Card Maker 0.98.0.10 (beta)
by Tim Stair

Table of Contents

Introduction

Card Maker is an application for generating layouts of information in a graphical representation. (a wordy way of saying "it can make business cards or game cards") Card Maker was inspired by the need to generate cards for prototyping board games quickly without a lot of overhead to make minor changes. The goal of the application is to simplify the process by which a card layout is created and decrease the amount of time required to make minor changes.

Warning

All information in this document is subject to change... of course. Card Maker should be used AT YOUR OWN RISK. The author of the application is not responsible for any damages caused by its use. (though I certainly hope nothing horrible will happen)

User Interface

Main Window

The main window has limited functionality and acts as a container for all of the other windows.

File Menu Item	Function
New Project	Creates a new default project.
Open Project...	Opens a project file.
Save Project...	Saves the current project to the loaded file (or prompts the user if the file does not yet exist).
Save Project As...	Allows the user to save the project as a different file name.
Print Preview Project...	Displays the standard Print Preview dialog for the project.
Print Project...	Displays the standard Pint dialog for the project.
Export Project to Images...	Displays the Image Export dialog for the project.
Close	Exits Card Maker

Edit Menu Item	Function
Undo	Undoes the last element or layout change.
Redo	Performs the last undone element or layout change.

View Menu Item	Function
Draw Element Borders	Toggles the red outline drawn around each element in the Canvas window.
Draw Formatted Text Word Borders	Toggles an outline drawn around each word in a formatted text field.
Update Preview	Triggers the preview window to update.

Project Menu Item	Function
Clear Image Cache	Clears any cached images from memory forcing them to reload from disc. (allowing changes made by external programs)
Update Known Issues	Displays / Updates the known issues dialog. Each problem area will be transitioned to upon clicking on an issue. Note: Your project *must* be saved to use this feature.
Refresh Layout	Refreshes the current layout (including the reference data). Shortcut: F9

Tools Menu Item	Function
Project Manager	Viewer for all projects within a given directory structure.
Remove Layout Templates	Dialog for removing layout templates.
Illegal Filename Character Replacement	Configures the characters to replace those that are invalid on export of the project. (empty by default)
Update Google Credentials...	Allows you to update the login/password used when connecting to a Google Spreadsheet reference.

Project window

The Project window contains a tree view indicating the card layouts and their associated reference files.

Renaming a Layout

To rename a Layout left-click a layout node and don't move the mouse until the editable text field appears. (like renaming a folder in windows explorer)

Context Menu Items (Layouts Node)

Menu Item	Node Type	Function
Add Card Layout...	Layouts	Adds a new layout based the parameters supplied in the dialog.
Add Card Layout From Template...	Layouts	Adds the selected Layout template from the dialog with the name specified.
Set Name Format...	Layout(s)	Adjusts the naming scheme associated with the project or layout when exporting images. Special codes in the name format: @[column_name] – A value from the reference ## – card number #L – layout name

Context Menu Items (Layout Node)

Menu Item	Function
Duplicate Layout	
Define as Template Layout...	Adds the selected layout to the list of layout templates. All aspects of the layout (including elements) are persisted except the name.
Remove Card Layout	Removes the selected layout
Print Preview Card Layout...	Displays the standard Print Preview dialog for the selected layout.
Print Card Layout...	Displays the standard Pint dialog for the selected layout.
Export Card Layout as Images...	Displays the Image Export dialog for the selected layout.
Add Reference...	Adds a reference file (CSV) to be associated with the selected layout
Add Google Spreadsheet Reference...	Adds a reference to a Google Spreadsheet. Each time you open the project the login/password will be requested.
Set Name Format...	Adjusts the naming scheme associated with the project or layout when exporting images. Special codes in the name format: @[column_name] – A value from the reference ## – card number #L – layout name

Context Menu Items (Reference Node)

Menu Item	Function
Set as Default Reference	Sets the selected Reference as the default for the layout associated with it.
Remove Reference	Removes the selected Reference from the list associated with the parent layout.

Layout Control window

Card Count Group

Control	Function
#	Sets the default card count. This is intended for layouts that do not have a reference. Layouts with references cannot use this function.
Numeric up/down	This sets the card index for this layout. Use this to cycle through all of the cards associated with this layout.

