

Character Sheets

Go back to [Wargrounds Canberra](#)

- Working Form: [Fill in Character Sheet](#)
- Working Location: <https://www.curufea.com/image.php>
- Test (Unchained, Fletching)
 - Form: [Fill in Character Sheet](#)
 - Location: https://www.curufea.com/image_test.php

```
<?php
```

```
// Character Sheet creator for Wargrounds Canberra
// Version 1.3 1/8/24 - curufea@yahoo.com (titles fix)
// Currently hosted at www.curufea.com

// Path to our font file (relative to the location of this file)
$font = 'blackwoodcastle.ttf'; // for field names
$font_data = 'oldeenglish.ttf'; // for player data
$font_wargrounds = 'vinque_rg.otf'; // for the "W"
// Path to images (relative to the location of this file)
$imagepath = 'data/media/wargroundscanberra/';

// defaults for sizes - to be overwritten by form inputs
$tempwidth = round(intval($_GET["width"]));
if (($tempwidth>0) and ($tempwidth<4000)) {
    $width = $tempwidth;
} else {
    $width = 530; // pixel size x of image
};
$height = round($width*1.422); // size y of image
$fontsize = round($width/26.5);
$linespacing = round($fontsize*2);
$borderwidth = round($width/100);
$fontborder = round($borderwidth/2);

// default placeholder photo to be replaced by image from form url
$photo="https://www.worldhistory.org/uploads/images/15277.jpg";
if (isset($_GET["photo"])) $photo=$_GET["photo"];

// the image used for the coin piles
$coins=$imagepath."coinpiles.png";

// the image used for the weapons space filler
$weapons=$imagepath."weapons.png";

// array of default field names
$names = array(
    "Player Name:", "Character's Name:", "Titles/Nicknames:",
    "Race/Species:", "Hair Colour:", "Eye Colour:", "Skin Colour:", "Class &
```

```
Tier:", "School of Magic-", "Faction:", "Warband:", "Marx:", "Arrow
Fletching Colours-");
// the field names used in the form
$getnames = array(
    "name", "character", "title", "species", "hair", "eye", "skin", "class",
    "magic", "faction", "warband", "marx", "fletching");

// 400px x 400px jpeg images stored on the image path (in the Dokuwiki media
area) for faction logos
$factions_images = array("greyscales.png", "empire.png", "horde.png",
    "clans.png");
// used to cross reference the form data to the image name (the Bureaucracy
plugin for the Dokuwiki doesn't send selection number chosen)
$factions_titles = array("Greyscales", "The Empire", "The Horde", "The
Clans", "The Wardens");
// 2987px x 4250px jpeg images stored on the image path (in the Dokuwiki
media area) for faction backgrounds
$factions_backgrounds = array("2background.jpg", "1background.jpg",
    "3background.jpg", "0background.jpg", "4background.jpg");

// count number of field names
$numnames = count($names)-1;

// As it is used multiple times, make the font border thingy a function
function DoFontBorder
($im,$font_size,$start_x,$start_y,$colour,$font,$text,$font_border) {
    for ($x1=($start_x-$font_border);$x1<=($start_x+$font_border);$x1++) {
        for ($y1=($start_y-$font_border);$y1<=($start_y+$font_border);$y1++)
        {
            imagettftext($im, $font_size, 0, $x1, $y1, $colour, $font,
$text);
        }
    }
};

// check if any variables have been sent to this image - if they have,
overwrite the default player data
for ($count=0;$count<=$numnames;$count++) {
    $player_data[$count]= $_GET[$getnames[$count]];
    // Convert the text of the selection for faction to a number
    if ($count==9) {
        $player_data[$count]=
array_search($_GET[$getnames[$count]],$factions_titles);
        $bgimage=$factions_backgrounds [$player_data[$count]];
    }
};

// Background image texture - now themed to faction
$imageback = $imagepath.$bgimage;
```

```
$im2 = imagecreatefromjpeg("$imageback");

// Create image
$image = imagecreatetruecolor($width,$height);

// pick color for the text
$fontcolour = imagecolorallocate($image, 0, 0, 0);

// pick color for text borders
$bordercolour = imagecolorallocate($image, 255,255, 255);

// add background texture
imagecopyresized($image, $im2, 0, 0, 0, 0, $width, $height,imagesx($im2),
imagesy($im2));
imagedestroy($im2);

// x,y coords for imagettftext defines the baseline of the text: the lower-
left corner
// so the x coord can stay as 0 but you have to add the font size to the y
to simulate
// top left boundary so we can write the text within the boundary of the
image
$x = $fontsize;
$y = $fontsize;

$showstuff = true; // by default show all fields

// Add weapons to the character Sheet
$im6 = imageCreateFromPng($weapons);
imagecopyresized($image,$im6, $linespacing*2,
$linespacing*13,0,0,round($width/3.5),round($width/3), imagesx($im6),
imagesy($im6));
imagedestroy($im6);

// Add coin piles to the character Sheet
$im5 = imageCreateFromPng($coins);
if (strlen($player_data[10])>15) { // Move the coin piles down to fit long
warband names
    imagecopyresized($image,$im5, round($width/2),
$linespacing*7,0,0,round($width/2),round($width/8), imagesx($im5),
imagesy($im5));
} else {
    imagecopyresized($image,$im5, round($width/2),
$linespacing*6,0,0,round($width/2),round($width/8), imagesx($im5),
imagesy($im5));
};
imagedestroy($im5);

for ($count=0;$count<=$numnames;$count++) {
    $y=$y+$linespacing; // increment by estimated line separation height
```

```
// Clunky bit to do positioning. I'll just generate an array next time
if ($count==3||$count==7) $y=$y+$linespacing; // blank lines to
separate

if ($count==9) { // right column
    $y= $linespacing*5;
    $x= round($width/2);
};
if ($count==$numnames-1) { // centre the last text in the right column
    $text = $names[$count] . " 999"; // add possible length
    $text_box = imagettfbbox($fontsize,0,$font,$text); // makes an array
of co-ordinates for the text box
    // Get your Text Width and
    $text_width = $text_box[2]-$text_box[0];
    $x = round((($width*2/3)-($text_width/2))); // middle of the right
column minus half the size of the text
    $y=$y+$linespacing;
};

if ($count==$numnames) { // Do the new Fletching field
    $y=$fontsize+($linespacing*15);
    $x = $fontsize;
};

// Player data in olde english font
$text_box = imagettfbbox($fontsize,0,$font,$names[$count]); // workout
where the field name ends
$text_width = $text_box[2]-$text_box[0]+round($fontsize/2);
$text = $player_data[$count];

// Hide the magic and fletching fields if it isn't filled in
if (($count==8) and strlen($text)<1) $showstuff = false;
if (($count==$numnames) and strlen($text)<1) $showstuff = false;

// Do the border for the field names
if ($showstuff) DoFontBorder
($image,$fontsize,$x,$y,$bordercolour,$font,$names[$count],$fontborder);

// Field names in blackwood castle font
if ($showstuff) imagettfttext($image, $fontsize, 0, $x, $y, $fontcolour,
$font, $names[$count]);

if ($showstuff==false) $showstuff=true;

// Change the selected faction to readable text (and add faction logo)
if ($count==9) {
    $text = $factions_titles[intval($player_data[$count])];
    $faction= $imagepath.$factions_images[intval($player_data[$count])];
    $im3 = imageCreateFromPng($faction); // Put in the faction logo
```

```

        $ratio = imagesy($im3)/imagesx($im3); // Try to keep the logos in
the same aspect ratio
        imagecopyresized($image,$im3, round($width*0.84),
round($linespacing*2.3), 0, 0, round($width/8),
round($width/8*$ratio),imagesx($im3), imagesy($im3));
        imagedestroy($im3);
    };

// Print the data

// Put titles, magic type and fletching on the next line so it fits
    if (($count==8) or ($count==$numnames)) {
        DoFontBorder
($image,$fontsize,$x,$y+$linespacing,$bordercolour,$font_data,$text,$fontbord
der);
        imagettftext($image, $fontsize, 0, $x, $y+$linespacing, $fontcolour,
$font_data, $text);
    } else {
        if (($count==2) or ($count==7) or ($count==10)) { // make class and
tier smaller to fit for magic type and warband name
            $maxlen=15;
            if ($count==2) $maxlen=25;
            if (strlen($text)>$maxlen) { // split long lines in two
                $string_break=stripos($text," ",$maxlen-3); // look for the
first space after the $maxlen-3 character
                if (strlen($text)>$maxlen*2)
$text=substr($text,0,stripos($text,"",$maxlen*2)); // truncate ludicrously
long strings
                DoFontBorder
($image,round($fontsize*3/4),$x+$text_width,$y,$bordercolour,$font_data,subs
tr($text,0,$string_break),$fontborder);
                DoFontBorder ($image,round($fontsize*3/4),$x+$text_width-
($text_width*($count==2)),$y+($linespacing/1.5),$bordercolour,$font_data,sub
str($text,$string_break-strlen($text)),$fontborder);
                imagettftext($image, round($fontsize*3/4), 0,
$x+$text_width, $y, $fontcolour, $font_data,substr($text,0,$string_break));
                imagettftext($image, round($fontsize*3/4), 0,
$x+$text_width-($text_width*($count==2)), $y+($linespacing/1.5),
$fontcolour, $font_data,substr($text,$string_break-strlen($text)));
            } else {
                DoFontBorder
($image,round($fontsize*3/4),$x+$text_width,$y,$bordercolour,$font_data,$tex
t,$fontborder);
                imagettftext($image, round($fontsize*3/4), 0,
$x+$text_width, $y, $fontcolour, $font_data, $text);
            };
        } else {
            DoFontBorder
($image,$fontsize,$x+$text_width,$y,$bordercolour,$font_data,$text,$fontbord
er);
            imagettftext($image, $fontsize, 0, $x+$text_width, $y,

