

This document contains instructions on how to create customized cards for Decipher's Lord of the Rings Trading Card Game using [nanDECK](#).

If there are any questions or comments regarding the use of this utility or the content of this document, please reach out to [TelTura on the TLHH forums](#).

TABLE OF CONTENTS

[Packaged Installation](#)

[Manual Installation](#)

[Post-installation Setup](#)

[Card Generation](#)

[Batch File](#)

[Run nanDECK](#)

[How Do I...?](#)

[...get custom portraits to work?](#)

[...decide what size to make my portraits?](#)

[...make twilight and culture symbols?](#)

[...insert newlines or use other formatting?](#)

[...use names with accents?](#)

[...make a full art card?](#)

[...include a golden artifact overlay?](#)

[...change the csv to something besides cards.csv?](#)

[...add my own templates?](#)

[Troubleshooting](#)

[Accented names aren't working](#)

[Random words are bolded that I don't want bolded](#)

[For some reason a random accent is being added to words](#)

[My card's portrait is all black \(but the rest is fine\)](#)

[My card is all black \(except maybe the portrait and numbers\)](#)

[My card is all white with a little message](#)

[My card has a bunch of scroll bars where text should be](#)

[CSV Documentation](#)

[Customizing Generation](#)

[settings.txt](#)

[game_text_replacements.txt](#)

[all_text_replacements.txt](#)

[Making Custom Resources](#)

[Custom Templates](#)

[Custom Culture Icons](#)



First, choose whether to use Packaged Installation or Manual Installation. First-time installers should use the Packaged Installation. If any components do not work after performing the Packaged Installation, follow the instructions under Manual Installation for the affected components.

Make sure to follow the instructions under [Post-installation Setup](#) afterwards regardless of which method you use.

Packaged Installation

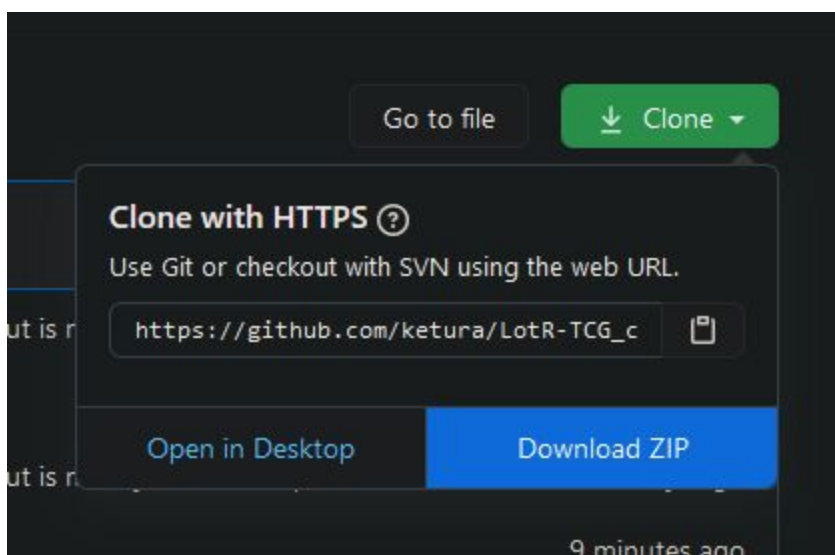
Go and download the *.zip for the latest version from the github page for this project: https://github.com/ketura/LotR-TCG_card_generator/releases . This package includes packaged templates, example images, and nanDECK all together.

Unzip this file to the location of your choice. This includes the Original- and Double-sized templates but does not include the Huge templates (for size reasons). If you want to include Huge templates, follow step 4 in Manual Installation below. For details about template sizes, please see [\[card_size\]](#).

After this, follow the instructions in the [Post-installation Setup](#) section, below.

Manual Installation

1. Download the card generator script from github here: https://github.com/ketura/LotR-TCG_card_generator



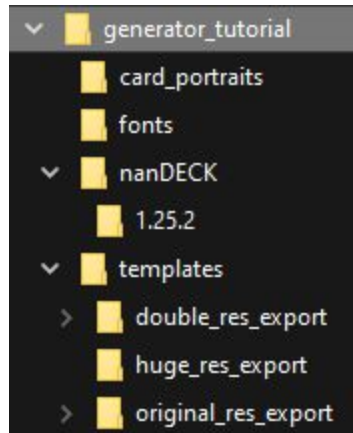
This can either be cloned or you can download a zip file by clicking the Clone button and then clicking Download ZIP. Extract the card generator files to a folder of your choice. All following steps will be copied to the same location.

2. Download nanDECK. Version must be 1.25.2 or higher. The latest version can be obtained at nanDECK's website: <http://www.nand.it/nandeck/>
3. "Install" nanDECK. This is done by opening the zip file and extracting the files into the nanDECK folder where you installed the generator.
4. Download the fonts and template files for the resolutions you wish to generate (any or all of Huge, Double, or Original) from this reddit post here:
https://www.reddit.com/r/LOTRTCG/comments/gsdvsa/lord_of_the_rings_tcg_card_templates_and_resources/
5. Extract the template files into the generator install location. Keep the folder names for each size; they should be called
 - a. "huge_res_export"
 - b. "double_res_export"
 - and
 - c. "original_res_export".

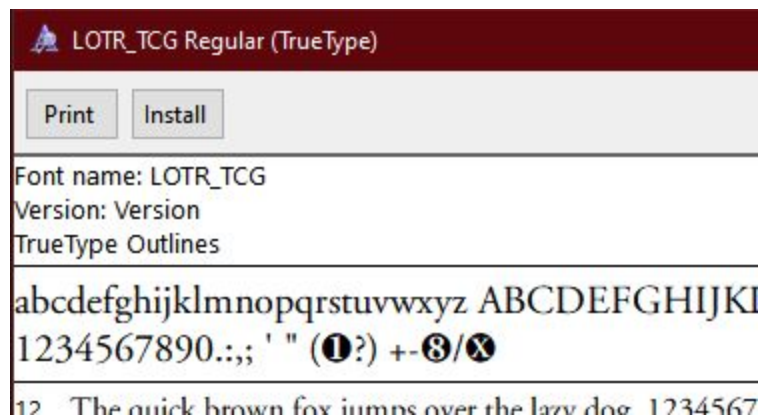
Remember, you do not need all three if you do not want cards of each size.

Post-installation Setup

You should now have a folder structure that looks something like this, with the files from the above steps in the appropriate places. If it doesn't appear similar to the picture below, refer to the Manual Installation steps for the affected component.



Extract the fonts and install all of them onto your machine. They are in the fonts folder of the Packaged Installation, or in the zip file you downloaded from the Manual Installation. On Windows this can be done by double-clicking each extracted *.ttf file and clicking the “Install” button in the upper-left hand corner.



There are 7 fonts to install:

1. LOTR_TCG-Regular
2. LOTR_TCG-Bold
3. LOTR_TCG-Italic
4. LOTR_TCG-BoldItalic

5. LOTR_TCG-Lore
6. LOTR_TCG-Title
7. LOTR_TCG-Icons-Bold

Card Generation

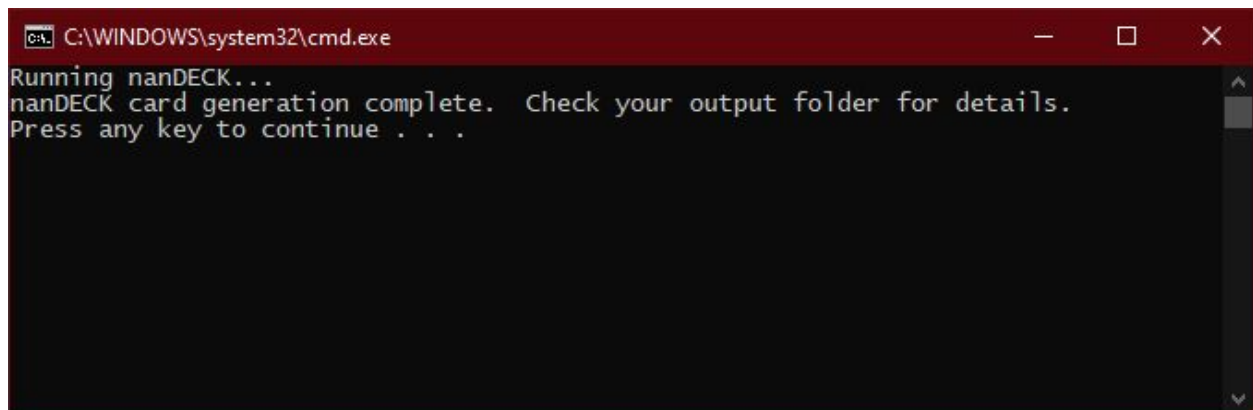
There are two ways to use this nanDECK script:

1. A batch file which opens nanDECK, builds the entire deck, and exits.
2. Open nanDECK itself, which can generate individual cards and use other features of the built-in editor.

