenico

daniel.domenico@ifpr.edu.br

LAMP:



Linux



Apache



Objetivo das aulas

- Vamos desenvolver um CRUD com PHP:
 - **CRUD:** Create, Read, Update and Delete
 - Padrão MVC (Model, View, Controller)
- Ferramentas necessárias (pilha LAMP):
 - Apache com PHP instalado
 - Suporte a PDO e MySQL
 - Banco de dados MySQL
 - phpMyAdmin ou outro cliente com suporte a MySQL
 - Visual Studio Code (VSCode)
 - Linux (não é obrigatório, mas é recomendado)

MVC

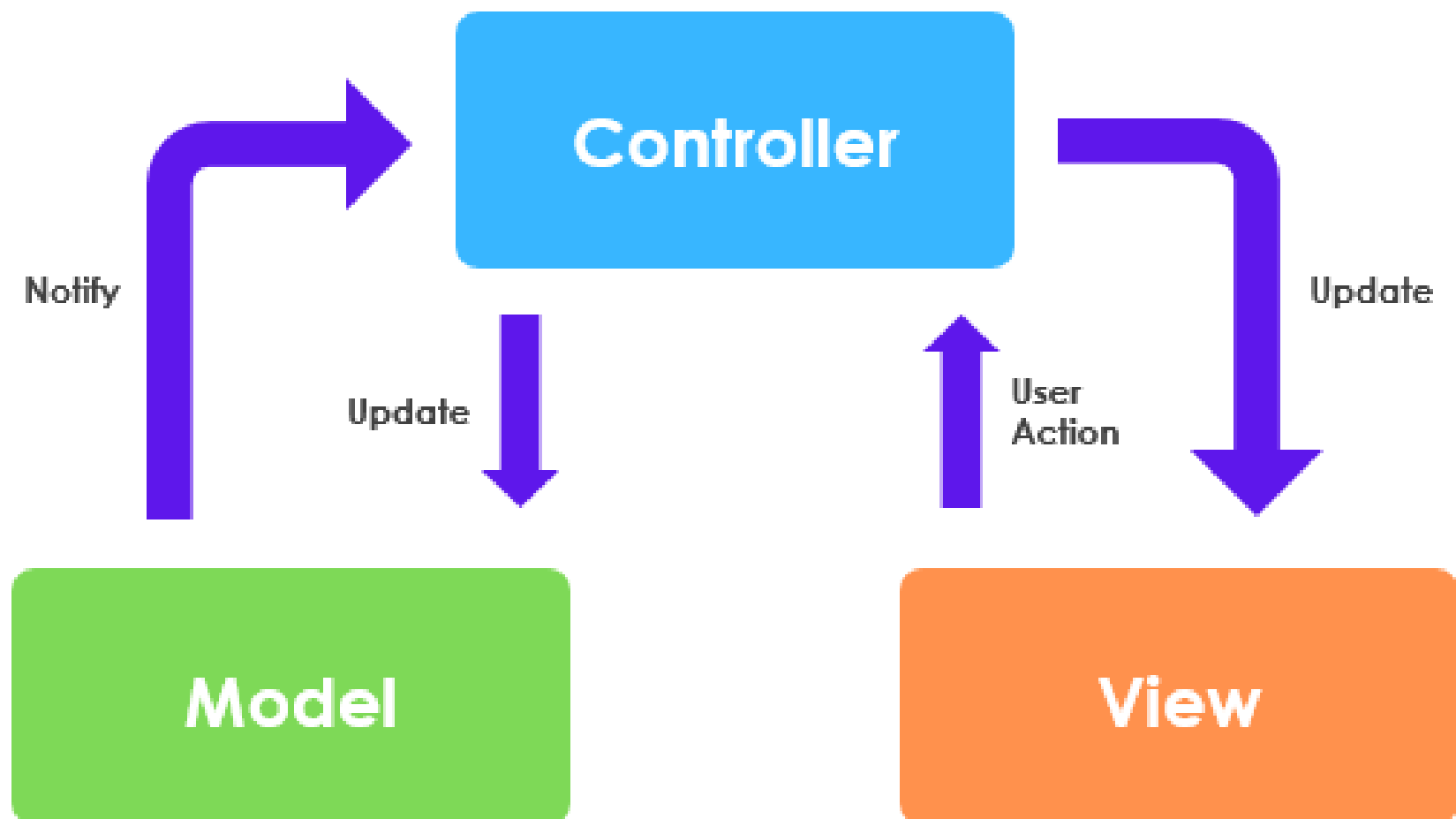
- Forma de estruturar um projeto, dividindo a implementação em (três) camadas com funções específicas
- Vantagens do MVC:
 - Facilita a manutenção e alteração do código (organização)
 - Isolamento das regras de negócio da lógica de apresentação
 - Diminui o acoplamento e aumenta a coesão das classes
 - Possibilita o reaproveitamento de classes e partes da implementação em projetos futuros
 - As camadas são independentes
 - Padrão que pode ser utilizado em diversos tipos de projetos
 - Desktop, Web e Mobile

