

Fun Shoot Web

Fun Shoot Web ist eine Weiterentwicklung des Windows Programms Fun Shoot. Das Programm kann verwendet werden, um mit Meyton Schießanlagen auch mal ein Späßschießen durchführen zu können. Die Funktionsweise ist relativ einfach. Es wird eine Graphik benötigt, bei der jede Farbe einen anderen Punktwert bedeutet. Hierbei müssen folgende Rahmenbedingungen eingehalten werden:

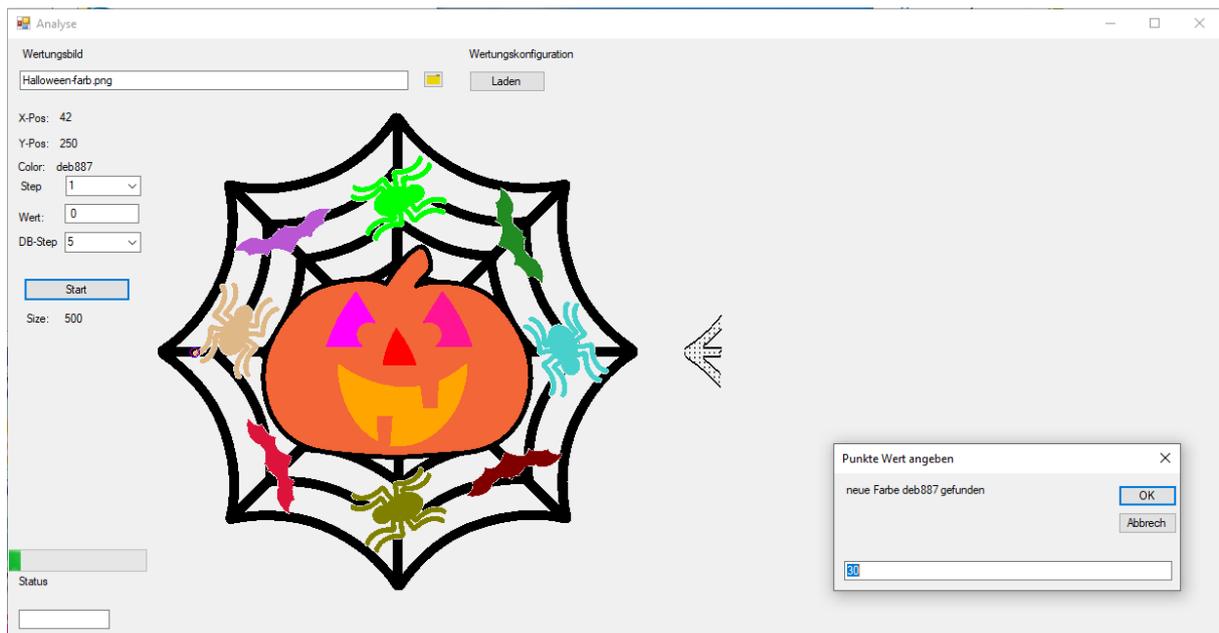
- Das Dateiformat muss eine verlustfreie Komprimierung unterstützen (z.B. png)
- Die Datei muss Quadratisch sein
- Ein Pixel muss für LG/LP 0,25 mm entsprechen
- Die Datei muss 500x500 Pixel (125x125mm) haben (kleinere Formate in Entwicklung)
- Die Datei sollte keine Zwischenfarben enthalten (Bei Konvertierung von SVG machen die meisten Programme antialiasing. Daher verwende ich hier Coreldraw)
- Es kann/sollte eine zweite Datei geben, die zum Ausdruck und Anzeige auf dem Bildschirm genutzt wird.

Die farbkodierte Graphik wird dann durch ein Analysetool für die Datenbank aufbereitet. Das Programm FunShot-Analyse läuft auf Windows Rechnern und benötigt eine aktuelle .NET Installation.

