

Manual de Laboratório para Desenvolvimento de Sistema Web

Disciplina: Seminário Interdisciplinar: Implementação de uma Aplicação Web (ADS102)

Proposta: Desenvolver um Pequeno Sistema Web para Funcionário

Introdução

Existem várias linguagens de programação que permitem o desenvolvimento de sistemas web, porém, dependendo do contexto, uma é mais vantajosa que a outra. Além disso, existem vários outros recursos como o JavaScript, HTML (Linguagem de Marcação) e estilos (CSS), os quais podem ou devem ser utilizados no desenvolvimento de uma aplicação web, dependendo do contexto. Neste sentido, além de programar as regras de negócios e funcionalidades na linguagem de programação, normalmente é necessário programar funções JavaScript para controles das páginas, como validações de formulários por exemplo, assim como códigos de estilos (CSS) para as páginas. Isso acontece tanto em ASP, PHP, Java, dentre outras linguagens de programação web.

Dentro deste contexto, esta atividade de prática possui o objetivo de complementar o aprendizado, aplicando as principais tecnologias estudadas, para o desenvolvimento de um pequeno sistema web. Através deste sistema web desenvolvido, você poderá desenvolver o paper da disciplina, no qual devem ser realizadas pesquisas bibliográficas para descrever as tecnologias na implementação do sistema. Alguns pequenos trechos de códigos fonte principais, devem ser apresentados no paper, os quais devem ser explicados com ao menos um parágrafo.

Objetivos

- Aplicar ou aprimorar os conhecimentos obtidos para configuração do ambiente para desenvolvimento de um sistema web;
- Criar um sistema web para gravação de funcionários;
- Através do desenvolvimento do sistema web, exercitar e aprofundar os conhecimentos obtidos na linguagem de programação PHP.

Atividade e Procedimentos

Desenvolver um sistema web para cadastro de funcionários. Neste sistema, deve ter uma página principal, com um link para a página onde existe o formulário para cadastrar funcionários.

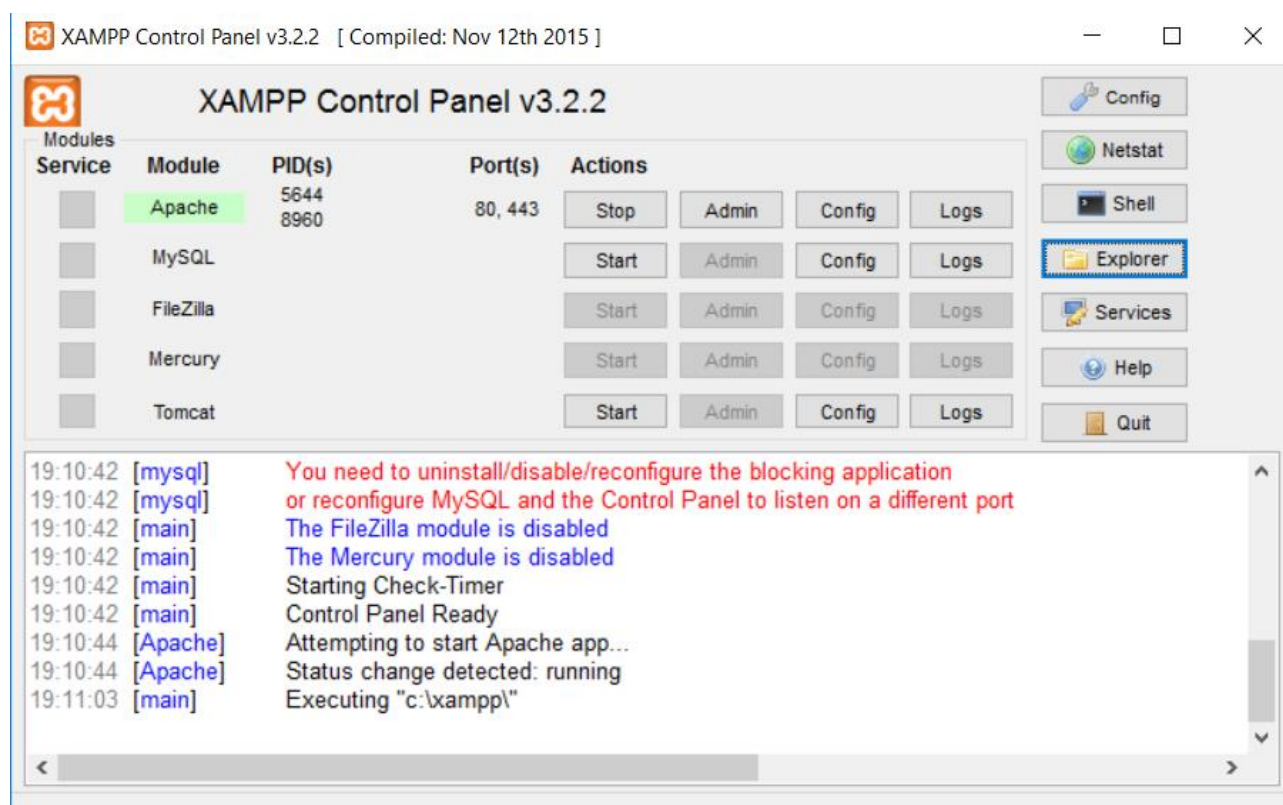
Seguem passos para o desenvolvimento:

- Criar um banco de dados chamado “Uniasselvi” no MySQL;
- Criar uma tabela chamada “Funcionario” com os campos “codigo”, “nome”, “cargo”, “descricao cargo”, “setor” e “salario (o comando SQL está descrito abaixo);
- Criar uma página inicial HTML com um link para a segunda página que terá o formulário para cadastro;
- Criar uma segunda página HTML, contendo o formulário de cadastro, onde no botão gravar será feita a requisição para cadastro do funcionário.

Desenvolvimento da Aplicação em PHP

Criar e configurar o projeto da aplicação web com XAMPP

1º Um dos ambientes mais fáceis para se desenvolver com PHP é o XAMPP, o qual já possui inúmeras ferramentas disponíveis, como servidor de aplicação, banco de dados MySQL (caso você já não o tiver instalado), entre outras. Conforme apresentada a tela do XAMPP na figura abaixo, você pode notar que o servidor Apache está inicializado (rodando), apenas ele, pois em meu computador, já existe o MySQL instalado.



2º Ainda falando do XAMPP, o botão “Explorer”, o qual está na figura anterior, abre o diretório raiz da ferramenta, conforme ilustrado na figura a seguir. Dentre suas pastas, uma das mais importantes para você desenvolver é a pasta “htdocs”, na qual é onde podemos criar nossas aplicações, separando cada uma por pasta.

Computador > Disco Local (C:) > xampp




Nome	Data de modificaç...	Tipo	Tamanho
apache	10/04/2020 12:39	Pasta de arquivos	
cgi-bin	10/04/2020 12:44	Pasta de arquivos	
contrib	10/04/2020 12:39	Pasta de arquivos	
htdocs	26/05/2020 19:12	Pasta de arquivos	
install	10/04/2020 12:44	Pasta de arquivos	
licenses	10/04/2020 12:39	Pasta de arquivos	
locale	10/04/2020 12:39	Pasta de arquivos	
mysql	10/04/2020 12:40	Pasta de arquivos	
perl	10/04/2020 12:41	Pasta de arquivos	
php	10/04/2020 12:44	Pasta de arquivos	
phpMyAdmin	10/04/2020 12:44	Pasta de arquivos	
sendmail	10/04/2020 12:44	Pasta de arquivos	
tmp	08/05/2020 14:18	Pasta de arquivos	
tomcat	10/04/2020 12:40	Pasta de arquivos	
webdav	10/04/2020 12:39	Pasta de arquivos	
apache_start	07/06/2013 08:15	Arquivo em Lotes ...	1 KB
apache_stop	07/06/2013 08:15	Arquivo em Lotes ...	1 KB
catalina_service	30/03/2013 09:29	Arquivo em Lotes ...	10 KB
catalina_start	07/06/2013 08:15	Arquivo em Lotes ...	2 KB

3º A imagem a seguir apresenta o diretório “htdocs”, sendo uma pasta normal do sistema. Mas o que for colado dentro dela, será disponibilizado pelo servidor de aplicação, localmente. É o que você precisa para testar a aplicação PHP desenvolvida.