Batch File

Double-click `lotr-tcg_card_generation.bat`.

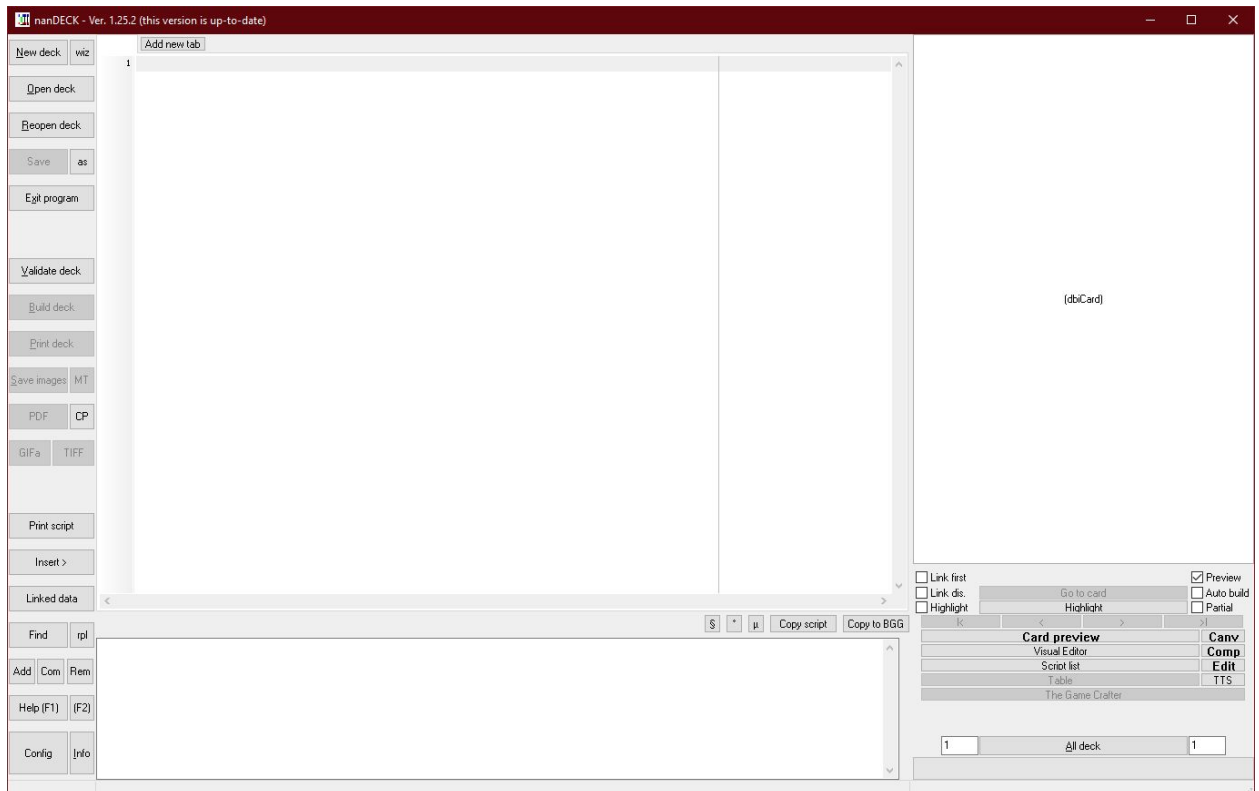
A black command prompt window will open as nanDECK runs in the background. Once the deck has been generated, a message will indicate when it is safe to close the window:

A screenshot of a Windows command prompt window. The title bar is dark red and contains the text "C:\WINDOWS\system32\cmd.exe" along with standard window control buttons (minimize, maximize, close). The command prompt area is black with white text. The text displayed is: "Running nanDECK...", "nanDECK card generation complete. Check your output folder for details.", and "Press any key to continue . . .".

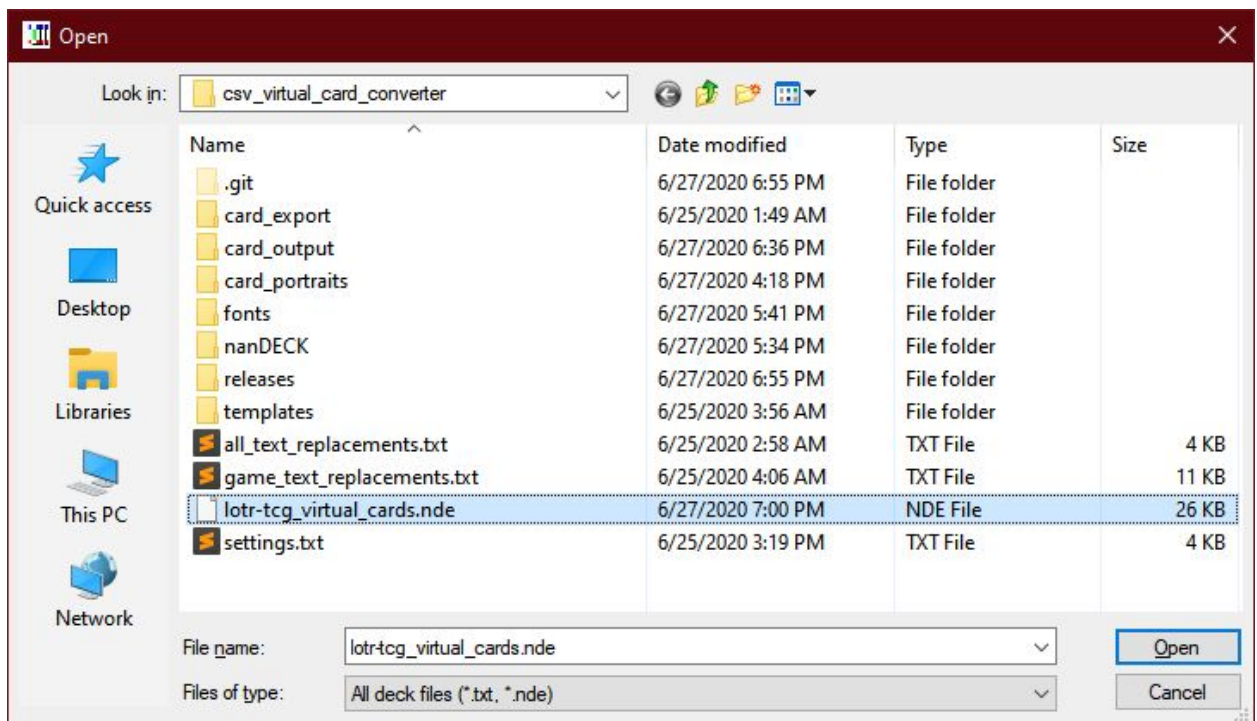
```
C:\WINDOWS\system32\cmd.exe
Running nanDECK...
nanDECK card generation complete. Check your output folder for details.
Press any key to continue . . .
```

Run nanDECK

1. Run nanDECK.exe. The major operations are on the left, while the card preview is on the right. In the center is the work area that contains the script (you can usually ignore it).



2. In the upper-left, click the "Open deck" button, and navigate to your copy of `lotr-tcg_virtual_cards.nde`:



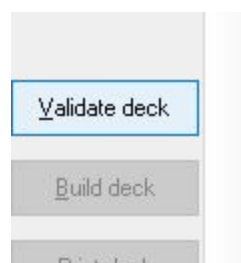
You should now see the big ol' script loaded in the center pane.

nanDECK calls any collection of cards a “deck”. This script has been preconfigured to open the cards.csv file as a deck. Whenever you want nanDECK to construct this deck, there are two steps that need to be performed: validation and building.

Validation checks that the script is valid (which it should be) and that the data in the CSV is valid (which is up to you).

