Escape Room Software

User guide

This documentation is associated with Escape Room Software version v5.0.0.0
© Escape Room Sofware 2016-2024, www.escaperoombuilder.com

Table of contents

3

1. Introduction

Escape Room Software is a software solution for real escape rooms. Basically it displays a countdown timer on one or multiple screens in an escape room so your players know how much time is left in the game.

Do more with ambiance

But you can do so much more! Perhaps influence the mood and ambiance by playing music. You can send text messages when players need a hint. Or weave an intricate storyline by sending videos or images to the screen at appropriate times.

Do more with player interaction

Add to that some advanced interaction possibilities: if the in-room screen is a touchscreen, you can provide players with a hint button or perhaps a quiz to earn extra time at a specific junction in the game.

Do more with external interaction

Escape Room Software works with a simple script editor that you can use to create your own preset sequences. These scripts can also be kicked off by external input, for example by the players pressing a button on the touchscreen, or an advanced room prop connected to a microcontroller-device (for example an Arduino), specific Home Automation Solutions, or even external websites (using an API).

Focus on what makes your room special!

Escape Room Software avoids complex technical dependencies for your core game. Suitable for single rooms as well as managing multiple rooms from a central chamber.

In this way, Escape Room Software can become the solid backbone of your escape game and provide a high level of user interactivity, increasing the immersiveness for both the player and the game master.

2. Feature overview

2.1. Basic

	Description	Details
	Countdown timer	Hours, minutes or cyclic
	Play intro movie	Optional, autostart timer after finishing movie
	Send messages	Manually using buttons, or scripted, colored
	Send alerts	Manually using buttons, or scripted
	Send images	Supported filetypes (.jpg, .jpeg, .png, .bmp)
	Play audiofiles	Supported filetypes (.mp3, .wav)
	Play & skip through video files	Supported filetypes (.avi, .mp4, .mov), timer sync.
	Manipulate countdown timer	Manually using buttons, or scripted, movie sync.
	Scary flicker effect	

2.2. Customization

	Description	Details
	Customizable layout	Texts, images, fonts, colors, sizes, locations
	Easy-to-use wizard & scripting tools	
	Manual action scripts	
	Timed action scripts	
	Variables	Define, update, compare, randomize
	Save customized rooms as .ZIP file	
	Scripting functions glossary	Also available in-software
	Sample room included	Showing various functions

2.3. Client interaction

	Description	Details
	Client-side touchscreen buttons	
	Client-side keyboard/pad/barcode scanner support	Requires keyboard, numpad or (USB)barcode scanner
	Run external processes on client or server	
	Accepts external Serial or TCP commands to kick off scripts	*For example:* messages sent by an external application, or (micro)controllers like Arduino or Raspberry Pi
	Send a custom Serial or TCP command	*For example:* to send messages to an external application, or (micro)controllers like Arduino or Raspberry Pi
	Home Automation integration	Z-Wave integration
	API	Receive and send API calls

2.4. Technical

	Description	Details
	Simple MSI installer	
	Host one or multiple rooms using one or more devices	One computer for SERVER and CLIENT or use one server computer and multiple clients
	Simple TCP broadcasting	TCP server autodiscovery is supported
	For Windows 7, 8 and 10 (.NET 4.5 required)	Please note that WinRT is not supported
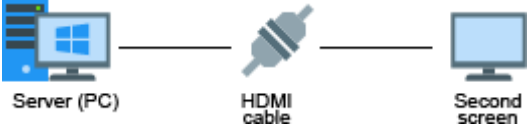
3. Setting up

3.1. Single computer setup

If you have just one room you can set up using just one Windows computer.

Single computer setup	Details
Server	Use one computer as both server and client
Client (screens)	Use a second computerscreen, place it in a room and, connect with a long HDMI cable. Use a MiraCast solution which can replicate a Windows desktop on the screen. For example: MiraCast is just a wireless HDMI cable. This makes a regular TV monitor behave as an actual second screen of your server computer.
Drawbacks	No or limited touchscreen possibilities. Multiple rooms on different screens of one computer is tricky. You could get it to work using one computer, but you'll probably end up with multiple server-side computers due to audio restraints.

Common examples of single computer setups:

Description	Requirements	Example
PC, classic	• 1 PC • Second screen • 1 HDMI cable	 Server (PC) — HDMI cable — Second screen
PC, wireless MiraCast	• 1 PC (MiraCast enabled) • Second screen • 1 MiraCast adapter	 Server (PC) (Miracast enabled) — Wireless MiraCast — MiraCast adapter — Second screen
Tablet, wireless MiraCast	• 1 tablet (MiraCast enabled) • Second screen • 1 MiraCast adapter	 Server (tablet) (MiraCast enabled) — Wireless MiraCast — MiraCast adapter — Second screen

3.2. Multiple computer setup

If you have multiple rooms, please consider using multiple Windows computers or Windows tablets.

Multiple computer setup	Details
Server	Use one server computer to host multiple room clients.
Client (screens)	Use actual computers in the rooms as clients. Usually you want tablets for this because they are small, easy to attach, sleek and have touchscreens. They must be tablets with Windows 10. An alternative is to hide a computer in the room where it is not accessible to the players and place the screen somewhere near it. Please note that a network connection (cabled or wifi) is required.
Drawbacks	Multiple computers needed.

Common examples of multiple computer setups:

Description	Requirements	Example
2 PC's, classic	• 2 PC's • Cabled network access	Server (PC) — UTP cable (wired network) — Client (PC) (hide in the room)
2 PC's, wireless	• 2 PC's • (stable) Wi-Fi access	Server (PC) — Wi-Fi — Client (PC) (hide in the room)
1 PC, 1 tablet, wireless	• 1 PC • 1 Windows tablet • (stable) Wi-Fi access	Server (PC) — Wi-Fi — Client (Windows tablet)
2 tablets, wireless	• 2 Windows tablets • (stable) Wi-Fi access	Server (Windows tablet) — Wi-Fi — Client (Windows tablet)

Example of a setup with multiple rooms:

Description	Requirements	Example
Multiple rooms example with 3 rooms	• 1 PC • (stable) Wi-Fi access • 3 Windows tablets	
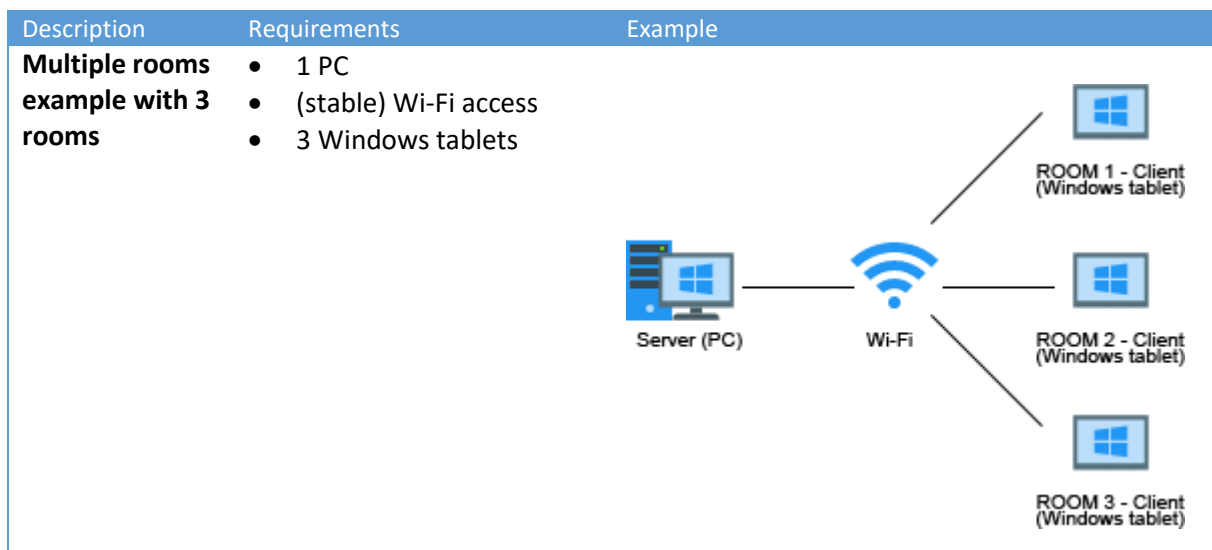
3.3. Hardware requirements

Computer requirements (for any computer used in your solution)

- Windows tablet/PC
- Windows 10 or higher is recommended
- .NET 4.8 must be installed
- Stable network access (cabled or WiFi)
- Firewall access to the configured TCP or UDP ports (you should allow access when the Windows firewall access screen pops up)
- Touchscreen is optional
- Speakers are optional
- MiraCast adapter is optional

3.4. Recommended update strategy

The key words are *"predictable outcome"*. Set up a test environment. A computer where you have the current version installed, as well as your room.

When a new version is released, download and install the update in the test environment.

In this way you will be able to check out the changes without affecting and interfering with your production (live) environment.

4. Getting started

4.1. Step 1

Run the main menu and create a new room by pressing the EDIT ROOM button.
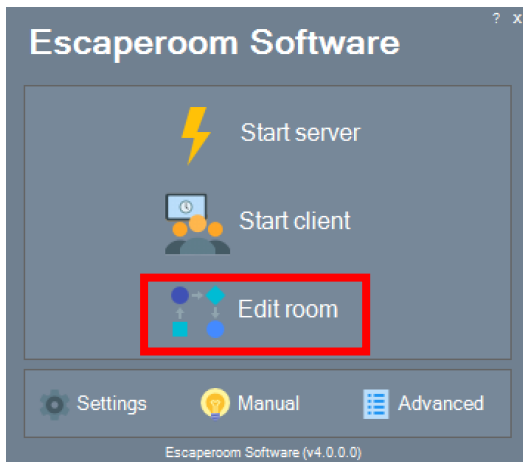
4.2. Step 2

Use the Wizard to easily configure a new and simple room.
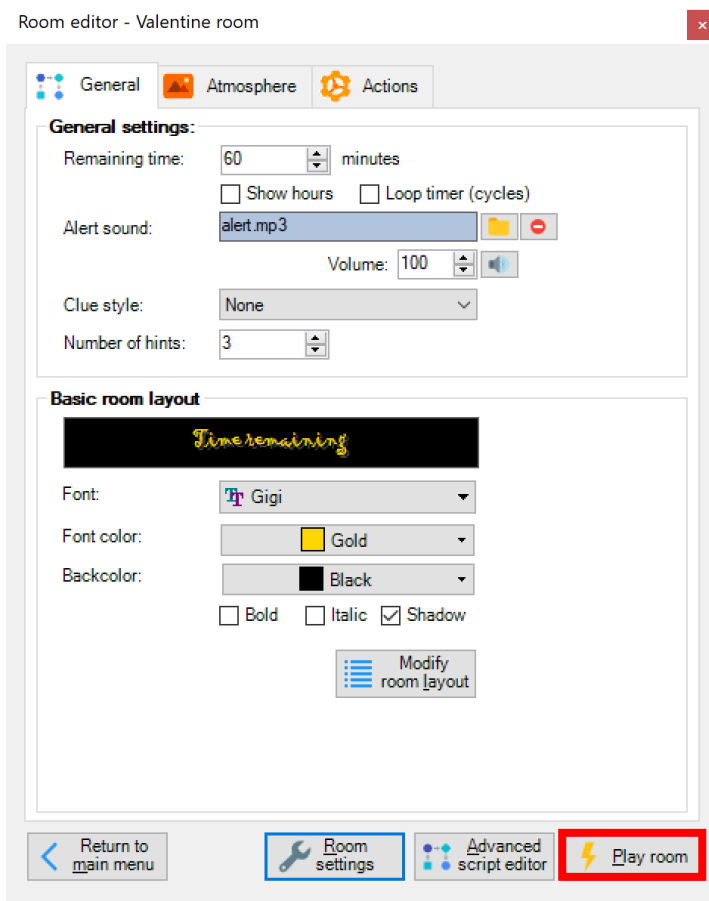
4.3. Step 3

Press the Play room button to test your room.