Nach dem Start wird über den Folder Button eine Datei ausgewählt. Über Laden wird diese dann in das Programm importiert. Anpassung der Werte:

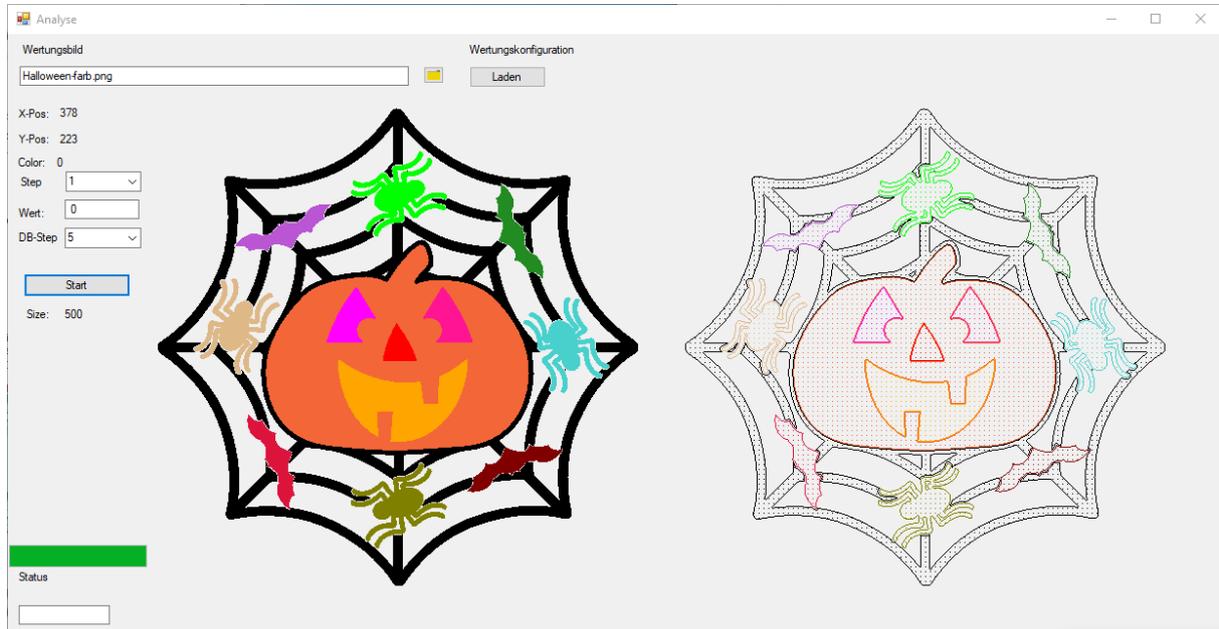
Step: Sollte nur auf sehr langsamen Rechnern größer 1 sein. Hier könne Pixel bei der Analyse übersprungen werden.

DB-Step: Bei Farbflächen werden nicht alle Punkte in die Datenbank eingetragen. Über DB-Step kann angegeben, wie groß der Abstand zwischen Punkten in der Datenbank maximal sein kann. Ein kleiner Wert ist nur in sehr wenigen Fällen notwendig (Bei Punkt Auswertungen). Bei den meisten Grafiken ist 5 ein sehr guter Wert. Bei größeren Graphiken (z.B. Quadraten) kann auf 10 erhöht werden, ohne die Genauigkeit der Auswertung negativ zu beeinflussen.



Anschließend wird die Auswertung über Start begonnen.

Jedes mal, wenn ein neuer Farbwert gefunden wird, fragt das Programm nach den Punktwert für diese Farbe. Über einen kleinen Kreis wird auch angezeigt, wo diese Farbe gefunden wurde. Die Werte werden auch in eine ini Datei eingetragen, so dass bei einem erneuten Durchlauf die bereits einmal eingegebenen Werte wieder als Vorschlag erscheinen.

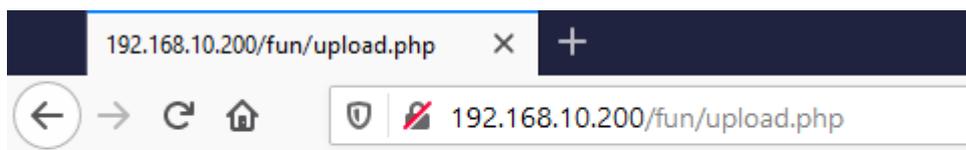


Als Ergebnis werden im Verzeichnis der Graphik zwei weitere Dateien gespeichert

Dateiname.ini = Zuordnung Farben zu Punkten

Dateiname.txt = Import Datei für Datenbank

Nun können die Dateien zum Server hochgeladen werden. Hierzu ruft man die Adresse <http://192.168.10.200/fun/upload.php> auf. Zuerst sind die Anzuzeigende Bilddatei und die DB Datei auszuwählen. Anschließend die Bildnummer, unter der da Bild in der DB eingetragen werden soll. (Die maximale Anzahl in der Config Datei auf dem Server angepasst werden)