Computador > Disco Local (C:) > xampp > htdocs

Nome	Data de modificaç...	Tipo	Tamanho
dashboard	10/04/2020 12:39	Pasta de arquivos	
uniasselvi	26/05/2020 19:17	Pasta de arquivos	
vendas	20/04/2020 20:03	Pasta de arquivos	

4º Para iniciar o projeto PHP, para este exemplo, deve ser criada uma pasta chamada “uniasselvi”, dentro de “htdocs”. Dentro desta pasta, devem ser criados três arquivos, “index.php”, “cadastro.php” e “insert.php”, e para isso, você pode utilizar qualquer editor de texto. A figura a seguir apresenta como deve ficar a estrutura do diretório criado.

Computador > Disco Local (C:) > xampp > htdocs > uniassearvi				▼	🔍
Nome	^	Data de modificaç...	Tipo	Tamanho	
 cadastro.php		26/05/2020 19:18	Arquivo PHP	1 KB	
 index.php		26/05/2020 19:16	Arquivo PHP	1 KB	
 insert.php		26/05/2020 20:45	Arquivo PHP	2 KB	

5º Agora, basta implementar os códigos fonte em cada arquivo. Começamos pelo “index.php”, o mais simples de todos, o qual possui apenas um link para a página “cadastro.php”, conforme solicitado no enunciado, início deste PDF.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="ISO-8859-1">
  <title>Página Inicial</title>
</head>
<body>

  <br><br>
  <a href="cadastro.php">Cadastro Funcionário</a>
  <br><br>

</body>
</html>
```

6º O segundo código implementado, é o formulário de cadastro. Para isso, abra o arquivo “cadastro.php”, criado anteriormente, e implemente o código a seguir. Perceba que o valor para o atributo “action” do form é a página “insert.php”, justamente onde será realizada a inserção do registro na tabela.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>

  <form action="insert.php" method="post">
    <table>
      <tr>
```

```

        <td colspan="2">Cadastro de Funcionário</td>
      </tr>
      <tr>
        <td>Nome:</td>
        <td><input type="text" name="nome"></td>
      </tr>
      <tr>
        <td>Cargo:</td>
        <td><input type="text" name="cargo"></td>
      </tr>
      <tr>
        <td>Descrição cargo:</td>
        <td><textarea name="descCargo" cols="15"
rows="15"></textarea></td>
      </tr>
      <tr>
        <td>Setor:</td>
        <td><input type="text" name="setor"></td>
      </tr>
      <tr>
        <td>Salário:</td>
        <td><input type="text" name="salario"></td>
      </tr>
    </table>
    <br><br>
    <input type="submit" name="Gravar">
  </form>

</body>
</html>

```

7º O último código fonte a ser implementado é o da página “insert.php”, o qual possui justamente a implementação da função de gravação do novo funcionário na tabela de nosso banco de dados. Então abra o arquivo “insert.php” e implemente o código fonte a seguir nele. Preste atenção na terceira linha do código abaixo, onde é configurada a conexão com o banco de dados. No meu caso, configurei o MySQL com usuário e senha com valor “root”, e por isso, foram passadas duas vezes este valor para a conexão. Caso você tenha configurado o MySQL sem senha, deve deixar em branco o parâmetro senha, ou caso tenha informado outra senha, deve informá-la.

```

<?php
/* Conexão com BD MySQL (usuário 'root', senha 'root' e banco 'uniassearvi') */
$link = mysqli_connect("localhost", "root", "root", "uniassearvi");

// Valida se conexão com banco de dados