4.4. Step 4

Start your newly created room using the START button.
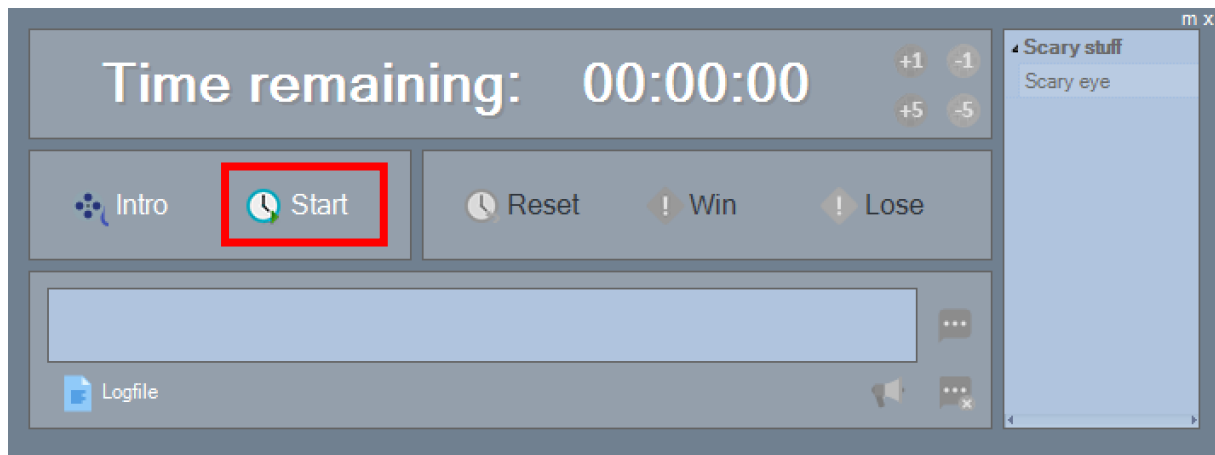
4.5. Step 5

Enjoy your newly created room!
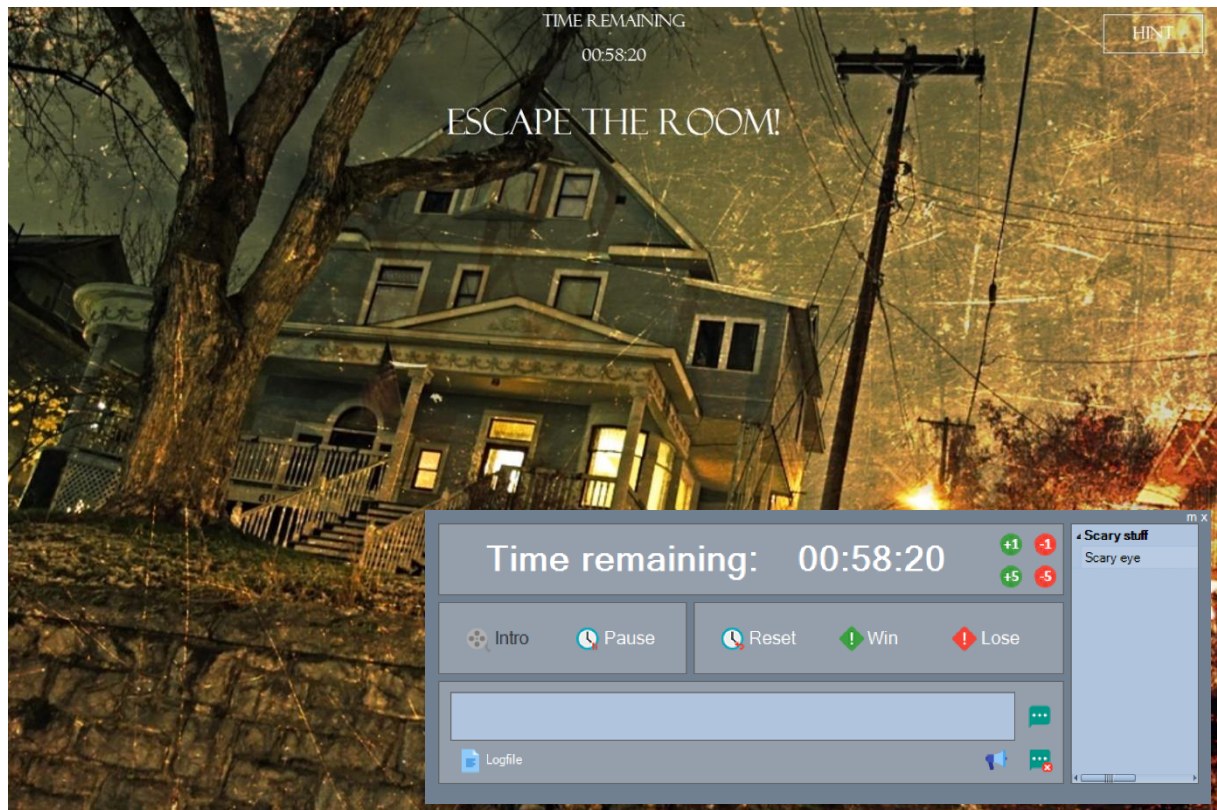
Please check the rest of this manual for more instructions and possibilities!

5. Screens overview

5.1. Start screen
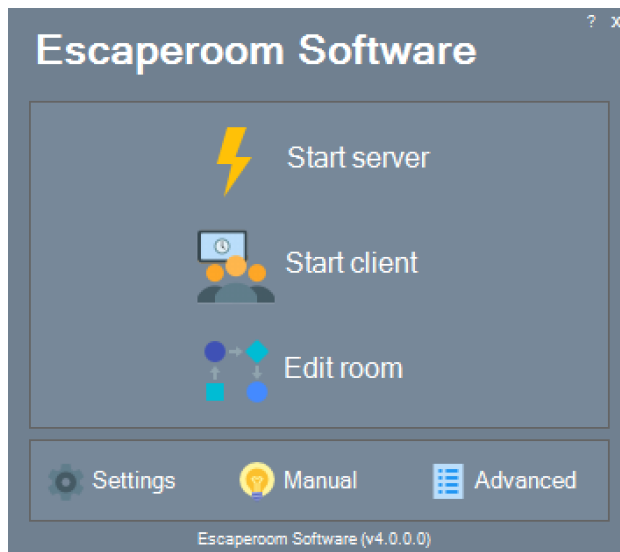
Start screen	Details
Start server	Start the selected room in SERVER MODE. Please note that audio for SERVER MODE is muted per default.
Start client	Start the selected room in CLIENT MODE. Use F12 to close CLIENT MODE.
Edit room	Start the selected room in EDITOR MODE. Please note that EDIT MODE runs the room in it's configured resolution.
Settings	Go to the settings screen.
Manual	Shows the PDF help file. You can also press the F1 key.
Advanced	Open the advanced screen when using multiple rooms. You can change the settings to automatically start in multi-room mode.
Version & license info	Shows version number and licenses in use. Click to visit our website to purchase licenses or check for updates.
Update available notification	If a new version is available a notification will be visible. Click the notification to download and install the latest version. You can turn this notification of in the settings.