Upload FunShoot

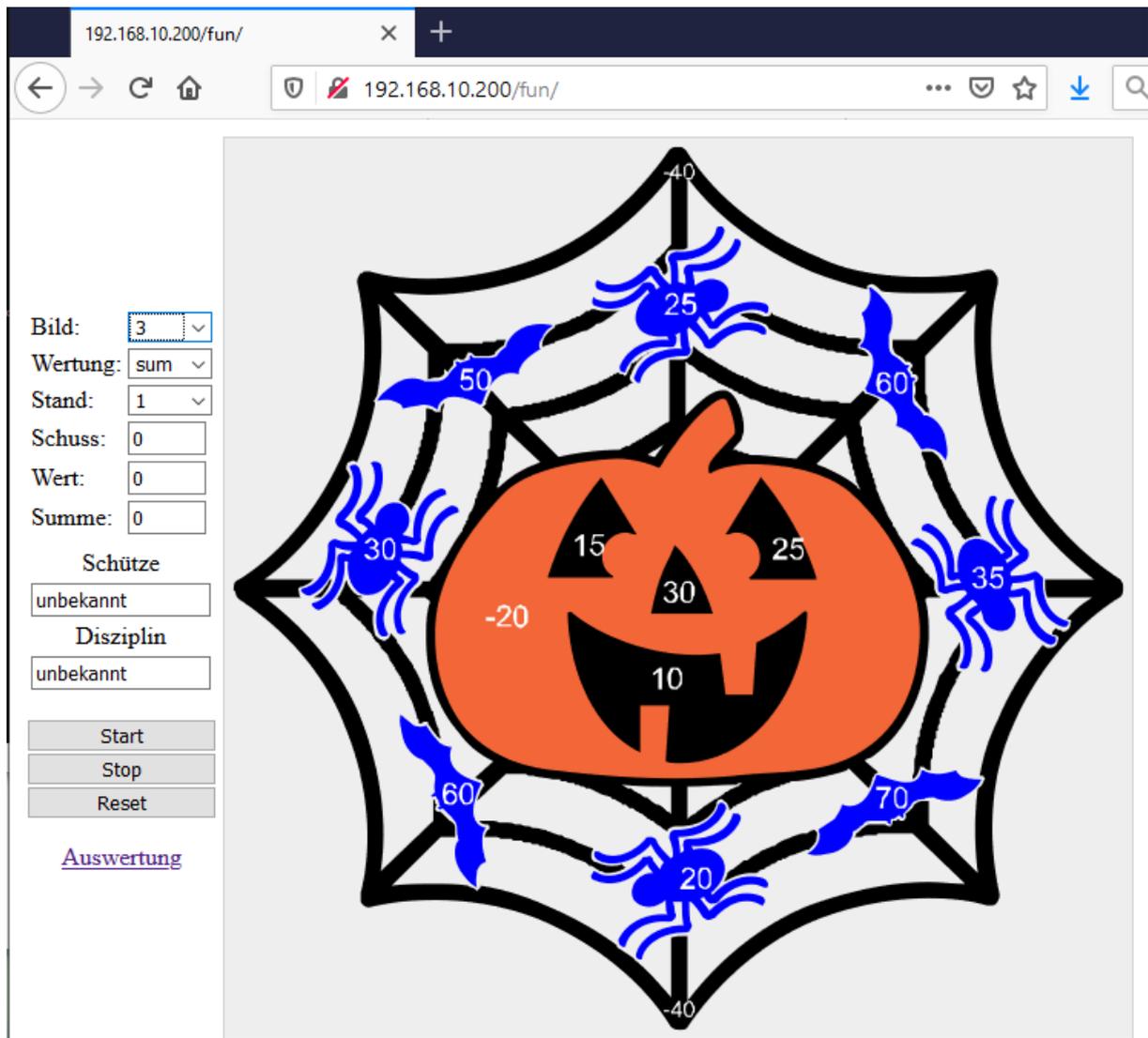
Bilddatei (png/svg): Halloween.svg

DB-datei (txt): Halloween-farb.png.txt

Bild Nr:

Beim Klick auf Verarbeiten werden die Dateien an die entsprechenden Stellen kopiert und die Werte in die Datenbank eingearbeitet. Alte Werten zu dieser Bildnummer werden vorher aus der Datenbank gelöscht.

Damit kann das Bild auch schon in der Applikation genutzt werden



Um das Schießen selbst durchzuführen, müssen folgende Einstellungen vorgenommen werden:

Bild: Welche Bildnummer soll für das Spaßschießen verwendet werden?

