

Libro Game Creator 3

MANUAL



Web: librogamecreator.wordpress.com

Author: Matteo Poropat (mefistofele74@yahoo.it)

Updated to version: 3.2.7

Date: 28/03/2016 (doc. rev. 115)

Table of contents

1	INSTALLATION.....	3
2	INTERFACE.....	4
2.1	Keyboard shortcut.....	4
2.2	Using LGC.....	4
3	BASIC ELEMENTS.....	6
3.1	Entities.....	6
	<i>System entities</i>	6
3.2	Attributes.....	6
	<i>System attributes</i>	6
4	GAMEBOOK MANAGEMENT.....	8
4.1	Exporting.....	8
	<i>RTF</i>	8
	<i>SQLite</i>	8
	<i>HTML</i>	8
	<i>eBook</i>	8
	<i>Graphviz</i>	8
	<i>ChoiceScript</i>	8
4.2	Importing.....	8
	<i>SQLite</i>	8
	<i>LGC2</i>	8
	<i>TXT</i>	8
4.3	Using the Shuffle.....	9
5	CONFIGURATION.....	10
6	SQLITE FORMAT.....	11

1 Installation

Libro Game Creator needs a PC running Windows, MacOS or Linux, with Java installed. If you're not sure it is present, open a terminal window and run the command

```
java -version
```

In case you don't have it already installed, you can download Java from

<https://www.java.com/it/download>

and follow the instructions to install it.

LGC needs **Graphviz** to generate the gamebook's graph. The program could be downloaded from <http://www.graphviz.org/Download..php>

To install LGC download the file (.zip or .tgz), uncompress it on your PC, anywhere you want, and run the correct launcher:

- **run.cmd** on Windows systems
- **run.sh** on Linux systems

2 Interface

GUI (Graphical User Interface) comes with two different modes: *basic* and *advanced*.

- advanced mode shows a full toolbar with all the available commands, and the attributes editor
- basic mode has a short toolbar with main commands and big icons from the old LGC 1.0.2 version

LGC's main frame are divided in tree big sections: toolbar, entity's tree and the editor.

2.1 Keyboard shortcuts

CTRL+N	New book
CTRL+Q	Quit
CTRL+E	New entity/chapter
CTRL+D	Remove selected entity
CTRL+M	Open entity's property editor
CTRL+A	Add an empty attribute to the selected entity
CTRL+Z	In the text editor: undo the last change
CTRL+F	Open the find&replace panel
CTRL+G	Open "go to chapter" dialog
CTRL+P	Open the map editor
F2	In the text editor: insert a link tag {link n:text} in the text In the chapter's tree: insert or modify the chapter's title
F3	Insert a link tag {link M:text} where M is the number of the first chapter available (if you've created chapters 1, 2 and 4, the generated link will be {link 3:text})
F4	Add a {todo} tag, used to keep trace of future works. In the main menu click View → Show todo list to see the printable version of the list
F5	Insert a link to the first available chapter in the standard gamebook format {link <number>:<number>}

2.2 Using LGC

LGC is a very simple software. You could start to write down your interactive novel. All you have to do is, when you want to add a link from the chapter you are writing to another, press F2.

With this command you insert a tag in the text: {link n:text} where n is the number of the next chapter, text is a phrase.

If you want to use the old style of gamebooks, you'll write something like (examples are from Joe Dever's *The Chasm of Doom*):

*If you wish to investigate the hut, turn to **{link 160:160}**.*

*If you wish to ignore the hut and continue on your ride along the Ruanon Pike, turn to **{link 273:273}**.*

When you complete your novel and export it in HTML the result will be something like:

If you wish to investigate the hut, turn to [160](#).

If you wish to ignore the hut and continue on your ride along the Ruanon Pike, turn to [273](#).

Otherwise, you could link the text:

*Do you prefer to **{link 160:investigate the hut}** or **{link 273:ignore it and continue on your ride along the Ruanon Pike}**?*

and obtain this in the HTML exported file:

Do you prefer to [investigate the hut](#) or [ignore it and continue on your ride along the Ruanon Pike](#)?



Using the **F5** shortcut you insert a link to the first available chapter with the title option: **{link <number>:@T}**. If "export chapter's title" is active in the Settings page, LGC try to use chapter's title instead of numbers, when exporting the book.