5.2. Settings

5.2.1. General

General settings	Details
Always start in advanced mode (multi-room)	When handling multiple rooms from one computer, you probably want to always start in multi-room mode.
Notify when new updates are available	Shows a notification when updates are available on the start screen and in multi-room mode. Turn this off when you don't have an internet connection available, or do not wish that one of your employees accidentally installs an update at an inconvenient time.
Debug mode (log all network messages to logfile)	Turn this on for more detailed logging and insight in what messages are being sent and in what order. Press F10 in editor, server or client modes.
Disable playing videos in server preview (improves performance)	When running both server and client on 1 PC, videos are usually played on both client and server. This can cause performance issues if the video quality is high. This feature skips playing of videos on the server side.

Directory settings	Details
Working directory	This is the folder where active room files, scripts and resources are unzipped. The **programdata** folder is configured per default.
Import/export directory	The folder where you collect your room ZIP files. You could choose to make this a shared folder or a dropbox folder to easily transfer room ZIP files to your clients.

Support	Details
Save debug files	This button exports your settings, room scripts and logfiles. Send the resulting .ZIP file to support@escaperoombuilder.com. This button also turns on 'debug mode'.

5.2.2. Advanced

E-mail settings	Details
SMTP host, port	Optionally configure an SMTP server. These settings enable the e-mail functionality of the scripting engine.
Username, password	Enter your credentials if your SMTP account requires authentication.
From address	Used as from and reply-to address in any e-mails sent.
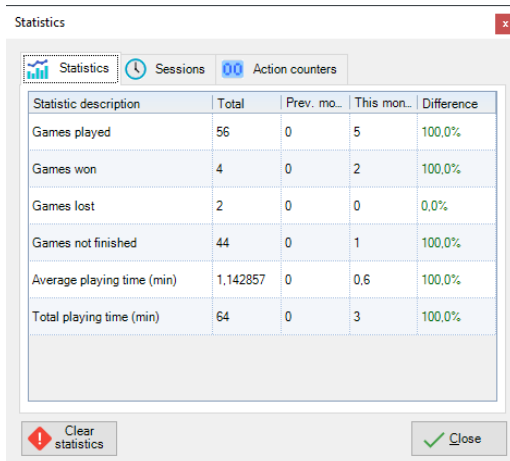
5.2.3. License

License	Details
License key	Enter your license key here.
Activate	Activate your software on-line. Please note that you only have to activate your SERVER installation. Your CLIENT installation does NOT need any activation.
Offline activation	If you don't have an internet connection available, please use manual activation to activate your software. You can activate manually by using the e-mail button and send the challenge code to info@escaperoombuilder.com in order to receive your activation code. If your application is not activated, a 15-minute usage time limit is enforced.

5.3. Statistics

5.3.1. General statistics

Statistics	Details
Statistic description	Describes the statistic shown in that line.
Total	Shows the grand total of KPI results.
Previous month	Shows the KPI results for the previous month.
This month	Shows the KPI results for this month.
Difference	A percentage, comparing this month with the previous month.
Clear statistics	Reset the statistics table

5.3.2. Sessions

Sessions	Details
Room	Name of the room.
Last modified	When the room was last modified.
Started	Game session start timestamp.
Stopped	Game session stop timestamp.
Minuteds elapsed	Game time in minutes
Game result	Win, Lose, InstaLose (=triggered loss), Unfinished
Custom 1-9	Using scripts, you can store variable values into custom fields 1-9 (see `SaveVariableToSessionStats`)
Export to CSV	Export to Comma Separated Values (CSV) file; this type of file can be imported into a Spreadsheet tool.

5.3.3. Action counters

This screen shows how often certain actions or clues were used during the game sessions.

Please note that you must **turn on** *Track Statistics* in an Action to track it in this overview.

Counters	Details
Room	Name of the room.
Date	Game session date.
Result	Game session result.
Counter	The action being counted.
Value	The num
Export to CSV	Export to Comma Separated Values (CSV) file; this type of file can be imported into a Spreadsheet tool.