Card Layout

Control	Function
Buffer	Sets the vertical and horizontal buffer between cards when printing.
DPI	Sets the DPI of the given layout. This only affects the printed / exported layout. The layout is still rendered 1:1 on the screen. (example: if you want a 3.5" wide card at 300dpi then the width needs to be 3.5 x 300 = 1050)
Width	The width of the layout in pixels.
Height	The height of the layout in pixels
Draw Border	Draws a black border 1 pixel wide on the edges of the layout. This is primarily intended for cutting cards out after printing.
Elements	A list of all of the elements in the layout. If an element is highlighted it is the selected item and can be affected in the Canvas and Element Control windows. The enabled value indicates whether to draw the element or not. You can toggle an elements visible by double-clicking or pressing the space bar. The order of the elements indicates how the items will be drawn. The bottom item is draw first and proceeds to draw the others up to the top element.
Add	Adds a new element.
Dupe	Duplicates the selected element.
Remove	Removes the selected element.
Rename	Renames the selected element. <u>Note</u>: Duplicate names of elements are not allowed within a layout.
Up / Down	Moves the element up or down in the list affecting draw order.

The table below contains shortcuts for use in the Elements list box.

Description	Keyboard Shortcut
Move selected element up 1 level	Shift + Up
Move selected element down 1 level	Shift + Down
Select the element 1 level up	Up
Select the element 1 level down	Down

You can also select 1 or more of the layout elements to move them in the Canvas window.

Element Control window

Element

Control	Function
Type	The type of element.
X	The x position of the element. 0 is the far left. X increases moving right.
Y	The y position of the element. 0 is the far left. Y increases moving down.
Width	The width of the element.
Height	The height of the element
Opacity	The opacity of the element. (applies to all types)
Rotation	Rotation of the element around the center of the element definition position.
Spreadsheet Data	Above the Definition text area there is a view of the current data associated with the given card index. You can right-click in the view to add a reference (to the given column) to the Definition field.
Definition	The text string defining the element contents. (see Definition / Scripting Reference for more details) When working with a Graphic element type the "..." button may be used to browse to and select supported graphic files. The "+" button to the right of the Definition field will pop up a list of possible items to insert into the Definition text area. (see Definition / Scripting Reference for more details)

Font (Text Elements)

Control	Function
Font	The font to render the text with.
Color	The color to render the text with.
Bold	Draws the font in bold. (may not be supported by all fonts)
Italic	Draws the font italicized. (may not be supported by all fonts)
Strikeout	Draws a line through the rendered text. (may not be supported by all fonts)
Underline	Draws a line under the rendered text. (may not be supported by all fonts)
Size	The size of the font to render.

Control	Function
Auto-Scale	Attempts to scale the font of any text that overlaps the element borders down to the desired element size. Warning: This can be CPU intensive. You should also setup your element with an optimal font size. The font size you specify in the element is the target font size maximum The more scaling performed the less accurate the output. (though your string should still appear in its entirety) **Note:** This can only be used with <u>Text</u> elements.
Line Spacing	Adjusts the spacing between text lines. **Note:** This can only be used with <u>FormattedText</u> elements and <u>Text</u> elements where \n is used.
Word Spacing	Adjusts the spacing between words (can be negative). **Note:** This can only be used with <u>FormattedText</u> elements.
H Alignment	The horizontal alignment of the text within the element space.
V Alignment	The vertical alignment of the text within the element space.

Notes:

- The text will appear to be red until the font has been set once.

- Any strings that are too large to fit in the desired element will be drawn with ellipsis to help find issues. The ellipsis will *not* be present if Auto-Scale is used.

Graphic (Graphic Elements)

Control	Function
Lock Aspect	Locks the image aspect ratio when drawing.
H Alignment	The horizontal alignment of the graphic within the element space. *Note: This only apples when Lock Aspect is checked.*
V Alignment	The vertical alignment of the graphic within the element space. *Note: This only apples when Lock Aspect is checked.*
Set Size To Image	This will set the size of the element to be that of the original image file.

Shape (Shape Elements)

Control	Function
Color	The color to render the shape with.