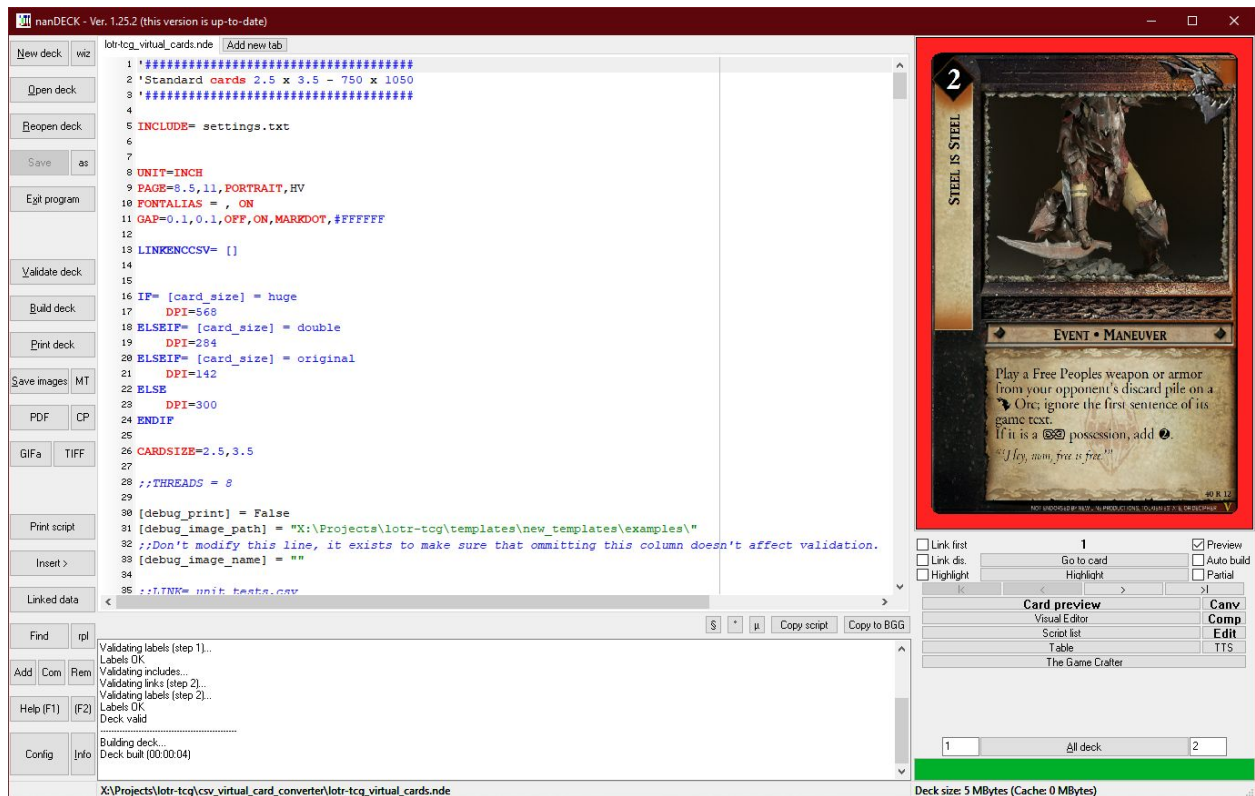
Building then takes a previously validated script and constructs images for each card.

If you click the “Validate Deck” button, it will validate the script and deck and then enable the “Build Deck” button. However, you can *also* right-click the “Validate Deck” button and it will perform a validation followed immediately by a build.



Right-click this button:

You should see the preview panel flash as it creates each card, before returning to the first card once it's done:

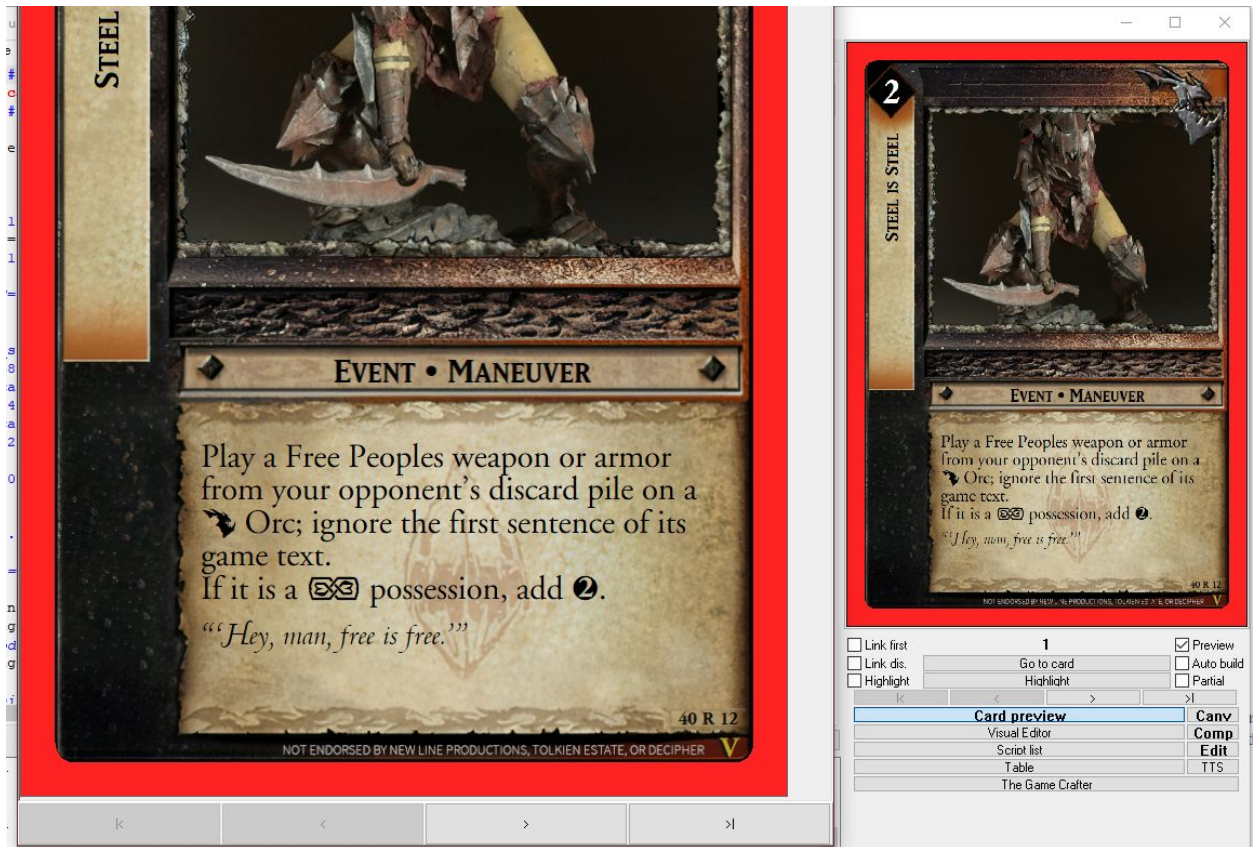


You can use the arrows beneath the card preview to navigate to the next/previous card, or to the beginning or end of the deck:



You can right-click the card preview image to regenerate just the current card, which is handy when testing formatting or wording for one card without waiting on the rest of the entire deck to be re-generated.

Don't be alarmed if the text looks scrunched or pixelated; the preview itself is squashed a bit. The actual exported card will have no such spottiness. To see a pixel-perfect full-sized version of the preview in a new window, click the "Card preview" button:



With the Card Preview button open, you can hover over the new window and use your mouse scroll wheel to quickly navigate up and down the card range.

That's it! Once you've made it this far, all that's left to do is for you to start entering your cards. Get designing!

How Do I....?

...get custom portraits to work?

All you need to do is put the image in the correct folder (“card_portraits” by default, but see [\[card_image_path\]](#) for how to customize this location). Once you have done this, then in your cards.csv, put the name of the image in the [image_name](#) column. Remember to include the extension, so use “example.png”, NOT “example”.

...decide what size to make my portraits?

Card portraits will be automatically scaled to fit the given template size (while preserving aspect ratio). Nevertheless, if you want the best results you will want your portraits to match the appropriate template size as much as possible:




Character		Modifier	
Huge	1116 x 836	Huge	991 x 736
Double	558 x 419	Double	495 x 369
Original	280 x 210	Original	250 x 185
Site		One Ring / Full Art	
Huge	913 x 1579	Huge	1292 x 1860
Double	457 x 791	Double	646 x 930
Original	229 x 395	Original	323 x 465

As you can see, the Character and Modifier sizes share the same approximate aspect ratio but are slightly different sizes, which is just how the original cards were designed (making extra room on the left for the title did a number on the space available).

This size includes a very narrow overlap of 1-3 pixels underneath the card template, which can vary further if the portrait border on the template is wobbly (such as with Moria or Gollum).

...make twilight and culture symbols?

You can use either parenthesis or square bracket tags to produce twilight symbols, culture symbols, and block symbols. If you're a TLHH veteran, it's compatible with the format on that forum.