Camadas do MVC



**INSTITUTO
FEDERAL**
Paraná

Campus
Foz do Iguaçu



Camadas do MVC

- **Model:** gerencia como os dados do sistema se comportam por meio de funções e das regras de negócio previamente definidas
- **View:** camada que apresenta as informações do sistema de forma visual ao usuário
 - Interage com o usuário por meio de botões, campos e mensagens
 - Permite entradas (questionamentos) e saídas (respostas)
- **Controller:** liga as requisições enviadas pelo View com as respostas do Model
 - Permite a comunicação entre o Model e o View
 - É a única camada que conhece quem é o responsável por executar a operação que retornará uma resposta ao usuário

Funcionalidades do projeto

- A qual camada pertencem estas funcionalidades (View ou Model):
 - Consulta ao banco de dados
 - Renderizador de elementos HTML
 - Cálculo de juros de uma aplicação financeira
 - Listagem de clientes cadastrados
 - Salvar um cliente na base de dados

PHP com banco de dados

- Utilizaremos a classe PDO para conectar ao MySQL
- Criar uma conexão:

```
$options = array(<...OPÇÕES>);  
  
$conn =  
    new PDO("mysql:host=XXXX;dbname=XXX",  
            "user", "pass", $options);
```

PHP com banco de dados

- Exemplos de opções da conexão PDO:

```
$options = array(  
    //Define o charset da conexão  
    PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8",  
    //Define o tipo do erro como exceção  
    PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,  
    //Define o tipo de retorno do array de resultados  
    //(campo => valor)  
    PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC);
```


PHP com banco de dados

- Executando consulta com PDO:

```
//Função que retorna a conexão PDO com o banco  
$conn = conectar_db();  
  
$sql = "SELECT * FROM exemplo";  
  
$stmt = $conn->prepare($sql);  
$stmt->execute();  
$result = $stmt->fetchAll();
```

PHP com banco de dados

- Retornando consulta com PDO:

```
/* foreach para percorrer os registros,  
   acessando os campos pelo nome */  
  
/* $result contém os registros retornados pela  
   consulta */  
  
foreach ($result as $reg):  
    $curso = new Curso($reg['cod_curso'],  
                       $reg['nome_curso']);  
endforeach;
```

PHP com banco de dados

- Executando INSERT/UPDATE/DELETE com PDO:

```
//Função que retorna a conexão PDO com o banco
$conn = conectar_db();

$sql = "INSERT/UPDATE/DELETE... <...PARAMS>";

$stmt = $conn->prepare($sql);
$stmt->execute([...PARAMETROS]);
```

Variáveis superglobais PHP

- Podem ser acessadas por qualquer método em qualquer arquivo .php. Exemplos:
- **\$_GET['value']** : retorna o valor de *value* recebido na página pelo método GET
 - Método GET envia as informações de um formulário HTML concatenadas na URL da página
- **\$_POST['value']** : retorna o valor de *value* recebido na página pelo método POST
 - Método POST envia as informações de um formulário HTML sem deixá-las visíveis