See shape information in the Definition / Scripting Reference section.

There is also an assistant to help define the shape. Each time you change a property value the definition of your element will automatically change to the settings indicated.

Border

Border is an outline for the element itself (the rectangle defining the object).

Control	Function
Color	The color of the element border.
Thickness	The thickness of the border (in pixels). If this is 0 no border is drawn.

Outline

Outline is an outline for the drawn element itself. (text/shape outline)

Control	Function
Color	The color of the outline border.
Thickness	The thickness of the outline (in pixels). If this is 0 no outline is drawn.

Canvas window

The Canvas is a WYSIWYG (what you see is what you get) editor for the element shape and location. It directly effects the Element Control window values. It is highly recommended that you further tweak any location/scale values in the Element Control window for neatness sake!

The currently selected Element in the Layout Control window will be drawn with a green border. This item can be moved and manipulated like many other standard graphics tools. I'm not going to detail out how to do this. If you're new to graphics editing you should spend some time in the Canvas manipulating an element to see how it functions.

Note: The green border on selected items will not appear if you have disabled the **Draw Element Borders** option.

You can use the right mouse button to select the element under the cursor. If more than one item is under the cursor a context menu will display a list to select from.

The table below contains shortcuts for use in the Canvas.

Description	Keyboard Shortcut
Move selected element up 1 level	Shift + Up
Move selected element down 1 level	Shift + Down
Select the element 1 level up	Control + Up
Select the element 1 level down	Control + Down
Move selected elements up/down 1 pixel	Up / Down
Move selected elements left/right 1 pixel	Left / Right
Lock to axis	Shift (and move the mouse along the desired axis to lock to)
Zoom In / Out	Ctrl + +/- (+ zooms in, - zooms out)
Toggle Mode – switch between resize-move(normal) and move-only	m

Note: Zoom may result in a slightly inaccurate representation on things that are measured programatically. (this is a bug) Formatted Text elements in particular may be laid out slightly

differently when zoomed in. It is recommended that you always review your work at a zoom of 1.0.

Logger window

Context Menu

Control	Function
Copy Line to Clipboard	Copies the selected line to the clipboard.
Copy all text to Clipboard	Copies all of the text in the logger to the clipboard.

Preview window

The preview window is opened by pressing F5 or accessing the View > Update Preview menu item. This dialog allows you to see a scaled version of the Canvas. You can adjust the zoom from 10% to 100% using the numeric field.

CSV Reference Files

Creation

CSV reference files have 3 strict content rules:

1. The first row must contain the column names. The columns may be named with spaces but should not use the [or] characters (or anything else that might confuse the parser when trying to translate references).

2. The first column may be named anything, but the contents should always be the card count of the given row. (an empty entry assumes 1 instance of the given row)

3. Never use quotes in the strings in Excel or Calc (OpenOffice). Both applications handle this differently and it makes a mess of things. The automatic quotes around a string (if you look at your CSV file in a text editor) are necessary and should be left as is. Use the "\q" escape code (as described below) if you need a " character.

Special Functions

Function	Definition Details
Comment out a line.	In the first column set the card count to 0.
Specifying the layout	You can specify the allowed layout for a card by name. Any row without a matching layout name with the current layout will not be included in the list of cards. Name the column **allowed_layout** in your CSV file and specify the desired layout for every card.

Color String Format

Colors are formatted into strings as follows:

RRRGGGBBB
or
RRRGGGBBBAAA

Each block is a 3 digit representation of the color value for the given color. (R – Red; G – Green; B – Blue; A - Alpha)

Example: 012123167

You can also specify a color using hexadecimal values:
RRGGBB
or
RRGGBBAA

Example: 882859

Be sure to include any leading zeroes. The type of color value you specify (hex vs. decimal) is based solely on string length: 6/8(hex)

Some editors may remove the leading zeroes. If you are specifying the color as hex you can insert **0x** at the beginning of the string (ie. 0X882859). **I highly recommend doing so.**

Element Overrides

You can override an element's settings per card in the CSV by providing a column with the following format:

override:[element name]:[element value]

Example: **override:text_field:x**

Any values specified in this column will override the **text_field** element's **x** value. When rendered the element's x location would be that of the override value. (locking the horizontal position in the example)

Any colons involved in the name of an element will cause the override to fail. **DO NOT USE COLONS IN YOUR ELEMENT NAME**.