```

```
$fontcolour, $font_data, $text);
    };
};
};

// Add the photo to the character Sheet
$im4 = imagecreatefromjpeg($photo);
$ratio = imagesy($im4)/imagesx($im4); // Try to keep the photo in the same
aspect ratio
// Draw border
imagefilledrectangle($image, round($width/1.8), round($width/1.5),
round($width*31/32), round($width*6/16/$ratio+$width), $fontcolour);

// Resize and place in character Sheet
imagecopyresized($image, $im4, round($width/1.8)+$borderwidth,
round($width/1.5)+$borderwidth, 0,0, round($width*31/32)-round($width/1.8)-
$borderwidth-$borderwidth, round($width*6/16/$ratio+$width)-
round($width/1.5)-$borderwidth-$borderwidth, imagesx($im4), imagesy($im4));
imagedestroy($im4);

// Add the large W
imaggottext($image, $fontsize*3, 0, round($width*0.8), round($width/6),
$fontcolour, $font_wargrounds, "W");

// tell the browser that the content is an image
header('Content-type: image/jpeg');
// output image to the browser
imagejpeg($image);

// delete the image resource
imagedestroy($image);
?>
```

To do

- Possibly API use for wherever photos get stored (or just upload them) - may need to edit photos to fit

From:
<https://curufea.dreamhosters.com/> - Curufea's Homepage

Permanent link:
https://curufea.dreamhosters.com/doku.php?id=wargroundscanberra:character_sheet&rev=1722493866

Last update: 2024/07/31 23:31

