Atitit 图片验证码功能设计文档总结

目录

[1.1. 使用图片验证码img src标签设置图片。。验证码图片有png，jpg，svg等格式。。 1](#_Toc27906)

[1.2. Php png图像 1](#_Toc22875)

[1.3. Php的svg图片生成 2](#_Toc10016)

[2. Cde 3](#_Toc23120)

## 使用图片验证码img src标签设置图片。。验证码图片有png，jpg，svg等格式。。

Svg的head是这样header('Content-Type:image/svg+xml');

大概流程是生成字串放入session。。提交的时候校验session字串即可。。。

## Php png图像

<?php  
// /sdk/catchImg.php  
require *\_\_DIR\_\_* . '/../vendor/autoload.php';  
use Minho\Captcha\CaptchaBuilder;  
  
session\_start();  
$captch = new CaptchaBuilder();  
  
$captch->initialize([  
 'width' => 120, // 宽度  
 'height' => 40, // 高度  
 'line' => false, // 直线  
 'curve' => false, // 曲线  
 'noise' => 0, // 噪点背景  
 'fonts' => [] // 字体  
]);  
// C:\wamp\bin\php\php5.6.31\php.exe composer.phar install  
//C:\wamp\bin\php\php5.6.31\php.exe composer.phar update  
$captch->create();  
$captch->save('778.png',1);  
$\_SESSION['captch77'] = $captch->getText();  
  
error\_log('$\_SESSION[captch77] from ctimg.php '.$\_SESSION['captch77']);  
  
$captch->output(1);  
//echo $captch->getText();

## Php的svg图片生成

Pnp的png图像生成测试环境遇到点问题，需要集成gd库，也依赖于os的图像库。。

所以可以使用svg格式。。可惜php没有好的svg库，，只要使用nodejs的svg库

svg-captcha

通过cli模式调用。。。

# Cde

<?php  
   
header('Content-Type:image/svg+xml');  
session\_start();  
  
// C:\wamp\bin\php\php5.6.31\php.exe composer.phar install  
//C:\wamp\bin\php\php5.6.31\php.exe composer.phar update  
$jscmd = "node cpt2.js "; // . base64\_encode(json\_encode($post));  
error\_log( $jscmd . *PHP\_EOL*,3,'error\_log.log');  
$returnContent = exec($jscmd);  
error\_log( '$returnContent:'.$returnContent.*PHP\_EOL*,3,'error\_log.log');  
  
$jsonObj=json\_decode($returnContent,true);  
$\_SESSION['captch77'] = strtolower( $jsonObj['text']);  
  
error\_log('$\_SESSION[captch77] from ctimg.php is:: '.$\_SESSION['captch77'].*PHP\_EOL*,3,'error\_log.log');  
echo $jsonObj['data'];

const code = require("svg-captcha");  
 var ret=code.create({  
 size: 4,  
 ignoreChars: "0o1iIl",  
 noise: 0,  
 color: true,  
 background: "#fff",  
 fontSize: 60  
});  
***console***.log(***JSON***.stringify(ret) )