Assignment No.13

Problem Statement

Implement a program to generate and verify CAPTCHA image.

Theory

CAPTCHA or Captcha which stands for **"Completely Automated Public Turing test to tell Computers and Humans Apart"** is a type of ***challenge-response*** test to ensure that the response is only generated by humans and not by a computer. In simple words, CAPTCHA is the word verification test that you will come across the end of a sign-up form while signing up for Gmail or Yahoo account. The following image shows the typical samples of CAPTCHA.

Almost every Internet user will have an experience of CAPTCHA in their daily Internet usage, but only a few are aware of what it is and why they are used.

What Purpose does CAPTCHA Exactly Serve?

CAPTCHA is mainly used to prevent automated software (bots) from performing actions on behalf of actual humans.

For example, while signing up for a new email account, you will come across a CAPTCHA at the end of the sign-up form so as to ensure that the form is filled out only by a legitimate human and not by any of the automated software or a computer bot. The main goal of CAPTCHA is to put forth a test which is simple and straight forward for any human to answer but for a computer, it is almost impossible to solve.

What is the Need to Create a Test that Can Tell Computers and Humans Apart?

For many, the CAPTCHA may seem to be silly and annoying! In fact, CAPTCHA has the ability to protect systems from malicious attacks where people try to **game** the

system. Attackers can make use of the automated software to generate a huge quantity of requests thereby causing a high load on the target server. This could lead to a degrade the quality of service of a given system, either due to abuse or resource expenditure. This can affect millions of legitimate users and their requests. CAPTCHAs can be deployed to protect systems that are vulnerable to email spam, such as the services from Gmail, Yahoo and Hotmail.

Who Uses CAPTCHA?

CAPTCHAs are mainly used by websites that offer services like online polls and registration forms.

For example, Web-based email services like Gmail, Yahoo and Hotmail offer free email accounts for their users. However, upon each sign-up process, CAPTCHAs are used to prevent spammers from using a bot to generate hundreds of spam mail accounts.

Designing a CAPTCHA System:

CAPTCHAs are designed on the fact that, the computers lack the ability that human beings have when it comes to processing visual data. It is more easily possible for humans to look at an image and pick out the patterns than a computer. This is because; computers lack the real intelligence that humans have by default.

CAPTCHAs are implemented by presenting users with an image which contains distorted or randomly stretched characters which only humans should be able to identify. Sometimes, characters are stroked out or presented with a noisy background to make it even harder for computers to figure out the patterns.

Most, but not all, CAPTCHAs rely on a visual test. Some Websites implement a totally different CAPTCHA system to tell humans and computers apart. For example, a user is presented with 4 images in which 3 contains picture of animals and one contain a flower. The user is asked to select only those images which contain animals in them. This Turing test can easily be solved by any human, but almost impossible for a computer.

Design Analysis / Implementation Logic:

Algorithm:

1. Display the distorted code and textbox to accept that code on webpage.
2. To execute these program use any web server (Ex.XAMPP).
3. Once entered code and distorted code matched, provide message **correct captcha**, otherwise generate new captcha.
4. Repeat step 3 until code matched.

Code

##########################start.html ####################

```
<! DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title> Captcha test starting page </title>
<meta name="description" content="Testing captcha ">
<meta name="keywords" content=" testing captcha ">
<style type="text/css">
<!--
.style1 {
        color: #000066;
        font-family: Verdana, Arial, Helvetica, sans-serif;
}
.style2 {color: #000066; font-family: Verdana, Arial, Helvetica, sans-serif; font-weight:
bold; }
a:visited {
        color: #660000;
        text-decoration: underline;
}
a:link {
        text-decoration: underline;
}
a:hover {
        text-decoration: none;
        color: #CC0000;
}
a:active {
        text-decoration: underline;
}
-->
</style>
</head>
```

```
<body>
<div align="center">
  <h1 class="style2"><br>
  Captcha example </h1>
  <blockquote><blockquote><blockquote>
    <h3 align="left" class="style1"><br>
        <a href="captcha-page.php">Start the captcha sequence here </a></h3>
    <p align="left" class="style1"> </p>
    <p align="left" class="style1"> </p>
    <p align="left" class="style1"> </p>
      </blockquote>
    </blockquote>
  </blockquote>
</div>
</body>
</html>
```

############################end.html ####################

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>Captcha test end page </title>
<meta name="description" content="Testing captcha ">
<meta name="keywords" content=" testing captcha ">
<style type="text/css">
<!--
.style1 {
        color: #000066;
        font-family: Verdana, Arial, Helvetica, sans-serif;
}
.style2 {color: #000066; font-family: Verdana, Arial, Helvetica, sans-serif; font-weight:
bold; }
a:visited {
        color: #660000;
        text-decoration: underline;
}
a:link {
        text-decoration: underline;
}
a:hover {
        text-decoration: none;
        color: #CC0000;
}
a:active {
```