When the option is active and the chapter has a title, it will be used in the output document. If the option is not active or the chapter does not have a title, the chapter's number will be used.

3 Basic elements

The basic component of LGC is called **entity**, and it has been developed to let LGC operate with more complex structures than the "simple and classic" printed gamebook.

3.1 Entities

There are two groups of entity

- **book's chapters**, main part of the LGC data, where you put the story
- **correlated entities**, everything could be useful to build something more complex like interactive applications, smartphone apps,... A correlated entity could contain:
 - **book's data** (author, title,...)
 - **game's data** (items, flags, variables,...) those entities are a representation of the state of the game

Every entity is defined by

- **a name (number if the entity is a chapter)**
- **a type**: chapter or generic entity
- **a group**, an alphanumeric identification

System entities

Under the node entities/setup you'll find gamebook standard entities where you put:

- **introduction**
- **rules**
- game's **data**

3.2 Attributes

Every entity has a certain number of attributes.

Every attribute is defined by:

- **a name**
- **a type** string/integer/boolean
- **a value**

System attributes

Paragraphs have some standard attributes that can not be edited or deleted. Those are the three flags described below and the chapter's text.

	Fixed (flag_fixed)	This paragraph is fixed, so it could not be shuffled with the rest of the book
	Ending (flag_final)	One of the book's ending

	Deadly (flag_death)	One of the book's chapters where the hero will find his death
---	----------------------------	---

4 Gamebook management

4.1 Exporting

RTF

A Rich Text Format file, with rules and intro. Chapters are linked, you could navigate the document (with ctrl-click)

SQLite

A SQLite database.

HTML

There are two type of HTML exportation:

- a single file with intro, rules and every chapter
- one files per chapter, plus one for the intro and one for the rules

eBook

An ePub2 ebook, one files per chapter, plus one for the intro and one for the rules.

Graphviz

A graph of the gamebook. LGC creates a .dot file, then starts Graphviz to transform the dot file in a JPG and in a SVG file.

More information (and download) <http://www.graphviz.org>

ChoiceScript

More information in the official website: <http://choicescriptdev.wikia.com>

4.2 Importing

SQLite

A SQLite database.

LGC2

Older LGC version.

TXT

A text file formatted like the following example. Chapter's number could with or without the period. Keywords rrepresenting links between chapters are listed in the **cfg/txtkeywords.list** file.

1.

'Point and flankers,' you shout, and immediately three groups of rangers peel away from the column and spur their horses to a gallop. You watch with pride as the expert horsemen take up their scouting positions to the front and side of the company. It is late afternoon when a ranger scout approaches the company from the west. He points towards a craggy outcrop where a thin spiral of wood smoke betrays a hut hidden beneath

the overhanging rock.

*If you wish to investigate the hut, **turn to** 160.*

*If you wish to ignore the hut and continue on your ride along the Ruanon Pike, **turn to** 273.*

4.3 Using the *Shuffle*

TBT

5 Configuration

You can change LGC parameters using **File / Preferences** menu to open the configuration window.

6 SQLite format

In LGC you could export the gamebook as a SQLite database, very useful to use in smartphone application, like an Android gamebook player.

The database is simple, it has two tables.

This is the script needed to create the db:

```
CREATE TABLE T_ENTITY (  
  id_entity] INTEGER NOT NULL PRIMARY KEY,  
  entity_name] TEXT NOT NULL,  
  entity_type] TEXT NOT NULL,  
  entity_group] TEXT NOT NULL  
);  
  
CREATE TABLE [T_ENTITY_ATTRIBUTES] (  
  id_entity] INTEGER NOT NULL CONSTRAINT [FK_ENTITY] REFERENCES [T_ENTITY]([id_entity]) ON DELETE  
  CASCADE ON UPDATE CASCADE MATCH FULL,  
  [id_attributes] INTEGER NOT NULL,  
  attribute_name] TEXT NOT NULL,  
  attribute_type] TEXT NOT NULL,  
  attribute_value] TEXT NOT NULL,  
  CONSTRAINT [sqlite_autoindex_T_ENTITY_ATTRIBUTES_1] PRIMARY KEY ([id_entity], [id_attributes])  
);
```