Wertung:

sum: Alle Punkte der vom Schuss berührten Flächen werden summiert.

max: Der Maximale Punktzahl der vom Schuss berührten Fläche wird ermittelt.

avg: Der Mittelwert aller vom Schuss berührten Flächen wird ermittelt

point: Der Punktwert der Fläche des Schusszentrums wird ermittelt.

Stand: Die Schüsse welches Standes sollen angezeigt werden?

Die Bilder mit Nummer 1 bis 89 können benutzerdefiniert per upload zum Server hochgeladen werden. Die Bilder 90-100 sind zur Programminternen Verwendung. Die Bilder 101 – 150 sind automatisch generierte 10er Karos. Die Bilder 151 - 200 sind automatisch generierte 15er Karos. Die Bilder 101 – 200 liegen nicht auf dem Server, sondern werden aus der Datenbank generiert.

Die Ergebnisse können dann natürlich auch für das gesamte Schießen ausgewertet werden. Hierzu wird auf Auswertung geklickt, oder die URL <http://192.168.10.200/fun/auswertung.html> angegeben.

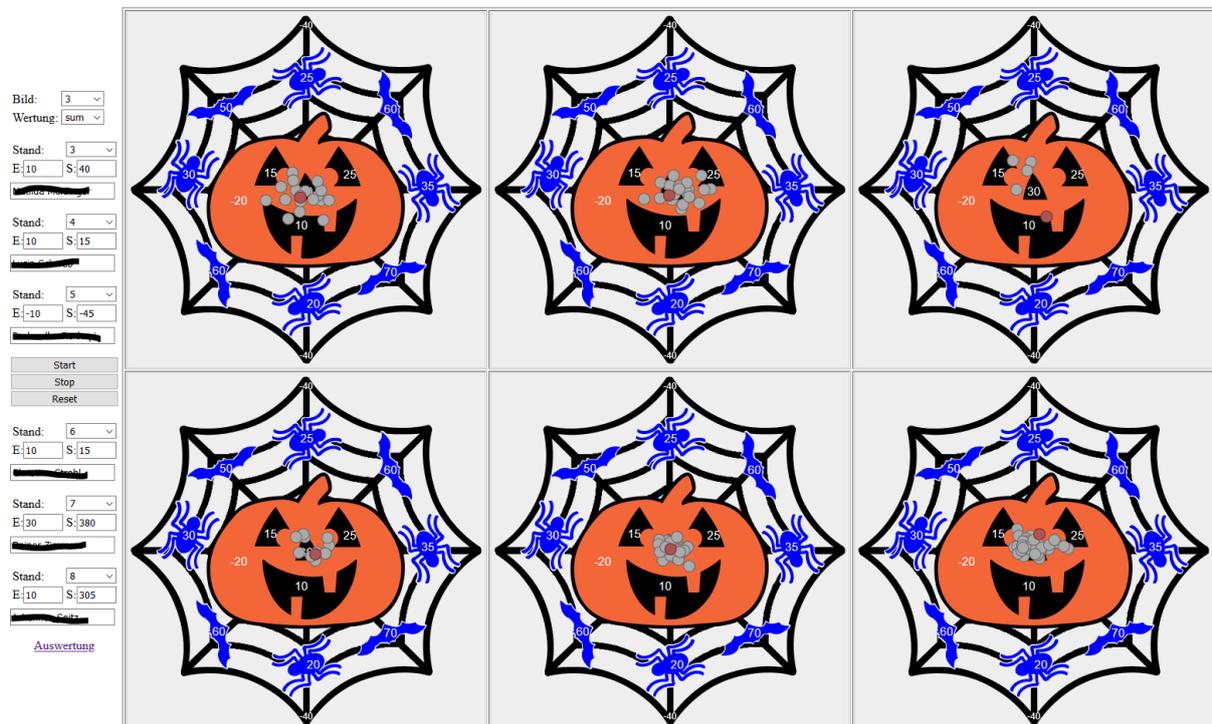
Hier werden alle Disziplinen und die letzten 10 Tage, von denen Ergebnisse in der SSMDB2 zu finden sind in der Auswahl angezeigt. Anschließend müssen noch Bild und Wertung analog zum Schießen

selbst ausgewählt werden. Wird als Format HTML ausgewählt, wird eine Tabelle im Browser angezeigt. Wird CSV als Format angezeigt, bekommt man eine CSV Datei zum Download, die anschließend in Excel / LibreOffice weiterbearbeitet werden kann.