Comando úteis PHP

- **echo:** imprime os valores na página
- **print_r():** imprime um valor de variável na página em formato compreensível ao usuário
- **.** : concatenação de string (Ex.: *\$str."texto"*)
- **foreach...endforeach:** percorrer elementos de uma lista/array
- **include_once("pagina.php"):** inclui um arquivo PHP para ser utilizado em outro arquivo
- **header("location: pagina.php"):** retorna para a página especificada

Passos do CRUD

- **1:** Criar base de dados no MySQL
 - Script SQL disponibilizado pelo professor
- **2:** Criar estrutura do projeto em PHP no padrão MVC
 - Estrutura inicial disponibilizada pelo professor
- **3:** Criar conexão com a base de dados utilizando PDO
- **4:** Implementar o CRUD de alunos

Modelo de classes proposto

Visual Paradigm Online Free Edition

CRUD



IFPR Foz do Iguaçu

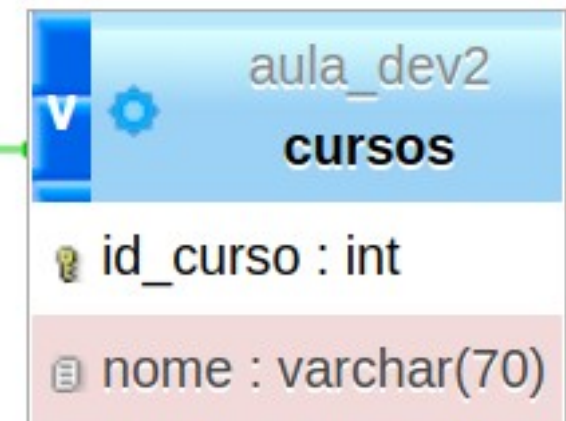
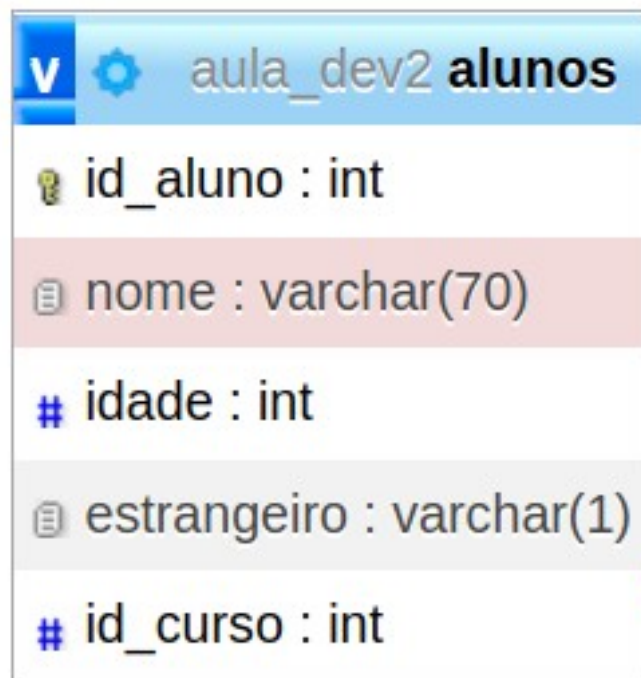
Visual Paradigm Online Free Edition

Modelo relacional da base de dados proposto



**INSTITUTO
FEDERAL**
Paraná

Campus
Foz do Iguaçu



Implementação

- Implementação do CRUD será baseado nos modelos propostos:
 - O mesmo poderá ser incrementado futuramente
 - Ex.: novas classes, novas camadas, validações, login....
- Veremos conceitos de orientação a objetos e persistência desses objetos
- Em cada aula:
 - Novas implementações serão realizadas para completarmos o CRUD com todas as operações