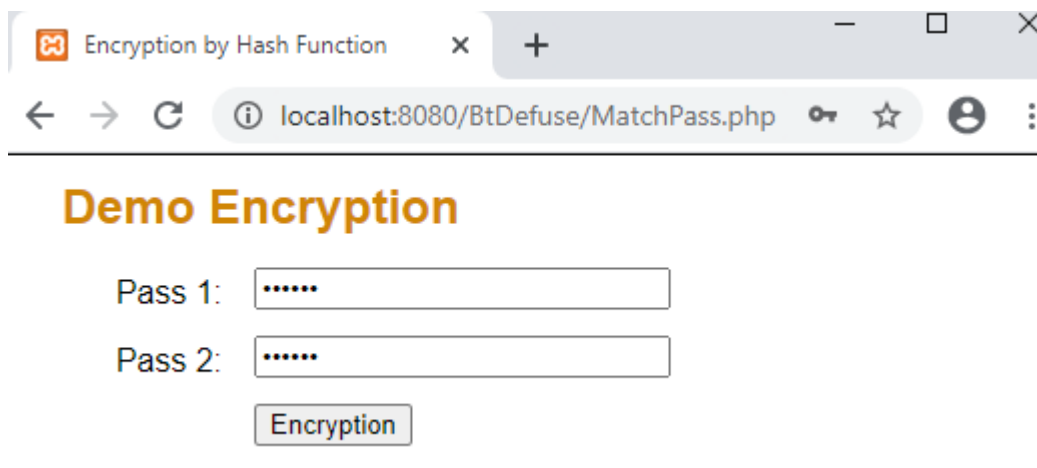


**Lab 12.1. Sử dụng các hàm và thư viện defuse() để mã hóa dữ liệu**

**Bài 01.** Xây dựng ứng dụng web, demo sử dụng hàm password\_hash() trả về mật khẩu với giá trị băm hoặc false nếu không thành công hoặc null nếu thuật toán không hợp lệ và hàm password\_verify() trả về true nếu mật khẩu và giá trị băm khớp nhau hoặc false nếu không khớp, với giao diện ứng dụng như sau:



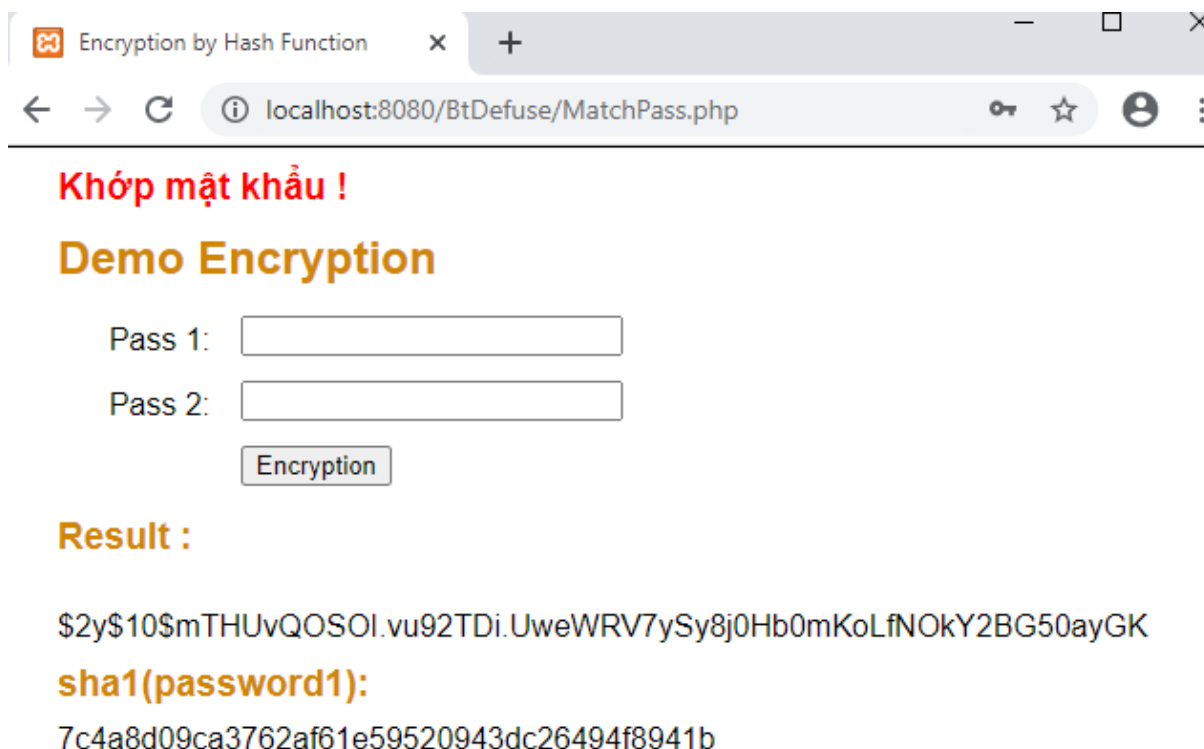
Encryption by Hash Function

Pass 1: .....

Pass 2: .....

Encryption

+ Giao diện kết quả khi người dùng, kích vào nút **Encryption** để kiểm tra



Encryption by Hash Function

Khớp mật khẩu !

Demo Encryption

Pass 1:

Pass 2:

Encryption

Result :

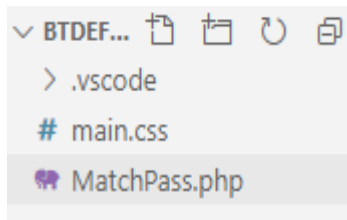
\$2y\$10\$mTHUvQOSOI.vu92TDi.UweWRV7ySy8j0Hb0mKoLfNOKY2BG50ayGK

sha1(password1):

7c4a8d09ca3762af61e59520943dc26494f8941b

**Gợi ý:**

+ **B1.** Tạo cấu trúc ứng dụng



+ **B2.** Tạo file **MatchPass.php** sử dụng hàm **password\_hash()** để thực hiện chức năng mã hóa mật khẩu sang giá trị băm và hàm **password\_verify()** để so khớp mật khẩu với giá trị băm.

```
<?php
```

```
//Chú ý:
```

```
//Đặt tùy chọn PASSWORD_DEFAULT và các phiên bản tương lai của PHP sẽ tự động
```

```
//sử dụng các thuật toán mạnh hơn để tạo băm mật khẩu mà không cần bạn phải cập nhật mã.
```

```
$action = filter_input(INPUT_POST,'action');
```

```
$result = "";
```

```
if(!empty($action) && $action == 'encryption')
```

```
{
```

```
    $password1 = filter_input(INPUT_POST,'password1');
```

```
    $hash = password_hash($password1,PASSWORD_DEFAULT);
```

```
    $result .= $hash;
```

```
    $password2 = filter_input(INPUT_POST,'password2');
```

```
    if(password_verify($password2,$hash))
```

```
    {
```

```
        echo "<h2 style='color:red'>Khớp mật khẩu !</h2>";
```

```
    }
```

```
else
```

```
{
```

```
    echo "<h2 style='color:red'>Không khớp mật khẩu !</h2>";
```