[1] or (1) = ❶	[Dunland] or (Dunland) = 
[2] or (2) = ❷	[dwarven] or (dwarven) = 
...	
[9] or (9) = ❹	[Elven] or (ELVEN) = 
[X] or (X) = ❸	etc

As you can see, it is not case sensitive.

For block symbols, you can either write 3T or 3K to get 3 with a tower/king symbol, or you can write (T) or (K) to get the tower or king symbols on their own.

Twilight and culture symbols are only replaced in the game text (not lore or anything else). The block symbols however are replaced in everything, including the card type bar.

...insert newlines or use other formatting?

nanDECK uses html formatting, so use
 tags to insert a newline. Common formatting such as for bold, <i>for italic</i>, or <u>for underline</u> all work, as well as others.

See [game_text](#) for more information about formatting the game text area.

...use names with accents?

First of all, there is a list of 60 names that automatically have their accents added, so if you're trying to get "Théoden", "Éowyn", or one of the Nazgûl it's already got you covered; just use the unaccented versions like "Theoden", "Eowyn" or "Ulaire X" to let the generator automatically

take care of it for you.

Otherwise, simply manually include the accented characters you want to use in the CSV as normal. There is a handy table under [all_text_replacements.txt](#) with common accented vowels.

...make a full art card?

Use the full_art tag in the [tags](#) column and provide an image that's the correct size. If you're making a One Ring card (you use "OneRing" in the template field), those are full art by default. See the [full_art](#) tag for more details.

...include a golden artifact overlay?

Make the card an artifact (by putting "Artifact" in the card_type column) or include the [artifact_overlay](#) tag. artifact_overlay is ignored on character, site, or onering layouts.

...change the csv to something besides cards.csv?

Unfortunately due to a bug within nanDECK this is not feasible to do within settings.txt at this time.

If you're willing to modify the script itself, open **lotr-tcg_virtual_cards.nde** and change line 36 to point to the csv you want to use:

```
LINK= "C:\users\me\documents\my_custom_file.csv"
```

Do so at your own risk, tho.

...add my own templates?

See [Making Custom Resources](#).

Troubleshooting

Accented names aren't working

If you're entering "Úlairë Cantëa" manually and see it being rendered as something like "Â?lairÂ« CantÂ«a", then there are a number of things that might be wrong. Try each of these in this order:

1. Verify that you followed the steps in [Post-installation Setup](#) regarding installing all 7 fonts. On Windows there should be no side-effects to reinstalling fonts again, so double-check that you got all 7 installed properly. Once you have done so, restart nanDECK.
2. Within your spreadsheet editor of choice (Calc, Excel, etc) make sure that when you save the CSV it uses a code page / encoding that supports accents, such as "Western Europe / Windows-1252 / Latin" or equivalent. Using basic ASCII may or may not work.
3. If using a Latin code page didn't work, try saving it as UTF-8.
4. If all else fails, but the generator's auto-replace works, then take a look at modifying [all_text_replacements.txt](#) to include a custom auto-replace rule for the name you're having trouble with.

Random words are bolded that I don't want bolded

Not all of the auto-format rules work in all cases, sadly. If you find yourself rubbing up against one that you'd just rather turn off, go into **game_text_replacements.txt** and remove the line that mentions the keyword you're having a hard time with. Just keep in mind you'll have to manually use tags around the associated keyword in all cases from then on.

Please reach out to [TelTura on TLHH](#) if you find a keyword that is consistently a problem and I'll consider removing it permanently.

See [game_text_replacements.txt](#) for more details.

For some reason a random accent is being added to words

For example, while developing this tool there was a rule to turn the dwarf name Uri to Úri, which unfortunately meant that any card that had the word “during” would show up as “dÚring”. The only way to solve this is to remove the offending line from **all_text_replacements.txt**; search in that file for the text that is getting replaced and remove that line.

Please reach out to [TelTura on TLHH](#) if you find a name that is consistently a problem and I'll consider removing it permanently.

See [all_text_replacements.txt](#) for more details.

My card's portrait is all black (but the rest is fine)

nanDECK was unable to find the portrait image and thus the black background is showing instead.

1. Make sure that you didn't make a typo in the image's filename
2. Make sure that you didn't make a typo in the image_name column of your CSV
3. Make sure you included the extension in the CSV (“example_name.png”, not “example_name”)
4. Make sure you are putting images in the correct spot. The default location is the “card_portraits” folder in your installation directory, but see [\[card_image_path\]](#) for how to see where you might have altered **settings.txt** to use a different directory

My card is all black (except maybe the portrait and numbers)

nanDECK was unable to locate the correct template, so what you're seeing is the card background showing through.

1. Make sure you didn't make a typo in the culture column of your CSV. See [culture](#) for a list of valid values.

2. Check the [\[card_size\]](#) entry in **settings.txt** to see what size you are currently using (defaults to “double”). Make sure that the associated folder has template images in it. For example, if your card size is “double”, then the templates/double_res_export folder should have a bunch of *.png files and two folders (“components” and “overlays”). If you do not see those three things, then see [Manual Installation](#) for retrieving the correct files.
 - a. If you see the same folder again, i.e. you have a templates/double_res_export/double_res_export folder, then you extracted your files too deep. Cut-and-paste the files up one level.

My card is all white with a little message

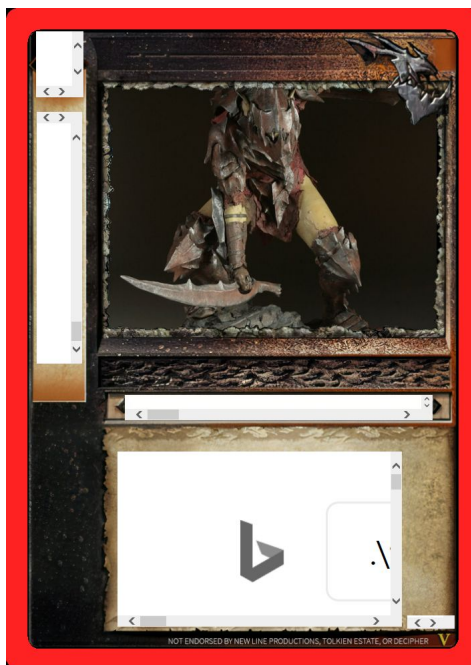
You made a typo in your template column. The only valid values are (not case sensitive):

- character
- modifier
- site
- onering

See [template](#) for more information.

My card has a bunch of scroll bars where text should be

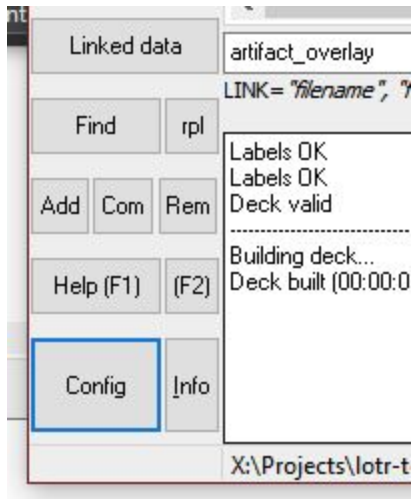
This so far has only been reported when using the batch file. If your card looks like this:



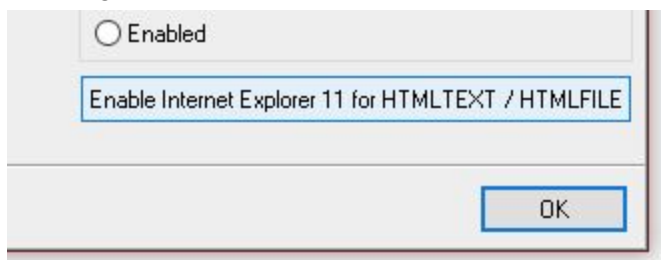
The issue is that nanDECK, which is configured in this script to use Internet Explorer's engine to render HTML, is unable to interface with IE properly. By default it can only find up to IE 6, and later versions have to be explicitly enabled.

Follow these steps:

1. Open nanDECK and click on the "Config" button in the lower-left-hand corner:



2. Click on the "Enable Internet Explorer 11 for HTMLTEXT / HTMLFILE" button in the lower right-hand corner:



3. Click OK, and then restart nanDECK.

If these steps do not fix the problem, try using the method described under [Run nanDECK](#) instead, as the problem only seems to happen when using the batch file.

CSV Documentation

When nanDECK runs the generation script, it opens cards.csv and iterates through each row, generating one card per row. All of the information needed to create a card is contained within that CSV, with the exception of the card portrait. Thus, a row in the spreadsheet and a card portrait placed in the correct folder is all that is needed to create a fully-formed card for the LotR-TCG.