Name	Disziplin	ScheibenID	Zeitstempel	Einzel	Gesamt
[REDACTED]	LP20 Training	-479816694	2019-01-11 19:09:46	24 + 6 + + -2 + 14 + 9 + -1	50
[REDACTED]	LP20 Training	-563879916	2018-10-05 19:16:58	21 + 14 + 8 + 20 + 26 + 21 + 6 + 16 + 12 + 19 + 0 + 12 + -10 + -10 + 15 + 11 + 1 + 2 + 45 + 25	254
[REDACTED]	LP20 Training	-563531774	2018-10-05 19:12:04	1 + -10 + 11 + -10 + 2 + -10 + -10 + 1 + 11 + 11 + -10 + 11 + 21 + 25 + 12 + -10 + 14 + 11 + 11 + 12	94
[REDACTED]	LP20 Training	-565230574	2018-10-05 18:48:24	9 + 9 + 10 + -10 + 15 + 31 + 9 + 10 + 10 + 5 + -2 + 19 + -5 + 6 + 11 + -10 + 12 + + 4 + 1	134

Im Folgenden sind die einzelnen Bilder in der Breite von 12,5 cm aufgeführt, so dass diese auf Scheiben ausgedruckt werden können. Hierzu können Die Ausdrücke auf bereits genutzte Scheiben aufgeklebt werden, oder auf die Rückseite von Trainings Scheiben ohne Loch (Krüger 6305B) ausgedruckt werden.

Es gibt auch die Möglichkeit 2 oder 6 Stände gleichzeitig anzuzeigen



Hierzu muss nur die URL erweitert werden /fun/?layout=6

Dart

Das Dart Spiel kann von bis zu 4 Schützen gespielt werden. Es funktioniert nach den Standard Dart Regeln (bis auf Over).

Oben Links kann das Spiel ausgewählt werden. Hier stehen 301, 501, 701, 1001 und Cricket zur Auswahl.

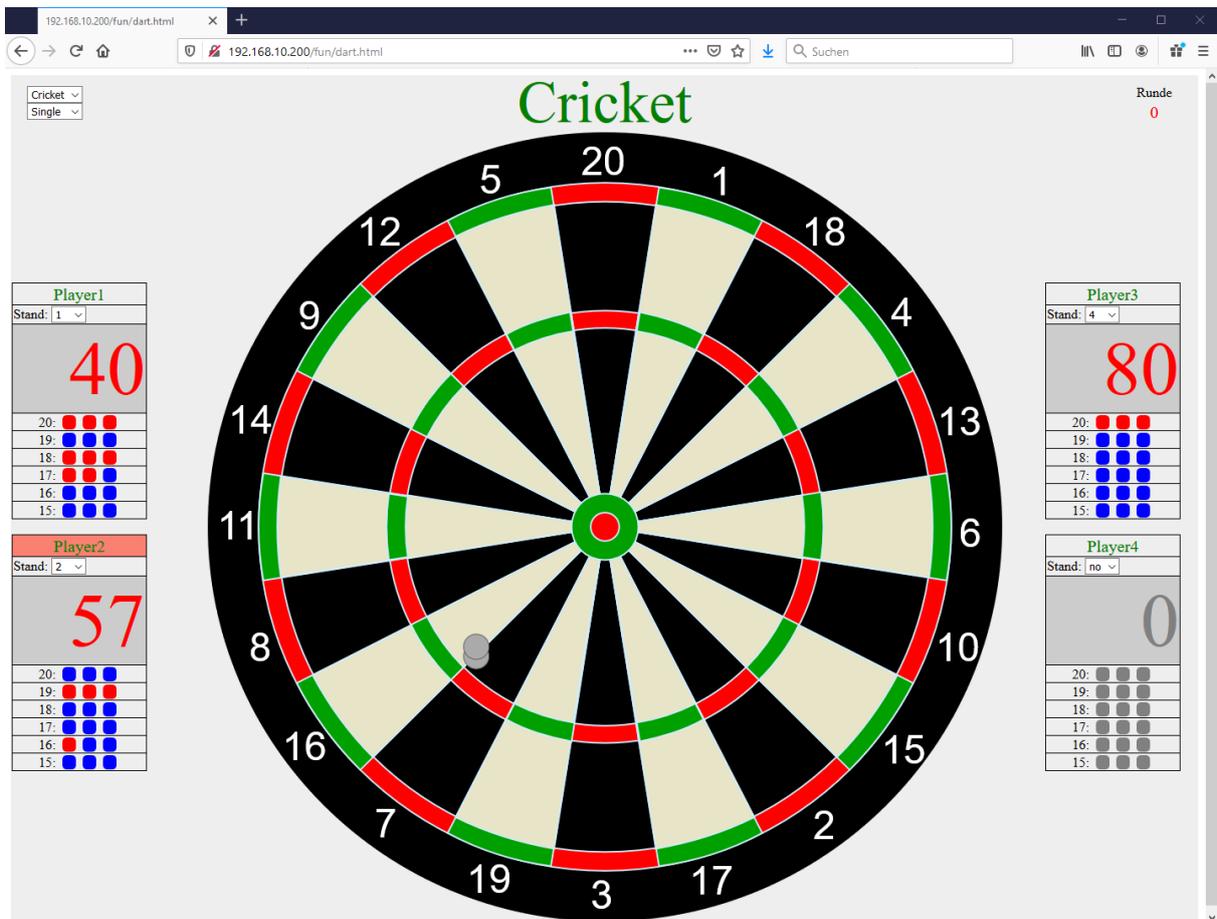
Darunter kann Ausgewählt werden, wie das Spiel beendet werden kann.