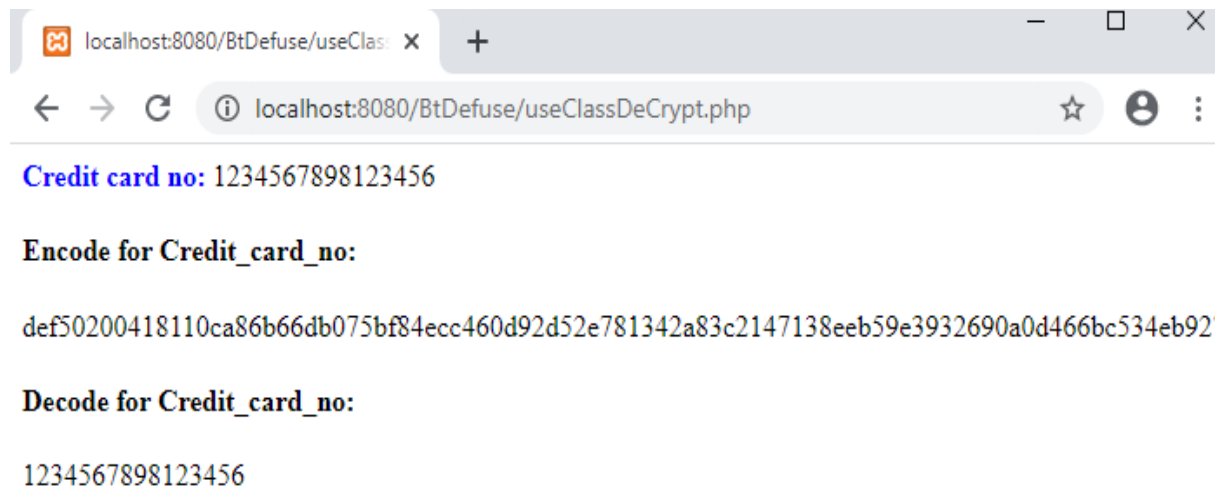
```
}
```

```
}
```

```
?>
```

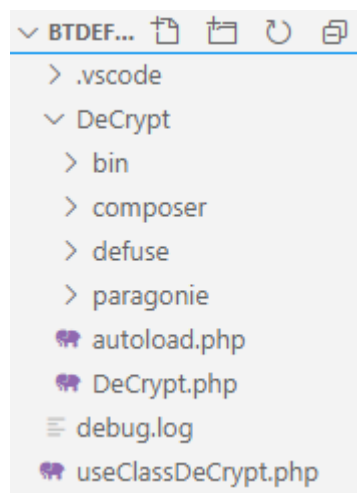
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Encryption by Hash Function</title>
  <link rel="stylesheet" href="main.css">
</head>
<body>
  <h1>Demo Encryption</h1>
  <form action="MatchPass.php" method="post" id="encryption" class="aligned">
    <input type="hidden" name="action" value="encryption">
    <label>Pass 1:</label>
    <input type="password" class="text" name="password1">
    <br>
    <label>Pass 2:</label>
    <input type="password" class="text" name="password2">
    <br>
    <label>&nbsp;</label>
    <input type="submit" value="Encryption">
  </form>
  <p>
    <?php
      if(!empty($result))
      {
        echo "<h2>Result :</h2> <br> $result ";
        echo "<br><h2>sha1(password1):</h2>";
        echo !empty($password1)?sha1($password1):"";
      }
    ?>
  </p>
</body>
</html>
```

**Bài 02.** Xây dựng ứng dụng web, sử dụng thư viện mã hóa defuse để mã hóa chuỗi Credit card no:1234567898123456 thành giá trị băm có giá trị khoảng 196 ký tự. Sau đó, giải mã giá trị băm thành chuỗi Credit card no có giá trị ban đầu. Với giao diện ứng dụng như sau:



**Gợi ý:**

- B1. Tải thư viện defuse: <https://github.com/defuse/php-encryption>
- B2. Tạo cấu trúc ứng dụng web như sau:



- B3. Xây dựng lớp **DeCrypt.php** sử dụng thư viện **Defuse** với hai phương thức **encrypt()** và **decrypt()** để mã hóa và giải mã dữ liệu.

<?php

```
require_once("autoload.php");
use Defuse\Crypto\Crypto;
use Defuse\Crypto\Key;
use Defuse\Crypto\Exception\WrongKeyOrModifiedCiphertextException;
```

```

class DeCrypt
{
    private $key;

    function __construct() {
        $this->key = Key::createNewRandomKey();
    }

    function encrypt($data) {
        $encrypted_data = Crypto::encrypt($data, $this->key);
        return $encrypted_data;
    }

    function decrypt($encrypted_data) {
        try {
            $data = Crypto::decrypt($encrypted_data, $this->key);
            return $data;
        } catch (WrongKeyOrModifiedCiphertextException $ex) {
            throw new Exception($ex->getMessage());
        }
    }
}
?>

```

- **B4.** Xây dựng giao diện file **useClassDeCrypt.php** để mã hóa và giải mã chuỗi **credit card no.**

```

<?php
require_once('DeCrypt/DeCrypt.php');

$credit_card_no = "1234567898123456";

echo "<span style='color:blue;font-weight:bold'>Credit card no: </span>" . $credit_card_no . "<br>";

$defuse = new DeCrypt();

$encrypt = $defuse->encrypt($credit_card_no);

$decrypt = $defuse->decrypt($encrypt);

echo "<h4>Encode for Credit_card_no:</h4>" . $encrypt . "<br>";

echo "<h4>Decode for Credit_card_no:</h4>" . $decrypt . "<br>";

?>

```