It is recommended that you keep your master spreadsheet in a fully-featured spreadsheet program such as LibreOffice Calc or Microsoft Excel, and only export to CSV when ready to export to nanDECK. However due to the simple nature of the CSV format, feel free to work with it in whatever manner appeals to you the most.

Below are the details of each column within the CSV.

id

This column is for you to use to keep a single card's identity consistent even if the title/subtitle changes over time. This column also dictates what the filename of the card will eventually be called when it is generated in nanDECK, so use whatever naming or numbering scheme makes sense to you. This can contain any characters so long as they are valid in a filename (so no '/' etc).

image_name

This is the filename nanDECK will attempt to find for the card portrait. It needs to include the extension, so this should look like "example.png" or "my_card.jpg", depending on the file you are providing. Do not omit the extension.

The default card portrait image directory is a folder called "card_portraits" in the same directory as the script. This location can be configured by altering the [\[card_image_path\]](#) entry in settings.txt (see [Customizing Generation](#)).

Sub-folders can be used within `card_portraits`, so long as the folder is included in `image_name`. For example, “set1/orcs/mycard.png”, so long as you have those folders in your `card_portraits` directory.

unique

T (for “true”) if this card should be unique, i.e. marked with a • symbol in front of its name. Any other value (even blank) will be interpreted as non-unique.

title

The name to use for the card’s title. This will be modified by the automatic search/replace rules as detailed in **`all_text_replacements.txt`**.

This will not be automatically sized-to-fit. If the title is too long to fit the card’s title space, you will probably want to edit the [title_font_size](#) column.

subtitle

The name to use for the card’s subtitle. This will also be modified by the automatic search/replace rules as detailed in **`all_text_replacements.txt`**.

This will not be automatically sized-to-fit. If the subtitle is too long to fit the card’s title space, you will probably want to edit the [subtitle_font_size](#) column.

culture

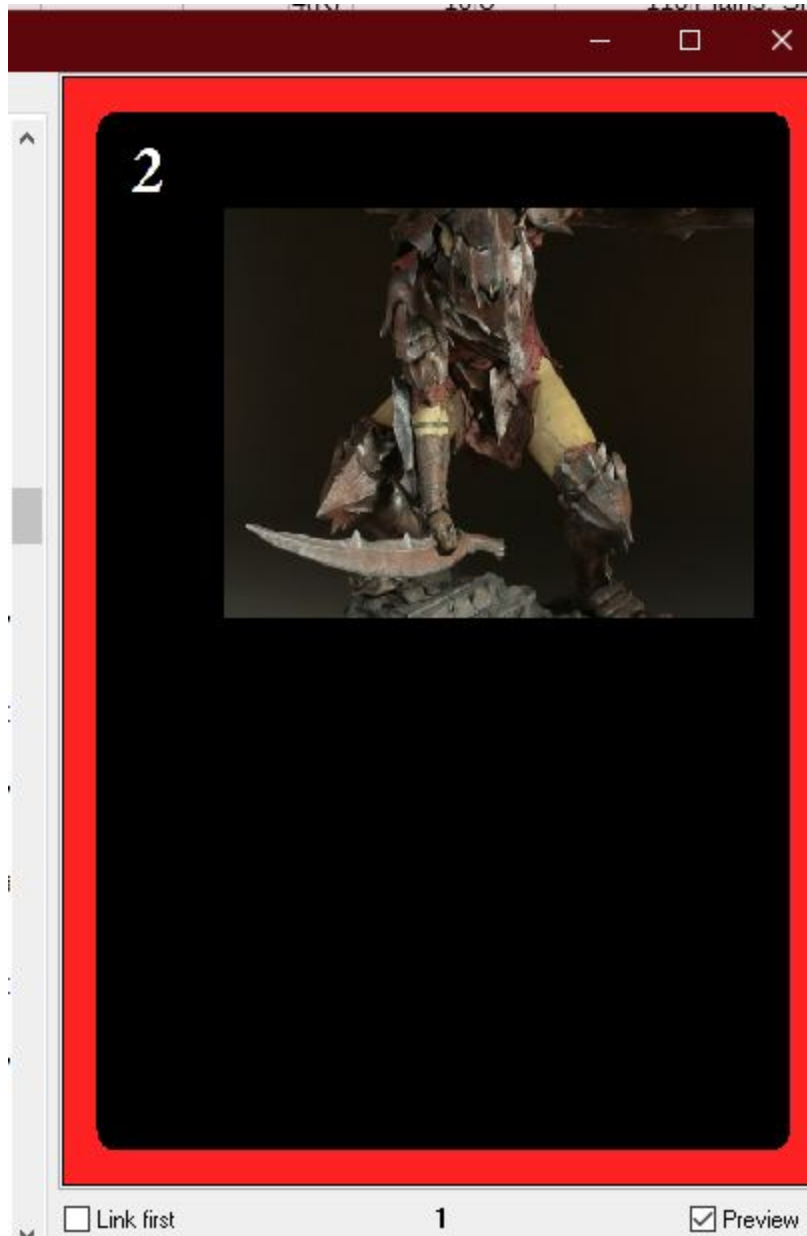
This is the name of the culture to use for the card. nanDECK will use this to figure out which template file to use, and as a result making new cultures is simply a matter of putting the appropriate template files in the templates folder for nanDECK to find (see [Making Custom Material](#)).

Sites, One Ring, and `full_art` cards do not need to specify a culture; this column will be ignored in those cases.

For all other card types, this is the list of values that are supported with the current version of the template files (case insensitive):

- Dunland
- Dwarven
- Elven
- Gandalf
- Gollum_freeps
- Gollum_shadow
- Gondor
- Isengard
- Men
- Moria
- Moria_balrog
- Orc
- Raider
- Ringwraith
- Rohan
- Sauron
- Shire
- Uruk-hai

If any culture is provided that causes nanDECK to fail to find an appropriate template (except for sites, one ring, or full_art cards), you will end up with what looks like a mostly-black card:



template

This column controls which layout nanDECK will attempt to construct. It is limited to one of the following values:

- **Character:** a companion, minion, ally, or follower, using the layout that has the title/subtitle at the top with a larger card portrait.
- **Modifier:** a possession, artifact, event, or condition, using the layout that has the title/subtitle along the left side with a smaller card portrait.
- **Site:** a site which uses the landscape layout
- **OneRing:** a slightly altered Modifier layout which puts the subtitle where the card type bar normally goes. Also has no default art, and is drawn like a full art card.

If any other template type is provided, the card will be pure white with a message indicating that the template was unrecognized:



(the message reads "unrecognized template type: asdf")

card_type

The type of card, meaning (probably) one of the following values:

- Companion
- Minion
- Ally
- Follower
- Artifact
- Possession
- Condition
- Event
- Sanctuary

Standards sites and One Ring cards can safely leave this column blank (or provide a value, it doesn't matter).

Some card types have logic associated with them. Ally will put the home site in the type bar, Follower will place overlay icons to create a yellow twilight cost, Artifact will draw the artifact overlay, and Sanctuary will use grey versions of the normal site icons where appropriate. Minions will never draw resistance.

In most cases, however, the most apparent use of this value is that it will be printed on the card itself, such as how “Event” goes on the card type bar in the middle of the card:



This means that for the most part, so long as one of the four standard template layouts are fine for your purposes, you can provide any value here as a custom card type.

card_subtype

“Subtype” here refers to Race for character cards, Item Class for possessions, Phase for events, and Target Area for conditions. Since a card can’t have more than one of these classifications, they are all summarized under the one column.

Subtype has no logic associated with it, except for checking that there is a value (see, for instance, how [Smeagol has no Race](#) and thus his type bar shows no extra dot).

This means that this value is really for choosing what to display as the second value in the type bar (third value for Allies):



Above you can see a subtype of Maneuver printed on the card type bar. Leave this value blank to omit the second section of the type bar (third for allies).

twilight

The twilight cost, probably only 1-2 digits. This is not enforced, however, so any text can be provided. If no twilight value is provided, nothing will be drawn. For sites, if this value is not provided, the twilight icon will also be omitted (which all site 1s do).

See also the [twilight_color](#) column, which can override the default white color.

strength

If this value is provided, a sword/shield icon will be drawn in the top icon slot and the value of this text drawn on top. This should probably only be a 1-2 digit number, or a 1 digit number preceded by a + or -. If no value is provided, the icon will be omitted, so be sure to use 0 if you want to keep it.