```

```
if($link === false){
    die("ERRO: Não foi possível conectar ao BD. " . mysqli_connect_error());
}

// variáveis criadas para obter valores dos parâmetros do formulário
$nome = mysqli_real_escape_string($link, $_REQUEST['nome']);
$cargo = mysqli_real_escape_string($link, $_REQUEST['cargo']);
$descCargo = mysqli_real_escape_string($link, $_REQUEST['descCargo']);
$setor = mysqli_real_escape_string($link, $_REQUEST['setor']);
$salario = mysqli_real_escape_string($link, $_REQUEST['salario']);

$codigo = 1;
// pegando o próximo código (sem utilização de sequence do banco)
$sql = "SELECT MAX(CODIGO) AS CODIGO FROM FUNCIONARIO";
if($result = mysqli_query($link, $sql)){
    if(mysqli_num_rows($result) > 0){
        if($row = mysqli_fetch_array($result)){
            if (intval($row['CODIGO']) > 0){
                $codigo = intval($row['CODIGO']) + 1;
            }
        }
    }
}

// Realiza inserção do novo registro na tabela do banco de dados
$sql = "INSERT INTO FUNCIONARIO (CODIGO, NOME, CARGO, DESCRICAOCARGO, SETOR, SALARIO) VALUES ('$codigo', '$nome', '$cargo', '$descCargo', '$setor', '$salario')";
if(mysqli_query($link, $sql)){
    echo "Gravação efetuada com sucesso!"; // aqui poderia ser incluído um código para redirect
} else{
    echo "Erro (Não foi possível inserir o registro na tabela) $sql. " . mysqli_error($link);
}

// Close connection
mysqli_close($link);
?>
```

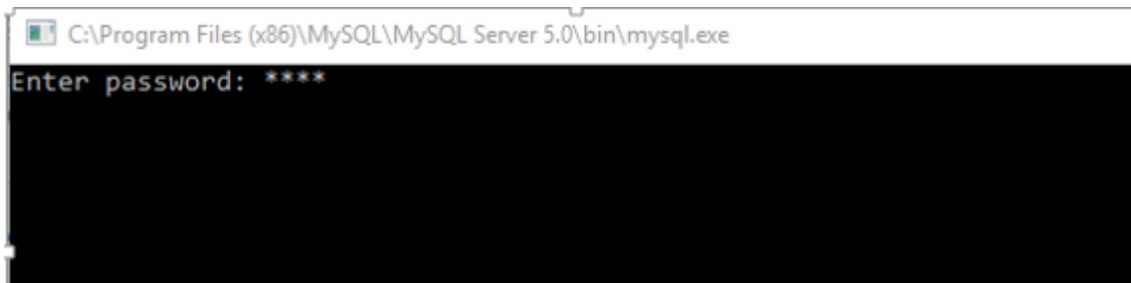
Preparar o banco de dados no SGBD MySQL

Vamos criar agora nosso banco de dados e suas tabelas, necessárias para nossa aplicação web. Acesse a linha de comando do MySQL, atalho sendo apresentado na figura abaixo.

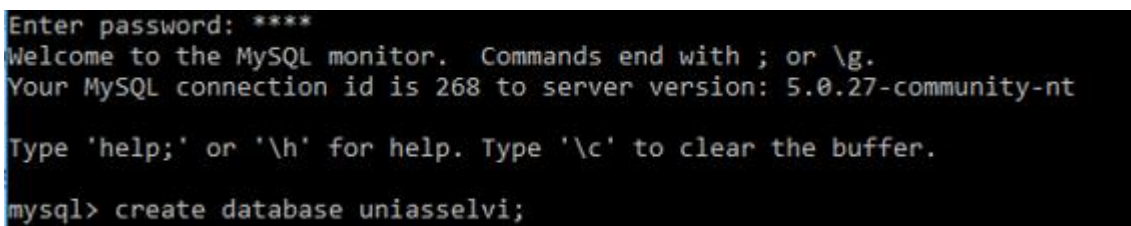


MySQL Command Line Client

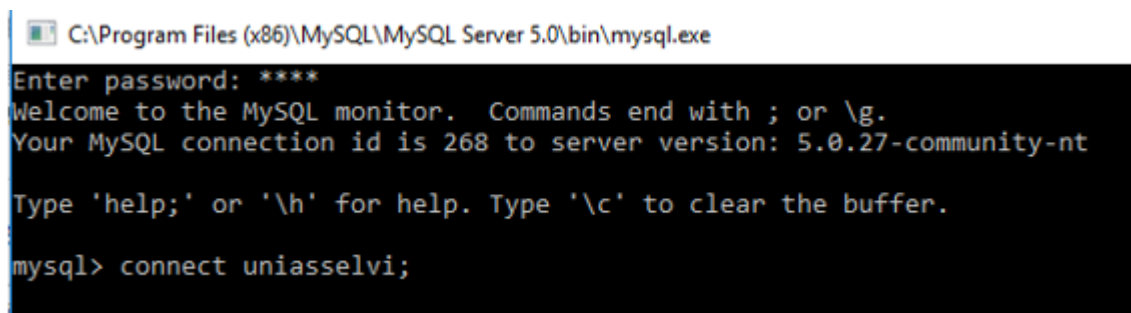
Será aberta a tela, conforme ilustrada na figura a seguir. Informe a senha “root”, caso você manteve esta senha durante a instalação do MySQL, e então pressione ENTER.



