

```
1  #!/usr/bin/php
2  <?php
3
4  // Set error reporting level
5  error_reporting(E_ALL & ~E_NOTICE & ~E_WARNING);
6
7  // Set working directory
8  chdir('/home/alvaro');
9
10 // Welcome message
11 echo "\n#####\n\n";
12 echo "##### MÀQUINA DE RECORDS by Àlvaro Cànovas #####\n\n";
13 echo "#####\n\n";
14 sleep(1);
15 echo "> Stop script with SIGINT [Ctrl+C] \n\n";
16 echo "> Starting... dir = " . getcwd() . "\n\n\n";
17 echo " _____\n\n\n";
18
19 // PDF generator tools
20 require('aux/TCPDF/tcpdf.php');
21 // $tcpdf = new TCPDF(PDF_PAGE_ORIENTATION, PDF_UNIT, PDF_PAGE_FORMAT, true, 'UTF-8', false);
22
23 // Get next image id iterator
24 $i = get_next_record_iterator();
25 while(TRUE)
26 {
27     // Select working mode randomly [5%]
28     $mode = rand(0,19);
29
30     if($mode >= -9999) // Capture 'record' working mode
31     {
32         // Generate filename
33         $filename = "record_" . $i . ".jpg";
34         echo "Click!\n\n";
35         // Capture image
36         shell_exec("raspistill -n -t 1 -w 2160 -h 3439 -o " . $filename);
37         $filesize = number_format(((filesize($filename) / 1024) / 1024), 2, ".", "");
38         echo "Capturing '$filename' [" . $filesize . "MB]\n\n";
39         if(!file_exists($filename)) { break; } // Errors found
40
41         // Generate pdf document in this directory
42         $docname = "doc_" . $i . ".pdf";
43         $tcpdf = new TCPDF(PDF_PAGE_ORIENTATION, PDF_UNIT, "A6", true, 'UTF-8', false); // initialize TCPDF
44         $pdfcontent = generate_document($tcpdf, $i, $filename, $filesize);
45         if(!file_put_contents($docname, $pdfcontent, LOCK_EX)) { break; } // Errors found
46
47         // Remove captured image once the document is created
48         if(!unlink($filename)) { break; } // Errors found
49
50         $docsize = number_format(((filesize($docname) / 1024) / 1024), 2, ".", "");
```

```

51     echo "Generating document '$docname' [" . $docsize . "MB]\n\n";
52
53     $dirsize = shell_exec("du -s HDD/ | cut -f 1");
54     $freespace = disk_free_space("HDD/");
55
56     echo "HDD free space: " . number_format(($freespace / 1024) / 1024), 2, ".", "") . " MB\n\n";
57
58     if($freespace < 536870912) { // 512 MB
59         // print 'record' document via CUPS
60         $printer = "[WIFI]_Brother_DCP-1610W"; // set printer
61         echo "Printing captured image via " . $printer . "\n\n";
62         shell_exec("lp -d " . $printer . " -o media=A6 -o fit-to-page " . $docname);
63         // shell_exec("lp " . $docname);
64
65         // specify destination folder
66         $outdir = "printed/";
67     }
68     else {
69         // move 'record' document to HDD
70         echo "Moving captured image to the Hard Drive\n\n";
71         // specify destination folder
72         $outdir = "HDD/";
73     }
74
75     // move document to specified folder [HDD/ or printed/]
76     if(!rename($docname, $outdir . $docname)) { break; } // Errors found
77
78     // Increment iterator
79     $i += 1;
80 }
81 else // Retrieve 'record' working mode
82 {
83     echo "Retrieving 'record'\n\n";
84     echo "Place the document inside the scanner\n\n";
85     $wait = 12; // seconds
86     echo str_repeat("-", $wait) . "\n";
87     for($j = 0; $j < $wait; $j++) { echo "#"; sleep(1); }
88     echo "\n\n";
89
90     // Start scanning
91     echo "Scanning...\n\n";
92     shell_exec("scanimage --format=tiff | convert tiff:- scan.pdf");
93
94     $replace_id = rand(1, get_last_HDD_iterator()); // Get 'record' id that will be replaced
95     // Move scanned 'record' document to the HDD replacing an old 'record'
96     echo "Replacing document containing 'record " . $replace_id . "' with the scanned one\n\n";
97     if(!rename("scan.pdf", "HDD/doc_" . $replace_id . ".pdf")) { break; } // Errors found
98 }
99
100 // Sleep