If you want to keep the drawing of the number but suppress the icon, add the `disable_strength` tag (see the [tags](#) column).

vitality

If this value is provided, a red orb icon will be drawn in the middle icon slot and the value of this text drawn on top. This should probably only be a 1-2 digit number, or a 1 digit number preceded by a + or -. If no value is provided, the icon will be omitted, so be sure to use 0 if you want to keep it.

If you want to keep the drawing of the number but suppress the icon, add the `disable_vitality` tag (see the [tags](#) column).

resistance

If this value is provided, a black orb icon will be drawn in the bottom icon slot and the value of this text drawn on top. This should probably only be a 1-2 digit number, or a 1 digit number preceded by a + or -. If no value is provided, the icon will be omitted, so be sure to use 0 if you want to keep it.

If you want to keep the drawing of the number but suppress the icon, add the `disable_resistance` tag (see the [tags](#) column).

If you want to draw the ring-bearer ring icon around this icon, add the `ring_bearer` tag (see the [tags](#) column).

signet

If this value is provided, a wax signet icon with this name will be searched for and drawn in the bottom icon slot.

Additional signets can be supported by providing the appropriately-named icon in the templates folder.

If you want to provide this value, yet suppress the drawing of the icon (why?), add the `disable_signet` tag (see the [tags](#) column).

site

This is used as home site for Allies or Minions, site modifier for conditions or possessions, and site number for sites themselves.

For Allies, the value of this column will be drawn in the card type bar as “Home X”, where X is the value of this column.

For minions or other non-site cards, if this value is provided, a compass icon will be drawn in the bottom icon slot and the value of this text drawn on top. This should probably only be a 1-2 digit number, or a 1 digit number preceded by a + or -. If no value is provided, the icon will be omitted, so be sure to use a 1 if you want to keep it.

If you want to keep the drawing of the number but suppress the icon, add the `disable_site` tag (see the [tags](#) column).

For sites, this text will be drawn in the lower-left hand corner (upper-left once you’ve rotated the card on its side as normal). By default, the Fellowship block icon will be drawn under the text here (a bright compass). If “T” is somewhere in this column, it will draw the Tower site icon. If “K” is somewhere in this column, it will draw the King site icon. If this column is empty, the site will instead draw the Shadows icon (a dark compass).

set_num

Set number of the card, used to create the collector’s info in the lower-right-hand corner.

rarity

Rarity of the card, used to create the collector’s info in the lower-right-hand corner. This value is not checked, so anything can be used.

card_num

Card number of the card, used to create the collector’s info in the lower-right-hand corner.

game_text

The active game text written in the bottom half of the card's face (not including lore). This will be modified by the automatic search/replace rules as detailed in **all_text_replacements.txt**, and will also be modified by **game_text_replacements.txt** rules.

This text is HTML formatted, meaning you can make bold, italicized, or other formatted text using standard tags. `Archer`, for example, will be rendered as **Archer** on the card. Note however that there are several automatic formatting changes that are made for you; see [Customizing Generation](#) for more details on that.

One of the automatic replacements made is to convert culture, twilight, and block tags into the appropriate symbol. [Moria] will convert to the Moria culture symbol, (1) will convert to the "1 twilight" symbol, and 3T will replace the T with the Tower block symbol. Again, see [Customizing Generation](#) for more details.

A `
` tag can also be used to manually insert a newline.

The color of this text can be modified on a per-card basis by altering the `game_text_color` column, below. The spacing between lines can be controlled by altering the `game_text_spacing` column, below. See also [game_text_width](#) to control how wide the game text section is.

lore

The lore text printed just below the game text in a more artistic, italicized font. This will be modified by the automatic search/replace rules as detailed in **all_text_replacements.txt**, and will also be modified by **game_text_replacements.txt** rules.

This text is HTML formatted, meaning you can make bold, italicized, or other formatted text using standard tags. Regular " and ' quotes will be replaced with their smart quote equivalents, see Customizing Generation below for more details on that.

A `
` tag can also be used to manually insert a newline.

The color of this text can be modified by altering the `game_text_color` column, below. The spacing between lines can be controlled by altering the `lore_spacing` column, below. See also [game_text_width](#) to control how wide the game text section is.

twilight_color

When this is a properly formatted hexadecimal HTML color code (#RRGGBB), this will alter the color used to draw the twilight cost text. If not provided, the text will use the internal default, which is white (#FFFFFF).

This is only really pertinent for shadow Followers, which draw the number in yellow. Other colors may of course be used for custom applications as needed.

title_font_size

This value will modify the font size used for the title only. If no value is provided, the internal default values will be used. These defaults are 9.7 for characters, 9 for modifiers, 9 for sites, and 11 for the One Ring. Any value may be provided to make a larger or smaller text, but note that if it's too large it's likely to simply run off the page.

subtitle_font_size

This value will modify the font size used for the subtitle only. If no value is provided, the internal default values will be used. These defaults are 7.8 for characters and 9 for modifiers (sites do not have a subtitle, and the One Ring uses the card type bar for its subtitle). Any value may be provided to make a larger or smaller text, but note that if it's too large it's likely to simply run off the page.

game_text_color

When this is a properly formatted hexadecimal HTML color code (#RRGGBB), this will alter the color used to draw the text font for the game text, lore, collector's info, title, subtitle, and card type bar. If not provided, the text will use the internal default, which is black (#000000).

This is mostly used for inverting the color to white when making full art cards. Other colors may of course be used for custom applications as needed.

icon_text_color

When this is a properly formatted hexadecimal HTML color code (#RRGGBB), this will alter the color used to draw the text font for the strength, vitality, resistance, and site number stats. If not provided, the text will use the internal default, which is white (#FFFFFF).

This is mostly used for debugging. Other colors may of course be used for custom applications as needed.

game_text_spacing

This value controls the spacing between lines of text in the game text area (but not the title/subtitle, which are fixed). If not provided this will default to 100, but can be increased or decreased to make game text lines more spaced or more cramped.

lore_spacing




This value controls the spacing between lines of lore text. If not provided this will default to 98, but can be increased or decreased to make lore lines more spaced or more cramped.

game_text_width

This value controls how wide the game text area is, represented as a percentage of the width of the card (i.e. “100%” means “as wide as the card is”). The left margin is fixed, but official cards will run more or less to the right to let the text flow differently.

Since both game text and lore are combined into a single set of text, this will affect both of them automatically at the same time.

If omitted, this will revert to the default width of 69%. If the [debug_regions] setting is set to true in **settings.txt** (see Customizing Generation, below), then the card will draw a pink rectangle helping you to visualize exactly what is going on with regards to the game text area.

<p>Default value (69%):</p>	
<p>72.5%</p>	
<p>60%</p>	

border_color

When this is a properly formatted hexadecimal HTML color code (#RRGGBB), this will alter the color used to draw the border that goes around the card. If not provided, the text will use the internal default, which is black (#000000).

This was unused by official cards, except that white was used for playtest runs (to save on ink, presumably). Other colors may of course be used for custom applications as needed.

tags

This column is the grab-bag where miscellaneous, infrequently-used card values are placed to handle various non-standard modifications to cards, where it was deemed that a whole column would be a waste.

Tags may be delimited however you like: comma-separated, space separated, pipe separated, whatever.

Following is a list of tags and what behavior they invoke.

full_art

Normally, after the card portrait has been drawn, the appropriate card template for that culture will be applied on top, filling in most of the remaining card space. However, the One Ring templates do not have a distinction between portrait and template; a single image is drawn underneath all the text. This is called a “full art” card, and when the `full_art` tag is used, any template layout will be altered to be compatible with it.

When the `full_art` tag is used, there are a number of alterations to the standard card generation:

- The card portrait (referenced in the `image_name` column) will be positioned and scaled (if necessary, preserving the aspect ratio) to cover the entire card background
- The twilight icon (a grey orb for Free Peoples, a black diamond for Shadow) will be drawn underneath the twilight cost

- The culture icon (if applicable) will be drawn in the upper-right-hand corner
- The artifact overlay will be disabled (unless the `artifact_overlay` tag is used to override this)
- The disclaimer at the bottom of the card (normally part of the card template) will be drawn in black text

When making a full art card, there are several other tags and features that you should consider using:

- Consider changing the `game_text_color` to white if your background demands it
- Consider using the `disclaimer_light` tag to also draw the disclaimer using white text instead of black
- Some full art cards prefer to include their own versions of various icons, such as The One Ring, The Ruling Ring, which has fire swirling around the strength icon. If you want a similar effect, copy-paste the appropriate icon into your card art, and then use the appropriate “`disable_X`” tags below to prevent card generation from covering up your custom artwork.
- If you want to make a `full_art` artifact, note that the artifact overlay includes both culture icon *and* golden card type bar. If you do not wish to include the golden card type bar, you will need to manually include the golden culture icon in your card art. If however you are fine with both culture + bar, then use the `artifact_overlay` tag to force their inclusion.