This will completely override any changes you make in Card Maker. When rendered the override will take effect. Any empty value specified within the column is ignored.

The various field type values must be specified correctly for the program to function correctly. See the table below for formatting the content of your override column. This table also contains the recognized field names. All field names are lower case.

Fields	Override string format
x	**(int)** Integer number values.

Fields	Override string format
y width height borderthickness opacity outlinethickness lineheight wordspace (+/-)	
autoscalefont enabled lockaspect	**(bool)** Boolean values: **true** or **false** (case should not matter)
rotation	**(float)** In the case of rotation use only integer number values.
bordercolor elementcolor outlinecolor	**(string)** (See the section Color String Format)
font	**(string)** String dictating the construction of a font as follows: *fontname;fontsize;bold;underline;italicized;strikeout* *fontname* – string name of a font ("Ariel") *fontsize* – size of the font (like the size in a word processor) *bold* – 1 or 0 (on or off) *underline* – 1 or 0 (on or off) *italicized* – 1 or 0 (on or off) *strikeout* – 1 or 0 (on or off) See one of your project files for an example font string (they are viewable in notepad or any other text editor). An invalid font string may result in crashes or other undesirable outcomes!
verticalalign	**(int)** Numeric value indicating the vertical alignment: 1 – Near 2 – Center 3 – Far
horizontalalign	**(int)** Numeric value indicating the horizontal alignment: 1 – Near 2 – Center 3 – Far

Fields	Override string format
variable	**(string)** The **Definition** field.
type	**(string)** One of the following strings: **Text** **Shape** **Graphic** **FormattedText**

Definition / Scripting Reference

The information below is intended for use in the Definition field of the Element Control. While it is possible to put this information in the CSV files driving the card it is not recommended.

#empty – This can be used in place of an empty string. Example:
#if(x == then a)#
is the same as
#if(x == #empty then a)#

All Elements

Function	Definition Details
Referencing a Data Source item	@[**columnname**] Replace **columnname** with the desired column name from the referenced data source.
Counter	##x;y;z# This will display a number based on the following properties: x + (current card index * y) with left padded 0's numbering z Example (on card index 5) with ##1;1;5# would be **00005**

Function	Definition Details
If Statement	Samples: #(if x == y then a)# #(if x != y then a)# #(if x == y then a else b)# #(if x != y then a else b)# **==** - x is equal to y (case insensitive) **!=** - x is not equal to y (case insensitive) **>** - x is greater than y **<** - x is less than y **>=** - x is greater than or equal to **<=** - x is less than or equal to x,y,a,b – string values **#nodraw** is supported as a result to trigger the element to not be drawn at all. You can specify an empty string by simply entering nothing. An example of an empty comparison is as follows: #if(x == then a)#
If Statement (grouped)	You can use the if statement with groups. The x and y values listed above can be formatted as: [val1;val2;val3] Allowing multiple items to be compared to multiple other items. The x set of strings is sought in the y set. The comparison is successful if all of the items in x exist in y. (the y set is not checked for existing in x) NOTE: This does not apply to numeric comparisons. Only **==** and **!=** are supported. TBD – keep this functionality?

Function	Definition Details
Switch Statement	#(switch;key;keytocheck1;value1<repeat>)# key – The value to check the others for. keytocheck1 – A comparison string value1 – A value string (this will be the overall result if key matches keytoccheck1) ;keytoccheck1;value1 can be repeated multiple times with varying strings. There should always be a key and a value. (an empty string is valid) If the key is **#default** the value will be the default if no other keys match. **#nodraw** is supported as a result to trigger the element to not be drawn at all.

Graphic Elements

Function	Definition Details
Referencing an Image	The path to the file should be specified. The path may also be relative to the project file. Supported Image Types: BMP, GIF, EXIG, JPG, PNG, TIFF, and PSD. PSD file support is rudimentary. 8 bit per channel PSD images should be functional. (RGBA) Relative path example: **images\file.bmp** would be valid if the project file were stored in the folder where the folder **images** existed
Draw no image	**none** (case insensitive) This will indicate to not draw anything. Note: Any invalid file path / name will also not draw.

