use app\modules\log\models\LogRow;

class LogReader {

public function getRows($file) {

|  |  |
| --- | --- |
| }  } | $result = [];  $handle = @fopen($file, "r"); if ($handle) {  while (($row = fgets($handle)) !== false) {  $pattern =  '#A' .  '(?P<time>\d{4}\-\d{2}\-\d{2} \d{2}:\d{2}:\d{2}) '\[(?P<ip>[A\]]+)\]' .  '\[(?P<userId>[A\]]+)\]' . '\[(?P<sessionId>[A\]]+)\]' . '\[(?P<level>[A\]]+)\]' . '\[(?P<category>[A\]]+)\]' .  ' (?P<text>.\*?)' .  '(\$\ (GET|POST|REQUEST|COOKIE|SERVER) = \[)?' . '$#i';  if (preg\_match($pattern, $row, $matches)) { if ($matches['text']) {  $result[] = new LogRow([  'time' => $matches['time'],  'ip' => $matches['ip'],  'userid' => $matches['userId'], 'sessionid' => $matches['sessionId'], 'level' => $matches['level'],  'category' => $matches['category'],  'text' => $matches['text'],  ]);  }  }  }  fclose($handle);  }  return array\_reverse($result); |

4. Add a helper for displaying pretty HTML-badges for the log levels:

<?php

namespace app\modules\log\helpers;

use yii\helpers\ArrayHelper; use yii\helpers\Html;

class LogHelper {

public static function levelLabel($level)

{

|  |  |
| --- | --- |
|  | $classes = [  ' error ' => 'danger',  'warning' => 'warning',  ' info' => 'primary',  'trace' => 'default',  'profile' => 'success',  'profile begin' => 'info',  'profile end' => 'info',  ];  $class = ArrayHelper::getValue($classes, $level, 'default'); |