`disclaimer_light, disclaimer_dark`

When these tags are used, the disclaimer is drawn at the bottom of the card. Normally this is baked into the card templates and is not needed, but if you are providing a custom template without one or if you want to force generation to use white or black text as appropriate for your card art, these tags may be used.

`disable_strength,disable_vitality,disable_resistance,disable_site`

Normally any non-site, non-ally cards that include a stat for strength, vitality, resistance, or site number will result in the appropriate icon being drawn and then the stat drawn on

top of that icon. However, in some circumstances it makes sense to disable this functionality, such as the icon being drawn in an alternate style in the card template or card art when the `full_art` tag is used.

Using these tags will suppress the drawing of the associated icon.

`disable_site` will only affect non-site cards.

disable_signet

Unlike stat-based icons, the signet icon is normally all there is to communicate about the slot, and so disabling it does not make sense. However, there are a pair of cards that were reprinted in later sets ([Boromir, Defender of Minas Tirith](#); and [Pippin, Hobbit of Some Intelligence](#)) and ruled to still “have” the signet, in spite of *also* having a resistance stat and thus not printing the signet on the card itself.

Thus the `disable_signet` tag exists to facilitate automated card exporters, which may need the tag applied to those two cards to avoid compromising the integrity of card data.

disable_culture, disable_twilight

With `full_art` cards, the twilight icon and culture icon are drawn at the top of the card, where normally they are part of the template. Use these tags to disable that behavior if, for instance, your full card art includes those icons baked-in.

ring_bearer

Ring-bearers are indicated by a golden ring icon drawn around the resistance icon. If you wish to have this icon drawn, include the `ring_bearer` tag. If you do not provide a resistance stat, or the card is a Minion, this will have no effect. This tag also only affects the character layout and will have no effect on modifier, site, or onering layouts.

artifact_overlay

Normally, nanDECK detects Artifacts automatically and includes the appropriate golden icon overlay for the card’s culture (for both the culture icon in the upper right, and the card type bar).

However, in the event that a new card type is created that should also use this golden overlay, the `artifact_overlay` tag can be used to force its use on other cards using the modifier layout. This tag has no effect on cards using the character, site, or onering layouts.

When making `full_art` modifier cards, normally the overlay is disabled. This can be overridden to force its use with `full_art` by including `artifact_overlay`.

right_arrow

By default, all sites (standard or sanctuary) will use the arrow that points to the left. Using the `right_arrow` tag will override this to use the arrow that points to the right instead. The `left_arrow` tag is superfluous, but may be used for consistency if so desired.

highlight_twilight

Normally Followers are detected automatically and draw the associated yellow twilight icons. If you wish to use this effect on other non-Follower cards, use the `highlight_twilight` tag.

shadow, free_peoples

The nanDECK script has no awareness of cultures being associated with Free Peoples or Shadow sides, partially because of Gollum and partially to enable the possibility of dream cards that similarly cross the barrier. As a result, there are certain cases where nanDECK has no way of knowing which side to use, and will default to Free Peoples unless a shadow tag is included.

There are two situations where this may come up:

- Follower cards (or cards with the `highlight_twilight` tag) will default to using the Free Peoples version (the yellow twilight orb). If the Shadow version is required (the standard black diamond with a yellow fringe and a yellow drop shadow), the shadow tag is required.
Don't forget to also alter the `twilight_color` column to a yellow shade (such as

#F1CB50).

- When generating full_art cards, the twilight icons must be drawn (as they are normally included in the card template). By default the Free Peoples version is used, but this can be overridden by applying the shadow tag.

In both cases the free_peoples tag is superfluous, but may be used for consistency if so desired.

notes

This column is unused by nanDECK, so feel free to use this margin to keep any notes or comments on your cards when you feel the need to.

Customizing Generation

This script has a few separate settings files that can be customized to alter aspects of the card generation process:

settings.txt - contains a number of settings related to folder paths, filenames, and default values to use when the CSV has blank values in certain columns.

game_text_replacements.txt - this file contains a number of search/replace pairs that are run against all game text. You can add entries to, for instance, introduce custom keywords.

all_text_replacements.txt - this file contains search/replace pairs that are run against all title, subtitle, game type, and game text/lore text. This is mostly useful for automatically replacing, for example, “Ulaire Cantea” with “Úlairë Cantëa” across the board.

These files can either be edited in your text editor of choice (Sublime Text, Notepad++, Notepad), or they can be opened by nanDeck using the “Open deck” button. You can have multiple tabs open within nanDECK, which I would encourage you to use when editing these files and swapping back to the main generation script to see the results.

Each of these files is detailed below.

settings.txt

```
[card_template_path] = "templates"  
[card_image_path] = "card_portraits"  
[output_path] = "card_output"
```

These settings respectively control the directories used for templates, portraits, and card output. They can either be a relative path in the same directory as the script, or a full path (such as “C:\my_images”)

```
[game_text_font] = "LOTR_TCG"  
[title_font] = "LOTR_TCG_Title"  
[lore_font] = "LOTR_TCG-Lore"  
[number_font] = "LOTR_TCG_Icons"
```


Each of these settings control which font is used for different parts of the card. `game_text_font` is used for the game text, `title_font` is used for titles, subtitles, and the center card type bar, `lor_font` is used for lore, and `number_font` is used for the twilight, strength, vitality, resistance, and site number icons, as well as collector's info.

The name listed here must be the name that the font itself uses internally; this name may surprise you. Check your installed fonts to get the correct name, if the obvious one doesn't work.

```
[font_scale_factor] = 1.0
```

All text on the card will have its font size multiplied by this number. To make text twice as big, make it 2.0, to make it half as big, change this to 0.5.

This is intended to be used when providing custom fonts that are proportionally larger or smaller than the defaults. It could however be used in a pinch to pack in more text or increase size for readability.

```
[card_size] = double
```

Which size of template to use, either huge, double, or original.

Card Output Size	
Huge	1420 x 1988
Double	710 x 994
Original	355 x 497

(These sizes are slightly different from the actual template sizes due to the fact that nanDECK calculates its output resolution based on a physical size (3.5 by 2.5 inches) and a target DPI, rather than an exact pixel size.)

```
[include_size_in_card_output] = True
```

If True, then the id filename from the CSV will have the template size appended to it. For instance, if your filename is normally “example.png”, this being set to True will produce a filename of “example_huge.png”, “example_double.png”, or “example_original.png”, depending on the template size it was generated for.

```
[default_game_text_color] = #000000  
[default_icon_text_color] = #FFFFFF
```

Between these two settings, the default colors of all text on the page are controlled. `default_game_text_color` controls the game text, lore, type bar, title, subtitle, and collector’s info. `default_icon_text_color` controls the twilight cost, site number, strength, vitality, and resistance number fields.

Note that this value is only the default; each card can control values for individual elements based on data in the CSV. This is mostly used for debugging text issues (by setting one of these to bright pink, for instance, #FFFF00).

```
[card_mask_path] = card_mask.png
```

Included in the installation is a file called `card_mask.png`, which as a final step is used to slice the corners of the card clean. This path can be overridden to use a different file if a different

card edge or corner style is desired. Fully transparent pixels will be “cut off” of the card, while any other pixel will be left alone.

```
[debug_regions] = False
```

If this is set to True, then brightly-colored regions will be drawn beneath the title, subtitle, and game text areas to indicate how big they actually are. This is useful as the game text width and title/subtitle font size can be configured in the CSV, so it may be handy to help see exactly how far you can push it.

Output looks like this:



```
[auto_write_files] = True
```

If this is set to True (the default), then every time you regenerate a card (by building the entire deck, or by right-clicking the preview to regenerate a single card), the image will be saved to disk in the output directory. This slows things down considerably, especially at larger template sizes. If you're doing lots of text changes or otherwise want the process to be faster while you tinker with your cards, turn this value to False. If you do however the batch file won't work, so once you're ready to export, make sure this is set to True!

game_text_replacements.txt

This file controls several types of search/replace operations that are automatically performed on all game text and lore. Remember that entries in this file *only* affect those two things and no other types of text; see [all_text_replacements.txt](#) if you want to affect title/subtitle, etc.