Over: Es müssen 0 oder weniger auf dem Zähler sein

Single: Der Zähler muss genau auf 0 heruntergeschossen werden

Double: Der Zähler muss genau auf 0 sein und mit einem Double ausgeschossen werden

Master: Der Zähler muss genau auf 0 sein und mit einem Double oder Triple ausgeschossen werden



Das Spiel wird immer wieder auf Anfang zurückgesetzt, wenn das Spiel, oder ein Spieler geändert wird.

Installationsanleitung auf Meyton Server (OpenSUSE 15.1)

Variante Script:

Kopieren der Dateien nach /tmp

```
prepare_fun.sh
```

```
funshoot.tgz
```

```
chmod 755 prepare_fun.sh
```

Variante Manuell:

```
zypper install php7 php7-mysql php7-fastcgi php7-fpm
```

Installed:

```
php7 php7-ctype php7-dom php7-fastcgi php7-iconv php7-json php7-mysql php7-pdo php7-sqlite  
php7-tokenizer php7-xmlreader php7-xmlwriter system-user-wwwrun php7-fpm
```

In Datei /etc/php7/fastcgi/php.ini

```
cgi.fix_pathinfo=1
```

```
/etc/php7/fpm # mv php-fpm.conf.default php-fpm.conf
```

```
/etc/php7/fpm/php-fpm.d # mv www.conf.default www.conf
```

In /etc/lighttpd/modules.conf

```
include "conf.d/fastcgi.conf"
```

In Datei /etc/lighttpd/conf.d/fastcgi.conf

```
fastcgi.server = ( ".php" =>  
    ( "php-local" =>  
        (  
            "socket" => socket_dir + "/php-fastcgi-1.socket",  
            "bin-path" => server_root + "/cgi-bin/php",  
            "max-procs" => 1,  
            "broken-scriptfilename" => "enable",  
        ),  
    ),  
    "php-tcp" =>  
        (  
            "host" => "127.0.0.1",  
            "port" => 9999,  
            "check-local" => "disable",  
            "broken-scriptfilename" => "enable",  
        ),  
    ),  
    "php-num-procs" =>  
        (  
            "socket" => socket_dir + "/php-fastcgi-2.socket",  
            "bin-path" => server_root + "/cgi-bin/php",  
            "bin-environment" => (  
                "PHP_FCGI_CHILDREN" => "16",  
                "PHP_FCGI_MAX_REQUESTS" => "10000",  
            ),  
            "max-procs" => 5,  
            "broken-scriptfilename" => "enable",  
        ),  
    ),  
),  
)
```

```
systemctl restart lighttpd
```

```
systemctl restart php-fpm
```

Entpacken des Tar Files unter /srv/www/htdocs/lana/fun

Datenbank einrichten:

```
echo "create database funshoot" | mysql -u root -pmc4hct
mysql -u root -pmc4hct funshoot < funshoot.sql
echo "grant ALL PRIVILEGES ON funshoot.* to 'meyton'@'localhost' identified by
'mc4hct'" | mysql -u root -pmc4hct
echo "flush privileges" | mysql -u root -pmc4hct funshoot
```

Beschreibung der Dateien:

index.php

Hauptdatei: Hier wird die Webseite inklusive der Javascript Inhalte erstellt.