5.4. Multi-room mode

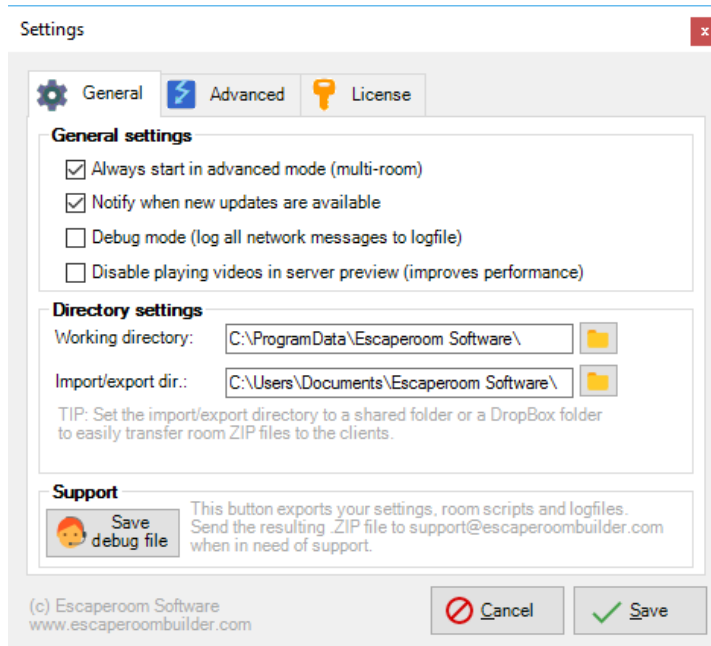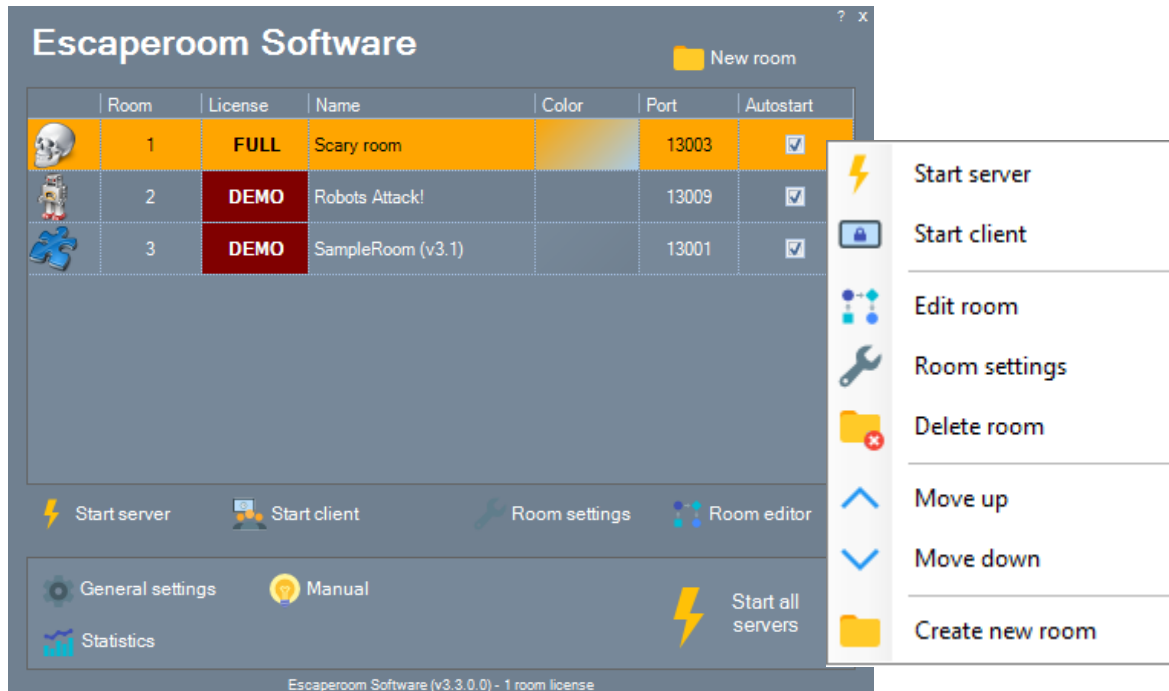| Main menu	Details
New room	Create a new room.
Selection grid	The selection grid shows all your rooms. Doubleclick to open the ROOM SETTINGS.
Image	A nice image to indicate the type of room.
Room	Shows the room slot number. You have licenses for a limited number of slots.
License	Shows whether a room is LICENSED or in DEMO mode.
Name	The room name.
Color	Shows the configured game master panel color for this room.
Port	Shows the configured port used for this room.
Autostart	Indicates that the room will be automatically started when pressing the START ALL SERVERS button.
Start server	Start the selected room in SERVER MODE. Please note that audio for SERVER MODE is muted per default.
Start client	Start the selected room in CLIENT MODE. Use F12 to close CLIENT MODE.
Room settings	Display the room settings dialog.
Edit room	Start the selected room in EDITOR MODE. Please note that EDIT MODE runs the room in it's configured resolution.
Move up/down	Move a room up or down a slot.
General settings	Go to the general settings screen.
Statistics	Go to the statistics screen.
Manual	Shows the PDF help file. You can also press the F1 key.
Start all servers	Starts all rooms that have "Autostart" checked.
Version & license info	Shows version number and licenses in use. Click to visit our website to purchase licenses or check for updates.
Update available notification	If a new version is available a notification will be visible. Click the notification to download and install the latest version. You can turn this notification off in the settings.

19

5.5. Room settings screen

5.5.1. General

General settings	Details
Room name	The room name.
Menu image	A nice image to indicate the type of room.
Import room	Import a room .ZIP file for the current selected slot.
Export room	Export a room .ZIP file based on the current selected slot.

Other settings	Details
Run when pressing "Start all servers" button	Indicates that the room will be automatically started when pressing the START ALL SERVERS button.
Autostart room timer on server start	Automatically start room (timer) when starting this room in server-mode.
Sound alert after each manual message sent	Whether or not an alert sound/flash should be sent after the game master sends a message to the room.
Snapshot save interval n seconds	Save a snapshot for failsafe recovery every n seconds. If a game session is ended improperly, on restart of that room you will be asked whether to continue based on the last saved snapshot. 0 = No snapshots 60 = Recommended setting

5.5.2. Export room
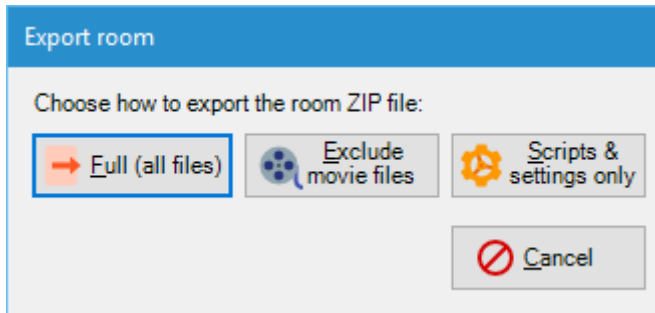
Export room button	Details
Full (all files)	Export a room ZIP file including all files.
Exclude movie files	Export a room ZIP file excluding all video files (possibly skipping gigabytes of movie files)
Scripts & settings only	Export a bare room ZIP file, only containing scripts and room settings. (smallest)

5.5.3. Connections

Main connection	Details
Connection (TCP)	Indicates the protocol that is used for connection. Please note that UDP connection is experimental and is no longer supported for all functions.
Server IP address	Leave the server IP field blank for clients to automatically detect the server IP. For firewall purposes, please note that this feature uses a TCP/UDP combination to detect the server IP.
Port	Please note that you must configure a different port *for each different escape room* in order for the right message to reach the right room clients.

Other connections	Details
Enable self-hosted webservice API on server (you must 'run as Administrator')	Turn on self-hosted API on a port. You must run the Server 'as Administrator' to enable access.
Enable home automation integration (Z-Wave)	Turn on and configure a Z-Wave COM port to enable home automation functions.

External controllers	Details
Add controller	Add a new external controller. Allows you to configure interaction with external controllers (Arduino, Raspberry PI).
Remove controller	Remove an existing external controller.

5.5.4. Self-Hosted API

Turn on self-hosted API on a port. You must run the Server 'as Administrator' to enable access.