The following things are automatically wrapped in tags, making them bold:

- All numeric loaded keywords (**damage +1**)
- All loaded and unloaded keywords at the start of a card (**Archer. Ranger.**)
- All phase actions and response (**Skirmish:**)

Loaded and unloaded keywords that are not capitalized will not be automatically bolded. For instance, the phrase "Spot a ranger to exert an archer" is left alone, and if you want it to be bold you will have to put manual tags in your CSV.

Culture tags are automatically converted, using either parenthesis or square brackets. In other words both [dwarven] and (dwarven) are replaced with the Dwarf culture symbol in the LOTR-TCG font. This is not case sensitive.

The numbers 1-9 and X wrapped in square brackets or parenthesis are automatically converted to the appropriate twilight symbol in the LOTR-TCG font. E.g. [1], (2), (X) all convert.

All single quote (') and double quote (") symbols will also be replaced with their smart quote equivalents.

If you want to add your own entry for keyword formatting replacement, it must look as follows:

```
HTMLKEY=,"Battleground.,"<b>Battleground.</b>",,C
```

In this case, the first quoted string is the text to search for, the second quoted string is the text to replace it with (which in this case is the same thing with the bold tag wrapped around it), and the final argument is whether or not it is case-sensitive. Put a C for case sensitive, or delete the C to be case-insensitive.

Standard HTML codes are supported by nanDECK, including **bold**, *italic*, and underline.

If you want to add more symbol replacements, use the following type of entry:

```
HTMLKEY=,"(moria)","&#9788"  
HTMLKEY=,"\91\moria\93\","&#9788"
```

The first value is the text to search for, the second the text to replace it with. In this case since we are replacing a tag with a unicode character, we have to put the HTML-encoded unicode symbol, which is &# followed by the 3-5 numbers of the unicode id.

The second line here is how to support tags with square brackets, which are a special character in nanDECK. \91\ means ASCII code 91, or [, while \93\ is ASCII code 93, or].

all_text_replacements.txt

This file controls search/replace operations that are automatically performed on all text, including game text, lore, title, subtitle, and card type bar. This is most useful for automatic common spelling mistake correction or accents.

Any digit 1-9 with a T or a K will have the tower block symbol or king block symbol automatically included. For example, 3T or 5K. If for some reason the block symbols need to be used separate from numbers, then they are also automatically replaced when in parentheses or square brackets, e.g. (T) or [K].

A list of 60 names that have accents in them are also included, so you never have to letter hunt again while typing Theoden, Smeagol, or Ulaire Cantea.

Find/replace entries in this file use the same format as the ones in [game_text_replacements](#), so see the section there for details on formatting.

If creating your own name entries, here is a handy table for the decimal unicodes for common

Tolkien accents. If you find yourself creating a replace entry for a name that is made by Tolkien, please message [TelTura on the TLHH forums](#) and I will consider adding it to the list of names for everyone.

Á	Á	á	á	Â	Â	â	â	Ä	Ä	ä	ä
É	É	é	é	Ê	Ê	ê	ê	Ë	Ë	ë	ë
Í	Í	í	í	Î	Î	î	î	Ï	Ï	ï	ï
Ó	Ó	ó	ó	Ô	Ô	ô	ô	Ö	Ö	ö	ö
Ú	Ú	ú	ú	Û	Ú	û	û	Ü	Ü	ü	ü

Making Custom Resources

The packaged materials will get you as far as creating basic cards for all cultures and all card types as released by Decipher, but the program also supports expanding on the available material out-of-the-box.

Custom Templates

Take a peek at the files inside the templates folder:



Templates use the following naming scheme:

Culture_layout_size.png

Layout is one of the following (it is omitted in the case of sites):

- character
- modifier

Size is one of the following:

- huge
- double
- original

Culture is not enforced to any degree. So long as the value in the CSV “culture” column lines up with a file in the templates folder, there are no other requirements.

Every culture has a character template and a modifier template, but it is not required to provide both. For example, in the image above you can see the “moria_balrog_character_double.png” file; Decipher never released a modifier template using the red-tinged Moria template, so there is no modifier version provided here. Any cards that wish to use this template will just need to have the “culture” column set to “moria_balrog” (and the “template” column set to “character”) and this template will get picked up automatically.

Likewise, the Gollum culture is actually represented here as two separate cultures, one called “gollum_freeps” and another called “gollum_shadow”. To add your own custom templates all it would take is to add a template file that follows a consistent naming scheme, and to reference that same culture name in your CSV.

For example, let’s say you wanted to create a template for one of the Legends Masterworks series of cards, such as the one used by [Gandalf, The White Rider](#).

1. Do all the photoshop work to create the white template
2. Follow the naming scheme to call the file “gandalf_masterwork_character_double.png”.
3. Place the file in the templates/double_res_export folder
4. Make a CSV row that has the culture of “gandalf_masterwork” and template of “character”

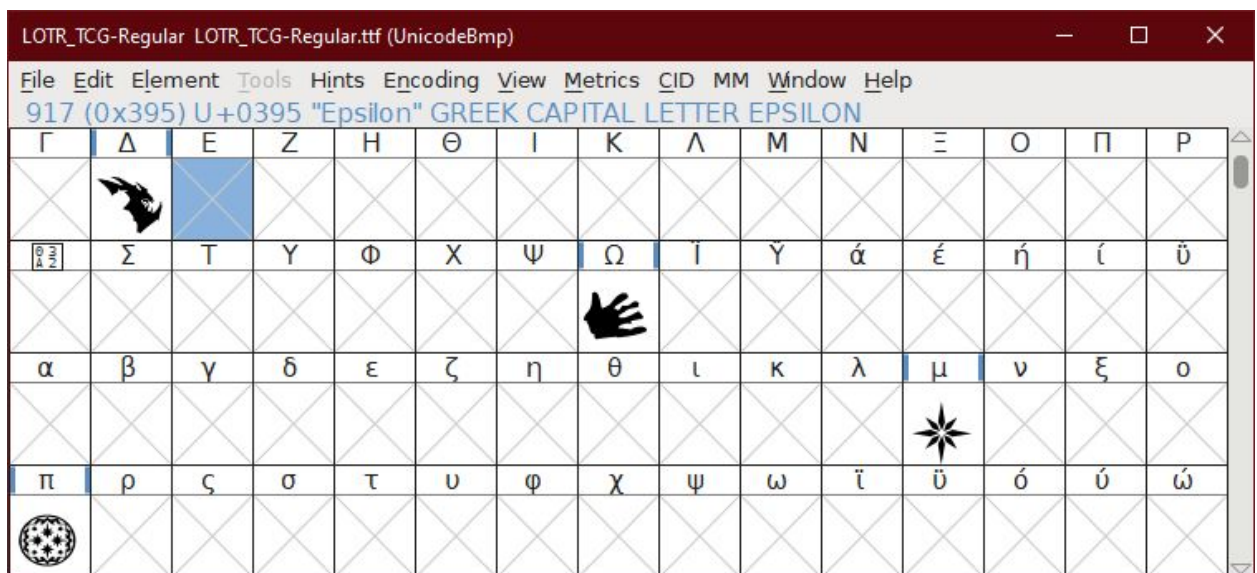
This should be all you need to do to create custom templates.

Custom Culture Icons

Most of the work involved in creating your own custom cultures is in getting the templates just right in the art department. Following the instructions under Custom Templates above will get you where you need to go, but there's also the matter of referencing cultures in text with the culture icon.

Here are the steps you need to follow to add a new icon:

1. Create a culture symbol that is clean, distinct, and easy to read. This is harder than it sounds.
2. Create an SVG single-color version of your culture symbol, using something like [Inkscape](#).
3. Download [FontForge](#) and use it to open up the LOTR_TCG-Regular.ttf font
4. Select some random symbol that is unlikely to be used in actual text and paste the SVG within it. This will take some tweaking to get the exact positioning and scale correct (note that some symbols such as Rohan are much lower than you might expect).



5. Take note of the number of the symbol you chose. In the above image I've selected the Epsilon slot right next to where Moria was placed, and there is blue text noting that it is code point "917" (the decimal representation of the hex unicode offset U+0395).

6. Export your new version of the LOTR_TCG-Regular font by going to File -> Generate Fonts
7. Double-click the TTF file and click Install like you did when you first installed the font
8. Modify **game_text_replacements.txt** to include a search/replace rule for your new culture. Put the code you noted in step 5 in the replace text. If your new culture is Dunedain and you put its symbol in the slot highlighted above, then that rule would look something like:

```
HTMLKEY=,"(dunedain)","&#917"  
HTMLKEY=,"\91\dunedain\93\","&#917"
```

And that should be it! Reference [Dunedain] in the game text column of your CSV and watch it get replaced automatically.