```
        text-decoration: underline;
}
-->
</style>
</head>
<body>
<div align="center">
  <h1 class="style2"><br>
  Captcha example end </h1>
  <blockquote><blockquote><blockquote>
    <h3 align="left" class="style1"><br>
       if you have landed here then everything worked fine.</h3>
    <p align="left" class="style1">Now go back to where you left off on the <a
href="../captcha-explained.html">captcha explained</a> page you were reading.  </p>
    <p align="left" class="style1"> </p>
    <p align="left" class="style1"> </p>
    <p align="left" class="style1"> </p>
      </blockquote>
    </blockquote>
  </blockquote>
</div>
</body>
</html>
```

captcha.php

```
<?php
session_start();
header("Content-type: image/png");
$image = imagecreate(60,20);
$background_color = ImageColorAllocate($image, rand(64,70), rand(70,73),
rand(100,110));
$colour1 = ImageColorAllocate($image, rand(121,255), rand(126,255), rand(197,255));
$colour2 = ImageColorAllocate($image, rand(100,255), rand(120,255), rand(105,255));
$colour3 = ImageColorAllocate($image, rand(131,255), rand(137,255), rand(100,255));
imagestring($image,5,8,2,$_SESSION["captcha"],$colour1);
imagestring($image,5,8,2,$_SESSION["captcha"],$colour2);
imagestring($image,5,8,2,$_SESSION["captcha"],$colour3);
imageline($image, rand(1,300), rand(1,3), rand(10,150), rand(0,150), $colour1);
imageline($image, rand(1,300), rand(1,3), rand(10,150), rand(0,150), $colour2);
imageline($image, rand(1,300), rand(1,3), rand(10,150), rand(0,150), $colour3);
imageline($image, 10, 0, 14, 10, $colour1);
imageline($image, 20, 0, 24, 30, $colour2);
imageline($image, 55, 0, 34, 60, $colour3);
imagepng($image);
imagedestroy($image);
?>
```

########################captcha-page.php ####################

```
<?php
session_start();
$PHP_SELF = $_SERVER['PHP_SELF'];
$errore = 0;
if(isset($_POST['submit']) && isset($_SESSION['captcha'])) {
	if(isset($_POST['code'])){
		$_code = $_POST['code'];
		if($_SESSION['captcha'] == $_code){
			header("Location: end.html");
			exit;
		} else {
			$errore = 1;
		}
	} else {
		$errore = 1;
	}
}
if(!isset($_SESSION['captcha'])){session_register('captcha');}
$PHP_SELF = $_SERVER['PHP_SELF'];
$stringa = '';
$cifre = 5;
for($i=1;$i<=$cifre;$i++){
	$letteraOnumero = rand(1,2);
	if($letteraOnumero == 1){
		// lettera
		$lettere = 'ABEFHKMNRVWX';
		$x = rand(1,11);
		$lettera = substr($lettere,$x,1);
		$stringa .= $lettera;
	} else {
		$numero = rand(3,7);
		$stringa .= $numero;
	}
}
$_SESSION['captcha'] = $stringa;
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
<title>Captcha entrance page</title>
<style media="all" type="text/css">
#captcha{margin:0 auto;text-align:center}
.errore{color:#f00}
```

```
.style2 {
	font-size: 14px;
	font-family: Verdana, Arial, Helvetica, sans-serif;
}
.style3 {font-size: 14px; font-family: Verdana, Arial, Helvetica, sans-serif; color:
#000099; }
</style>
</head><body>
<div id="captcha">
	<p><img src="captcha.php" />
</p>
  <p class="style3"><strong>To proceed to the web form please<br />
    insert the code you see above into the Code field below<br />
    and then click the enter button.
  </strong> <br />
    [ any letters you see should be inserted as CAPITALS and if you<br />
    have difficulty in reading the image above then refresh your browser <br />
    a few times until you see an image that is clear enough to copy
  ]  </p>
  <p class="style2">If you get the code wrong, you will be allerted and can try again </p>
  <form id="captchaform" action="<?php echo $PHP_SELF ?>" method="post">
		<p><label for="code">Code: </label><input type="text" name="code"
id="code" /></p>
		<p><input type="submit" id="submit" name="submit" value="Enter"
/></p>
  </form>
	<?php if($errore == 1){ ?>
  <p class="errore">The code you inserted was not correct .<br />
	<a href="<?php echo $PHP_SELF ?>">Click here to generate a new Captcha
code and try again</a></p>
	<?php } ?>
</div>
</body></html>
```

Conclusion

Hence, we have studied as well as generated and verified CAPTCHA image.