Crie o banco de dados, através do comando “create database uniassselvi;”, conforme apresentado abaixo. Neste caso, o nome de nosso banco de dados será “Uniassselvi”.



Conecte no banco de dados “Uniassselvi”, através do comando “connect uniassselvi;”, para então serem criadas, neste, as tabelas necessárias para nossa aplicação web. Tente utilizar a mesma nomenclatura descritas neste manual para o banco de dados e tabelas.



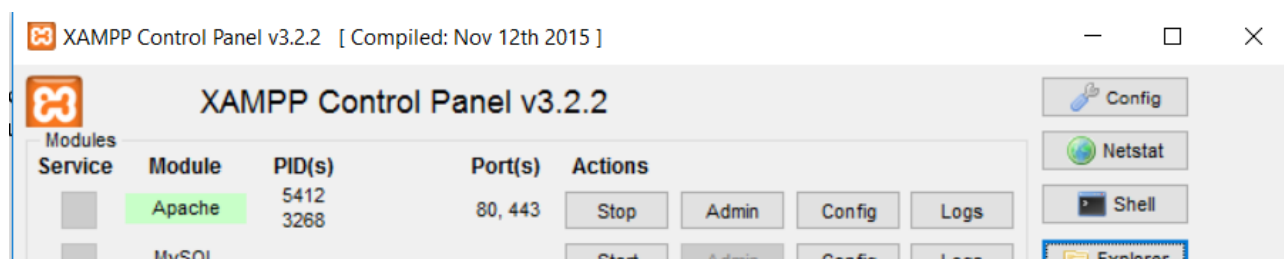
Agora, crie a tabela “Funcionario” no banco de dados “Uniassselvi”, criado acima. Segue comando SQL.

```
CREATE TABLE FUNCIONARIO (  
  CODIGO INT(9) NOT NULL,  
  NOME VARCHAR(30) NOT NULL,  
  CARGO INT(9) NOT NULL,  
  DESCRICAOCARGO VARCHAR(30) NOT NULL,  
  SETOR INT(9) NOT NULL,
```

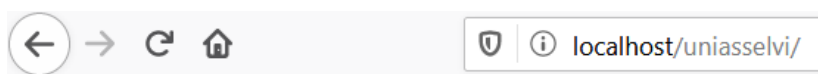

SALARIO DOUBLE(6, 2) NOT NULL

Execução da Aplicação PHP

Na tela do XAMPP, pressiona o botão “Start” para o módulo Apache, conforme figura a seguir, para que o servidor de aplicação seja iniciado.



No navegador, acesse a aplicação pela URL <http://localhost/uniassearvi/>, conforme ilustrado na figura abaixo.



[Cadastro Funcionário](#)

Após clicar no link “Cadastro Funcionário” da página inicial da aplicação, é apresentado o formulário desenvolvido. Basta informar os valores nos campos e pressionar o botão para gravar um novo funcionário no banco de dados.

← → ↻ 🏠

🔒 ⓘ localhost/uniassearvi/cadastro.php

Cadastro de Funcionário

Nome:

Cargo:

Descrição cargo:

Setor:

Salario:

Submit Query

Observação: A aplicação está funcionando, inclusive a gravação no banco de dados, porém, não foram adicionados controles JavaScript para evitar que sejam informados caracteres alfanuméricos em campos numéricos do formulário, o que gerará erro na gravação, caso informado. Se os mesmos tipos de dados forem informados nos respectivos campos, a aplicação funcionará. Estes controles, pode ser implementados por você, acadêmico, para que pratique o JavaScript.