The Server will show how to access local and LAN. You can leave roomID empty for 1 room or when notifying all rooms. The actionGUID can be found on the action dialog.
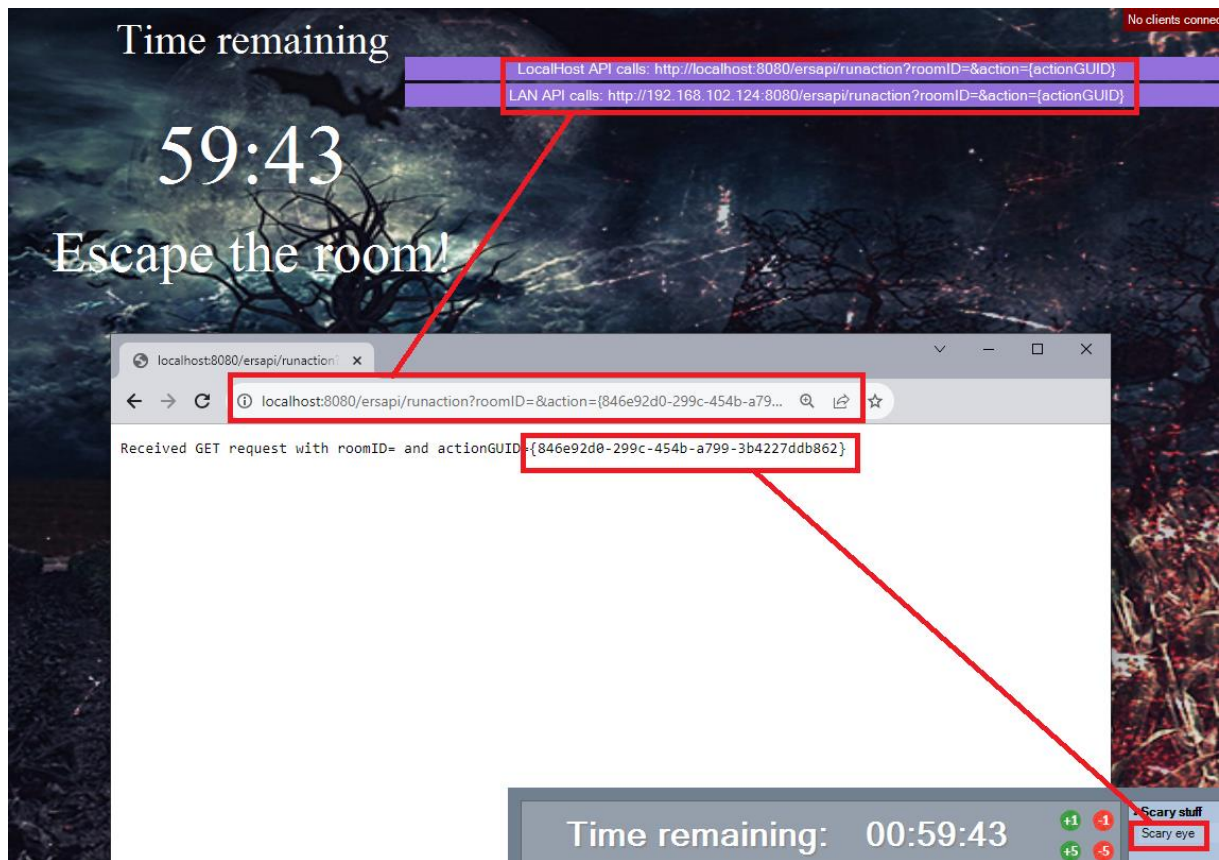
Local machine API call example

http://localhost:8080/ersapi/runaction?roomID=&action={846e92d0-299c-454b-a799-3b4227ddb862}

LAN API call example

http://192.168.100.2:8080/ersapi/runaction?roomID=&action={846e92d0-299c-454b-a799-3b4227ddb862}

*Please check **chapter 9** for examples on how to connect with **microcontrollers**.*

5.5.5. Home Automation Integration with Z-Wave

Using a Z-Wave Stick in a server USB-port you can control paired Z-Wave components.
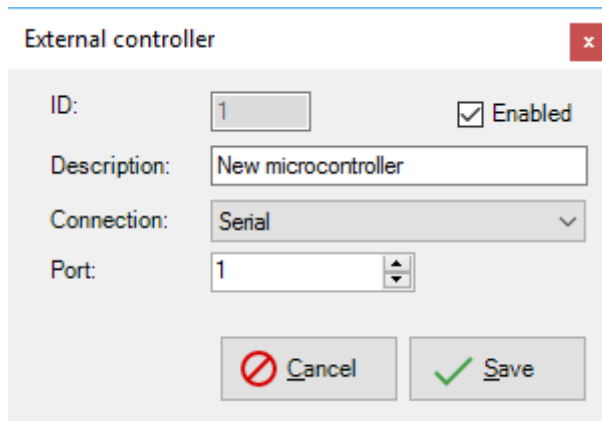
Supported components

- Smart plugs
- Door/window sensors

Troubleshooting

- Make sure the Z-Wave Stick is in a *powered* USB port
- Make sure the Z-Wave Stick COM port number is correct (this may change when switching USB ports)
- If Z-Wave devices won't connect to the Z-Wave Stick, put the Z-Wave Stick in "un-pair"-mode and keep it near the troublesome device. This should "un-pair" the device, after which you should be able to normally pair it to the Z-Wave Stick again
- Make sure the proper Z-Wave device id is chosen from the dropdown in the Escaperoom Software actions dialog

5.5.6. External controllers

External controller	Details
ID	Use the ID to assign specific actions to an external controller in the wizard or in the script editor.
Description	Describe the task or location of this external controller for future reference.
Enabled	Enable or disable an external controller.
Connection	Connection types TCP and Serial are supported. Please check the included demo projects for code samples and Escape Room Software rooms.
Port	Enter the port number for the external connection.

5.5.7. Server screen

Server screen settings	Details
Screen number	Select a screen to show SERVER MODE on when working with multiple screens. Use the value 0 to not specify a specific screen. After changing the screen, please reselect the screen location to update X, Y locations and sizes.
Identify screens	Press the magnifying glass button to identify the screen numbers (they show large red in the center of the respective screens)
Screen location	Use the screen location tool to quickly set a location and size to start when in SERVER MODE. Use 4 or 6 room location presets when hosting multiple rooms on the same server computer.
Custom location (location X, Y, width, height)	You can also manually set the location and size fields if needed using actual coordinates and sizes.
Game master panel color	Shows the configured game master panel color for this room.

This documentation is associated with Escape Room Software version v5.0.0.0
© Escape Room Sofware 2016-2024, www.escaperoombuilder.com

5.5.8. Client screen

Client screen settings	Details
Start client on same computer as server	When checked, a CLIENT MODE for this room is also started when starting SERVER MODE. Use this when you have 1 computer to run both the CLIENT and SERVER on.
Start when Windows starts	Toggle whether Windows automatically starts this room in CLIENT MODE when Windows starts. Use this on dedicated CLIENT computers. Please note that administrator rights are required to use this function.
Screen number	Select a screen to show CLIENT MODE on when working with multiple screens. Use the value 0 to not specify a specific screen.
Resolution	Select the resolution for the CLIENT. Keep the client resolution in mind when using advanced functions to place buttons or move text labels. TIP: Set the client resolution for more accurate display results. Please keep the client resolution in mind when using advanced scripting to place buttons or text labels.
Hide cursor on client screen	Hides the cursor on the client screen, making it more suitable for touchscreen (default this is turned ON).
Force focus on client window	Turn on if other software on a dedicated client computer causes the client application running there to lose focus (causing client keyboard or barcode input to fail). This feature checks and restores focus if needed every 2 seconds (default this is turned OFF).