Text Elements

Function	Definition Details
New Line	**\n** – Will be replaced with a new line
Quote Character	**\q** – Will be replaced with a " (necessary for CSV file annoyances such as OpenOffice and Excel handling of strings with commas in them)
Comma	**\c** – Will be replaced with a comma

Formatted Text Elements

These elements support HTML like tags for mid-line formatting. This is very new and may be buggy. Some fonts do not display the same between styles. They will appear to be vertically offset by some number of pixels. I recommend finding a font that does not have this issue (as there may be little I can do to resolve the issue on a per font basis).

Using the outline functionality with Formatted Text is not recommended. There are numerous draw issues that may come up. Be very careful to check the results before assuming everything is working correctly if you use outlined Formatted Text.

Function	Requires Closing Tag	Definition Details
Bold	Y	**\<b\>** – Will change the text within to bold
Italic	Y	**\<i\>** – Will change the text within to italicized
Strikeout	Y	**\<s\>** – Will change the text within to be drawn struck out
Underline	Y	**\<u\>** – Will change the text within to be underlined
New Line	N	**\<br\>** – Will be replaced with a new line
Space	N	**\<spc=#\>** - Number of spaces to add. May also be used without specifying the number to indicate a single space: **\<spc\>** (if for some reason you need that)
Background Color	Y	**\<bgc=########\>** - Will create a background color within the tag. (See the section Color String Format)
Background Image	Y	**\<bgi=[filename]\>** OR **\<bgi=[filename];[#];[#];[#];[#]\>** - Will create a background image behind each given word in the tag. The first option will draw the image at full size in the upper left hand corner of the given word. The second option has the following fields: 1. x offset 2. y offset 3. width 4. height
Font Size	Y	**\<fs=#\>** - Will change the size of the font for the words within the tag. **This feature is not functioning as desired and is not recommended.**
Font Color	Y	**\<fc=########\>** - Will change the font color within the tag. (See the section Color String Format)

Function	Requires Closing Tag	Definition Details
Y Draw Offset	Y	**<yo=#>** - Will affect the Y offset of the word when drawing. This is recommended when a style (bold or italic) does not align correcly with normal text. The number value may include the decimal point for more precise tuning.
X Draw Offset	Y	**<xo=#>** - Will affect the X offset of the word when drawing. This is recommended when a style (bold or italic) does not align correcly with normal text. The number value may include the decimal point for more precise tuning.

NOTE: The line spacing must be configured. Without configuring the line spacing each line will draw on top of the previous.

Shape Elements

Basic shape elements are described by this format: (Rectangles/ Ellipse)
#shapename;**thickness**;**horizontal_size**;**vertical_size#**
Other complex types have the same basic format with extra parameters:
#shapename;**thickness**;**horizontal_size**;**vertical_size;[extra parameters]#**

Value	Description
shapename	the name of the shape to be drawn
thickness	The thickness of the pen to draw with. 0 will fill the shape.
horizontal_size / vertical_size	The horizontal/vertical size in pixels of the rectangle to draw within. This overrides the Element size. The values may be negative to draw above or to the left of the Element.

Additional Parameters by Shape Table

shapename	Additional Parameters
roundedrect (rectangle with rounded edges)	**[EdgeSize]** EdgeSize – The distance to round from the corner.
grid	**[AllowPartialGrid];[GridWidth];[GridHeight]** AllowPartialGrid – Allows the grid to draw in an incomplete form on the right and bottom GridWidth – The width of a single rectangle in the grid. GridHeight – The height of a single rectangle in the grid.

Grid Note

- The partial grid option is only a best guess. If your thickness and other variables are not configured precisely you may find the last line draw incorrectly or missing entirely (generally obvious with 1 thickness). This issue may be resolved with a code fix in the future.

Order of Translation

1. Data Source References (@[x])
2. If
3. Switch
4. Text translations (Note: These only apply to Text elements)

Note: It is possible to create an endless loop if you dare enter @[?] values in the csv file. Enjoy! (why would you do this?)