```

```

101     $time = rand(0,30); // seconds
102     echo "I am going to sleep for " . $time . " seconds.\n\n";
103     echo str_repeat("-", $time) . "\n";
104     for($t = 0; $t < $time; $t++) {
105         echo "#";
106         sleep(1); // 1 second
107     }
108     if($time != 0) { echo "\n\n"; }
109     echo " _____ \n\n\n";
110 }
111
112 // Some error caused the loop to break
113 echo "\n\n\n> ERROR!\n\n\n";
114
115 ?>
116
117 <?php
118 function get_next_record_iterator()
119 {
120     $lastfileHDD = shell_exec("ls -t HDD/ | grep 'doc_' | head -n 1");
121     $lastfilePrinted = shell_exec("ls -t printed/ | grep 'doc_' | head -n 1");
122     $iteratorHDD = str_replace(".pdf", "", substr($lastfileHDD, 4));
123     $iteratorPrinted = str_replace(".pdf", "", substr($lastfilePrinted, 4));
124
125     $iterator = ($iteratorHDD > $iteratorPrinted) ? $iteratorHDD : $iteratorPrinted;
126
127     return $iterator + 1;
128 }
129 ?>
130
131 <?php
132 function get_last_HDD_iterator()
133 {
134     $lastfile = shell_exec("ls -t HDD/ | grep 'doc_' | head -n 1");
135     $iterator = str_replace(".pdf", "", substr($lastfile, 4));
136
137     return $iterator;
138 }
139 ?>
140
141 <?php
142 function generate_document($tcpdf, $i, $filename, $filesize)
143 {
144     // var $tcpdf is instantiated previously
145
146     $date = date('d-m-Y/ H:i:s/ ');
147
148     $htmlcontent = "
149 <!DOCTYPE html>
150 <html>

```

```

151 <head>
152 </head>
153 <body style="text-align: right;">
154     
155     <span style="display: block; font-size: 15px; color: #494949; padding-top: 485px;">
156         Record_ " . $i . ".jpg
157         <span style="font-size: 7px; color: #666666;"> " . $date . $filesize*1024 . "KB/ 41.399469, 2.190975</span>
158     </span>
159 </body>
160 </html>\n";
161
162 // set default monospaced font
163 $tcpdf->SetDefaultMonospacedFont(PDF_FONT_MONOSPACED);
164
165 // set title of pdf
166 $tcpdf->SetTitle('Maquina de Records - Record ' . $i);
167 $tcpdf->SetAuthor('Alvaro Canovas');
168
169 // set margins
170 $tcpdf->SetMargins(16, 4, 4, true);
171 $tcpdf->SetHeaderMargin(PDF_MARGIN_HEADER);
172 $tcpdf->SetFooterMargin(PDF_MARGIN_FOOTER);
173
174 // set header and footer in pdf
175 $tcpdf->setPrintHeader(false);
176 $tcpdf->setPrintFooter(false);
177 $tcpdf->setListIndentWidth(3);
178
179 // set auto page breaks
180 $tcpdf->SetAutoPageBreak(false, 0);
181
182 // set image scale factor
183 $tcpdf->setImageScale(PDF_IMAGE_SCALE_RATIO);
184
185 $tcpdf->AddPage('P', 'A6');
186
187 $tcpdf->SetFont('helvetica', '', 13);
188
189 $tcpdf->writeHTML($htmlcontent, true, false, false, false, '');
190
191 //Close and output PDF document
192 $doc = $tcpdf->Output('demo.pdf', 'S');
193
194 return $doc;
195
196 }
197
198 ?>
199
200

```