5.5.9. Languages
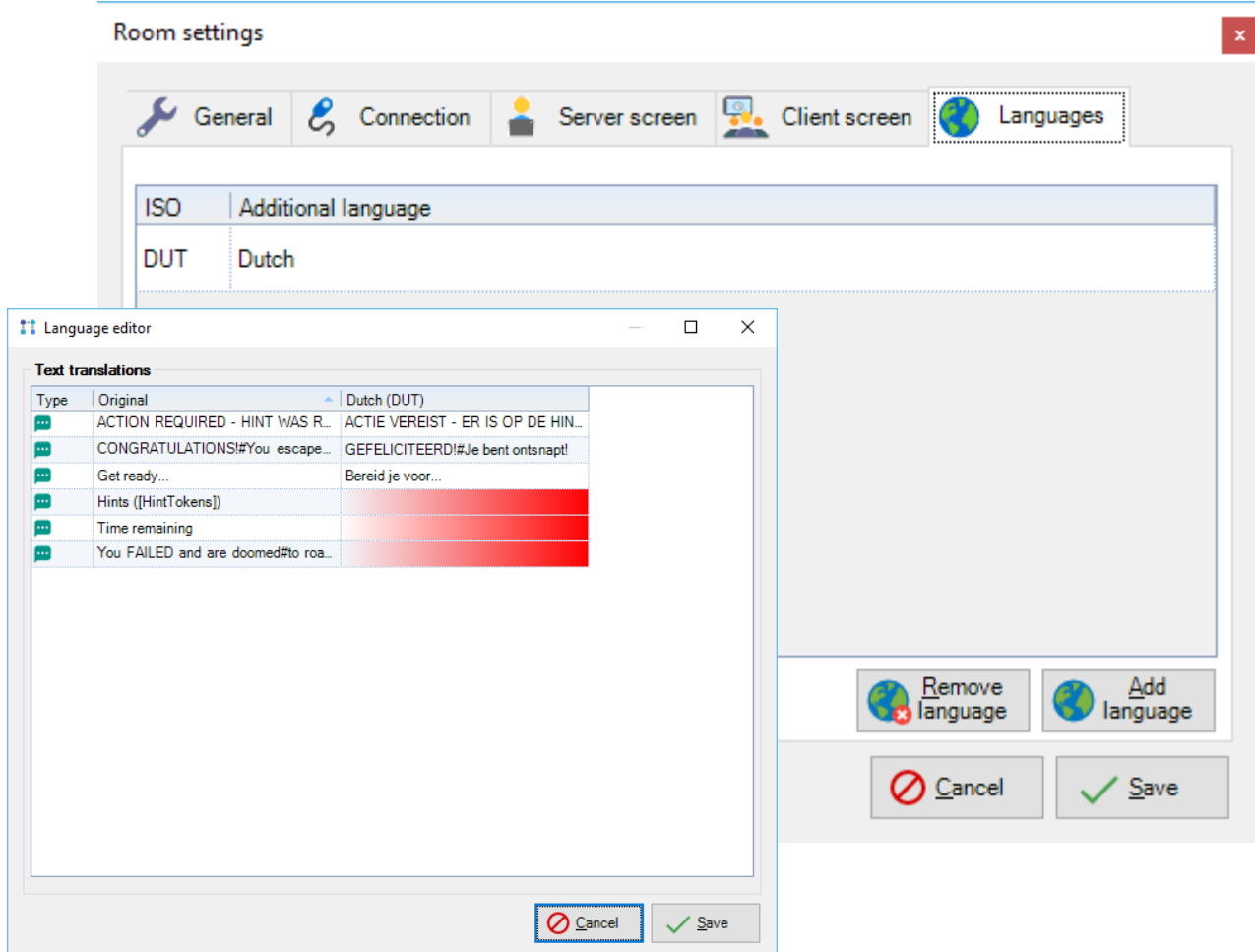
Language settings	Details
Add language	Add an additional supported language, on top of the usual language.
Remove language	Remove a language.

Language editor	Details
Grid	The grid shows the original language and the alternate language of all text in the room file. Please note that untranslated lines will show red and will not be translated when running a game session.

5.6. Game master panel
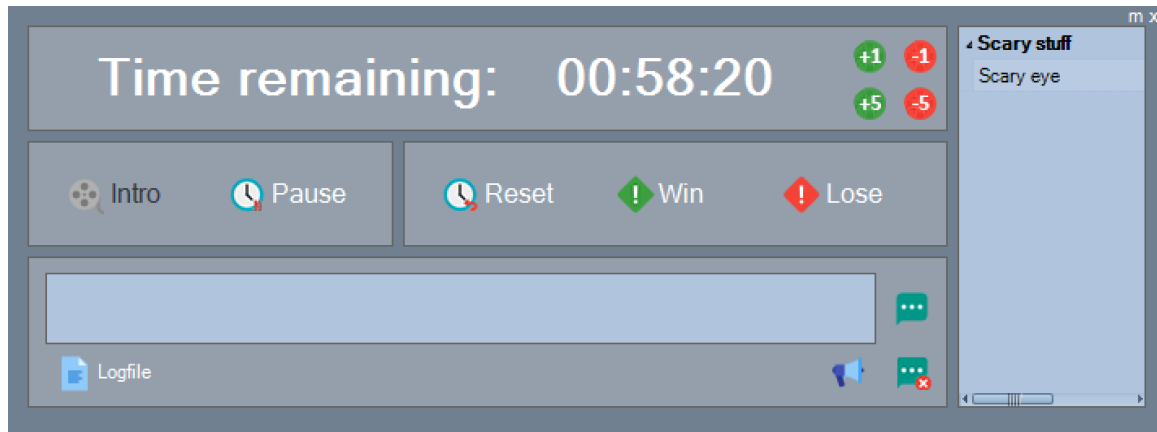
Game master panel	Details
Time remaining	Shows remaining time. Turns red when <10 minutes. When the timer expires, the room is automatically lost and runs the LOST script.
Time +/- buttons	Increase or decrease the remaining time with 1 or 5 minutes. An alert will be sent to client on each change.
m-button (red or white)	Indicates whether the room sound is muted. Click to toggle audio.