Aufruf: <http://192.168.10.200/fun/index.php>

Parameter:

<http://192.168.10.200/fun/index.php?STAND=2> (Es wird automatisch Stand2 ausgewählt)

<http://192.168.10.200/fun/index.php?LAYOUT=2> (Anzahl Scheiben 1|2|6 sind möglich)

<http://192.168.10.200/fun/index.php?BILD=2> (Es wird automatisch Bild 2 ausgewählt)

config.inc.php

Konfigurationsdatei. Hier werden die Verbindungsparameter zu den Datenbanken. Parameter, die am häufigsten geändert werden müssen:

PICANZ=6 (Kann erhöht werden, wenn eigene Daten hochgeladen werden)

MAXSTAND=15 (Wie viele Stände sollen zur Auswahl stehen)

STARTSTAND=3 (Welcher Stand soll beim Start des Programmes ausgewählt sein)

PIXELSIZE=25 (Pixelgrößer in hundertstel mm)

PIXELSIZE 25: Bild 125mm

Pixelsize 50: Bild 250mm

Pixelsize 70: Bild 350mm

PIXELSHOOT=18 (Abdeckungsrastrer Schuss (11 oder 18 sind möglich))

Sinnvolle Kombinationen PIXELSIZE/PIXELSHOOT: LG 25/18, 9mm 50/18, KK 50/11, 7,62mm 70/11

functions.inc.php

Funktionen, die von mehreren Skripten genutzt werden. Insbesondere sind hier die Funktionen untergebracht, die den Schusswert ermitteln

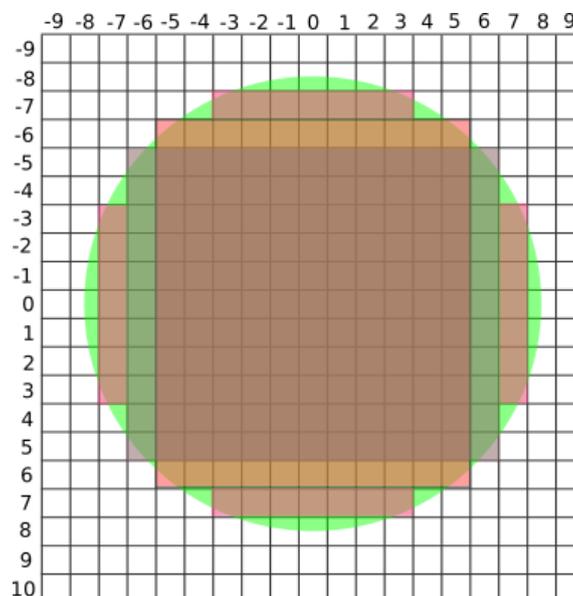
getimage.php

Dieses Script liefert die Bilder aus. Bei den benutzerdefinierten Bildern werden diese aus dem Dateisystem genommen. Bei den automatisch generierten Bildern (> 100) werden die Daten aus der Datenbanktabelle transval genommen und eine svg Datei daraus generiert. Die Bilder können auch im separat im Browser ausgegeben werden <http://192.168.10.200/fun/getimage.php?ID=103>

schussdata.php

Wird einmal pro Sekunde aus dem Javascript abgefragt, ob neue Schussdaten vorhanden sind. Wird keine Option SCHUSS mitgegeben, werden alle Schüsse der Scheibe ausgeliefert. Wird der Parameter Schuss mit angegeben, wird nur etwas zurückgeliefert, wenn ein Schuss mit höherer Nummer zu diesem Stand da ist.

Der Die Auswertung des Schusses erfolgt dabei über folgende Abdeckung, wenn in der Datei config-inc.php \$PIXELSHOOT=18 angegeben wird:



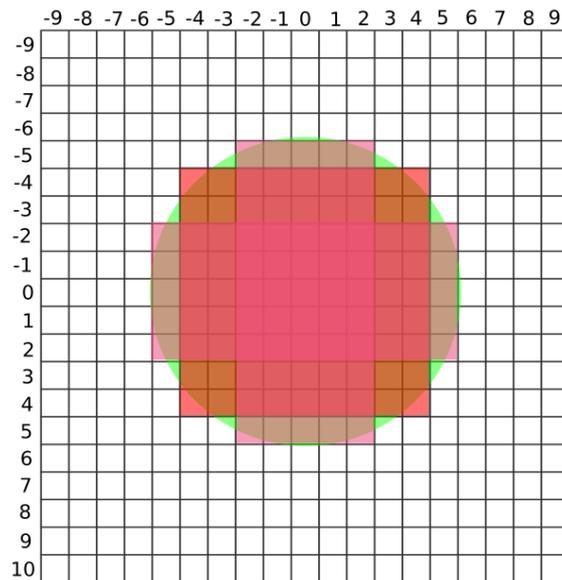
Der Grüne Kreis stellt einen Diabolo Schuss mit 4,5mm (16*0,25mm) dar. Die Flächen darüber, wie über den SQL Befehl die Datenpunkte ermittelt werden. Ein SQL Befehl sieht dann wie folgt aus:

```
select sum(distinct value) as wert from picval where ( (x>=236 and y>=217 and x<=246 and y<=229)
or (x>=235 and y>=218 and x<=247 and y<=228) or (x>=238 and x<=244 and y=216) or (x>=238 and
x<=244 and y=230) or (y>=220 and y<=226 and x=234) or (y>=220 and y<=226 and x=248) ) and
pID=6
```

Im Falle einer Automatisch generierten Scheibe:

```
select sum(distinct tval) as wert from picval inner join transval on picval.pID = transval.pID and
picval.value = transval.pval where tID=101 and ( (x>=305 and y>=124 and x<=315 and y<=136) or
(x>=304 and y>=125 and x<=316 and y<=135) or (x>=307 and x<=313 and y=123) or (x>=307 and
x<=313 and y=137) or (y>=127 and y<=133 and x=303) or (y>=127 and y<=133 and x=317) )
```

Bei \$PIXELSHOOT=11 ergibt sich folgende Abdeckung:



auswertung.html

Bereitstellung der Auswahlmöglichkeiten für die Auswertung.

Wird eine Disziplin ausgewählt, werden unter Datum nur noch die Daten angezeigt, an denen die Disziplin in der Datenbank vorhanden ist. Die Anzahl gibt immer an, wie viele Daten in der Datenbank gefunden wurde. Pro Sekunde können ca. 100 Schuss ausgewertet werden. Bei 100 Scheiben a 20 Schuss kann die Berechnung des Ergebnisses ca. 20 Sekunden dauern.

aformdata.php

Stellt Daten bereit, die vom Formular auswertung.html nachgeladen werden.

ergtabelle2.php

Berechnet die HTML oder CSV Ergebnistabellen.

dart.html

Html Seite inklusive Javascript. Benötigt im Backend die Datei dartshoot.php.

dartshoot.php

Liefert den nächsten Schuss nach der angegebenen Uhrzeit und vom angegebenen Stand aus. Die Parameter werden per post Aufruf übergeben. Das Programm fragt die Werte in einer Loop mit timeout (anzahl loops) ab. Wird innerhalb der angegebenen loops kein Schuss (89) abgegeben muss der xhtml Request neu gestellt werden. Daher wird vom Java Script nach 90 Sekunden ein neuer Request abgesetzt. Durch diese Kombination wird verhindert, dass auf dem Server Prozesse ewig laufen, auch wenn der Browser nicht mehr auf die Antwort wartet.

Das Programm liefert einen JSON Output.

Verzeichnis tools

In diesem Verzeichnis sind einige Hilfsprogramme, die zur Laufzeit nicht benötigt werden. Neben den Programmen zum automatisierten testen gibt es auch die Scripte gen-karo10.php und gen-karo15.php die svg Dateien mit zufälligen Zahlen im Karo generieren. Die Daten werden dann auch gleich in die Datenbank geschrieben.

Aufruf: php gen-karo10.php 90 101 > karo10-101.svg

Die erste Zahl ist das Referenz Karo in der Tabelle picval. Die zweite Zahl ist die Bilder ID, unter der die Werte in der Tabelle transval eingetragen werden. Für das 15er Karo ist die Referenz 91.

Datenstruktur

Die Datenbank funshoot enthält 2 Tabellen

picval: enthält je Bild die Punktwerte zu den jeweiligen x und y Koordinaten.

transval: enthält Übersetzungswerte für die automatisch generierten Bilder

Für die automatisch generierten Bilder gibt es immer eine ID in der picval Tabelle. Die IDs für diese Bilder beginnen bei 900.

900: Dart

901: Karo10

902: Karo15

903: Schirm

904: Meyton

Ohne diese Übersetzungstabelle müssten alle Datenpunkte für jedes Bild in der picval Tabelle abgelegt werden. Durch diesen Mechanismus verringert sich die Anzahl der Einträge in der DB von ca. 30.000 pro Bild auf 50-250 Pro Bild