Game panel	Details
Intro	Play intro. Runs the INTRO script.
Start / Pause	Start or Pause the room. Runs START script on initial start.
Reset	Reset the room.
Win	Instantly win the room. Runs the WIN script.
Lose	Lose the room. Runs the INSTALOSE script which kicks off the normal lose (time expired) script by default.

Message panel	Details
Message / send	Sends text message to the client.
Clear message	Clears all messages and images on the client.
Alert tone	Sends an ALERT to the client.
Logfile	Shows the logfile window.

Clues panel	Details
Clues list	Shows all manual clue actions available fort his room grouped per category. Doubleclick to activate a manual clue action.
	Useful for prepared clues and scary effects. The number of times you used specific clues is indicated, some clues may show as "stiked through" to indicate progress.

5.6.1. Logfile window

```
Logfile: C:\Users\Gijsbert\AppData\Local\Temp\Escaperoom Software\Es...    —    □    ✕

17-7-2018 12:12:04 63 > SENT > umtRemoveButton, ClueYes
17-7-2018 12:12:04 63 > RECV > umtRemoveButton, ClueYes
17-7-2018 12:12:04 64 > SENT > umtRemoveButton, ClueNo
17-7-2018 12:12:04 64 > RECV > umtRemoveButton, ClueNo
17-7-2018 12:12:04 65 > SENT > umtAddButton, Clue; Clue; ; 10; 10; 175; 100;
17-7-2018 12:12:04 65 > RECV > umtAddButton, Clue; Clue; ; 10; 10; 175; 100;
17-7-2018 12:12:04 66 > SENT > umtAddKeyCode, 1234; {df686281-26ce-46df-a890
17-7-2018 12:12:04 66 > RECV > umtAddKeyCode, 1234; {df686281-26ce-46df-a890
17-7-2018 12:12:04 67 > SENT > umtAutomaticTimerChange, 3599;1
17-7-2018 12:12:04 67 > RECV > umtAutomaticTimerChange, 3599;1
17-7-2018 12:12:04 68 > SENT > umtAutomaticTimerChange, 3598;2
17-7-2018 12:12:04 68 > RECV > umtAutomaticTimerChange, 3598;2
17-7-2018 12:12:05 69 > SENT > umtAutomaticTimerChange, 3597;3
17-7-2018 12:12:05 69 > RECV > umtAutomaticTimerChange, 3597;3
17-7-2018 12:12:06 70 > SENT > umtAutomaticTimerChange, 3596;4
17-7-2018 12:12:06 70 > RECV > umtAutomaticTimerChange, 3596;4
17-7-2018 12:12:07 71 > SENT > umtAutomaticTimerChange, 3595;5
17-7-2018 12:12:07 71 > RECV > umtAutomaticTimerChange, 3595;5
```

Clear log   ☐ Pause logging refresh   Browse logfiles   Manual refresh

Name	Details
Log	Displays the log results. Logfiles are saved in the Windows temporary folder. Every day a new logfile is saved. Errors will be shown in RED. Home automation logs will be shown in BLUE. Microcontroller logs will be shown in PURPLE.
Manual refresh	Press the refresh button to see the most recent information.
Browse logfiles	Opens the Windows file explorer in the temporary logfile folder.
Pause logging refresh	Pause logging to be able to scroll through the log manually.
Clear log	While testing your room, you may want to clear Today's logfile now and then to avoid scrolling for recent info.

5.7. New room wizard dialog

5.7.1. General
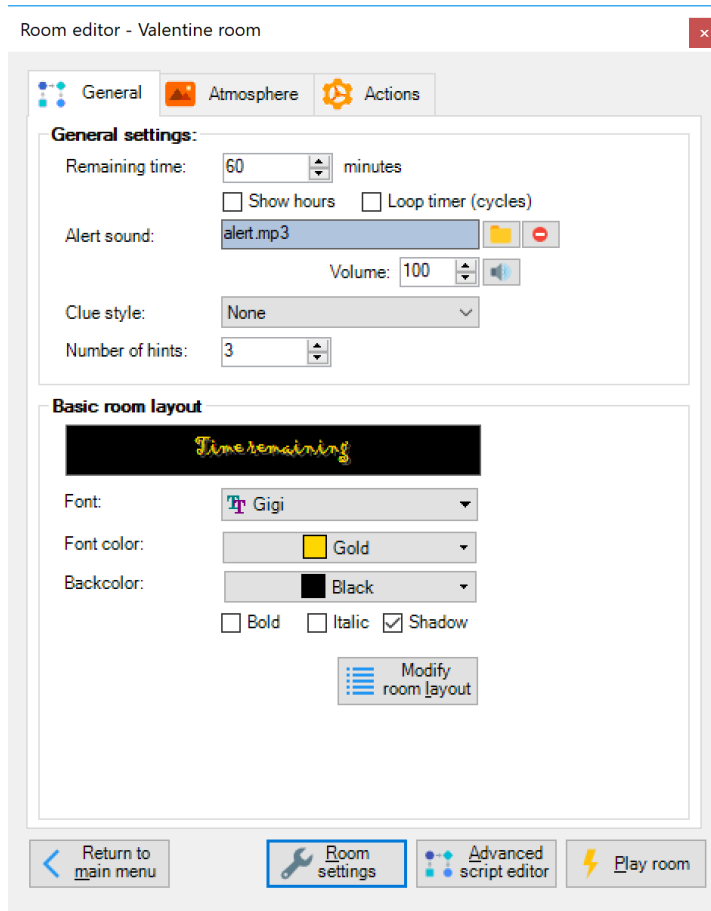
General settings	Details
Remaining time	Configure the initial remaining time for this room.
Show hours	Whether or not to show hours on the timer (default OFF)
Loop timer (cycles)	When timer finishes, instead of losing the game, the timer will restart. This enables you to create cycles.
Alert sound, volume	Configure a specific alert sound for this room.
Clue style	Select a style of hint button (counter, button or none)
Number of clues	Max. number of hints when using a counter hint button.

Basic room layout	Details
Font	Select a main font for all labels.
Font color	Select a font color for all labels
Backcolor	Select a backcolor for the client screen
Bold, italic, shadow	Configure additional font properties
Preview room layout	Preview the game screen and set locations and sizes for labels. You can also override colors and font styles.

5.7.2. Modify room layout
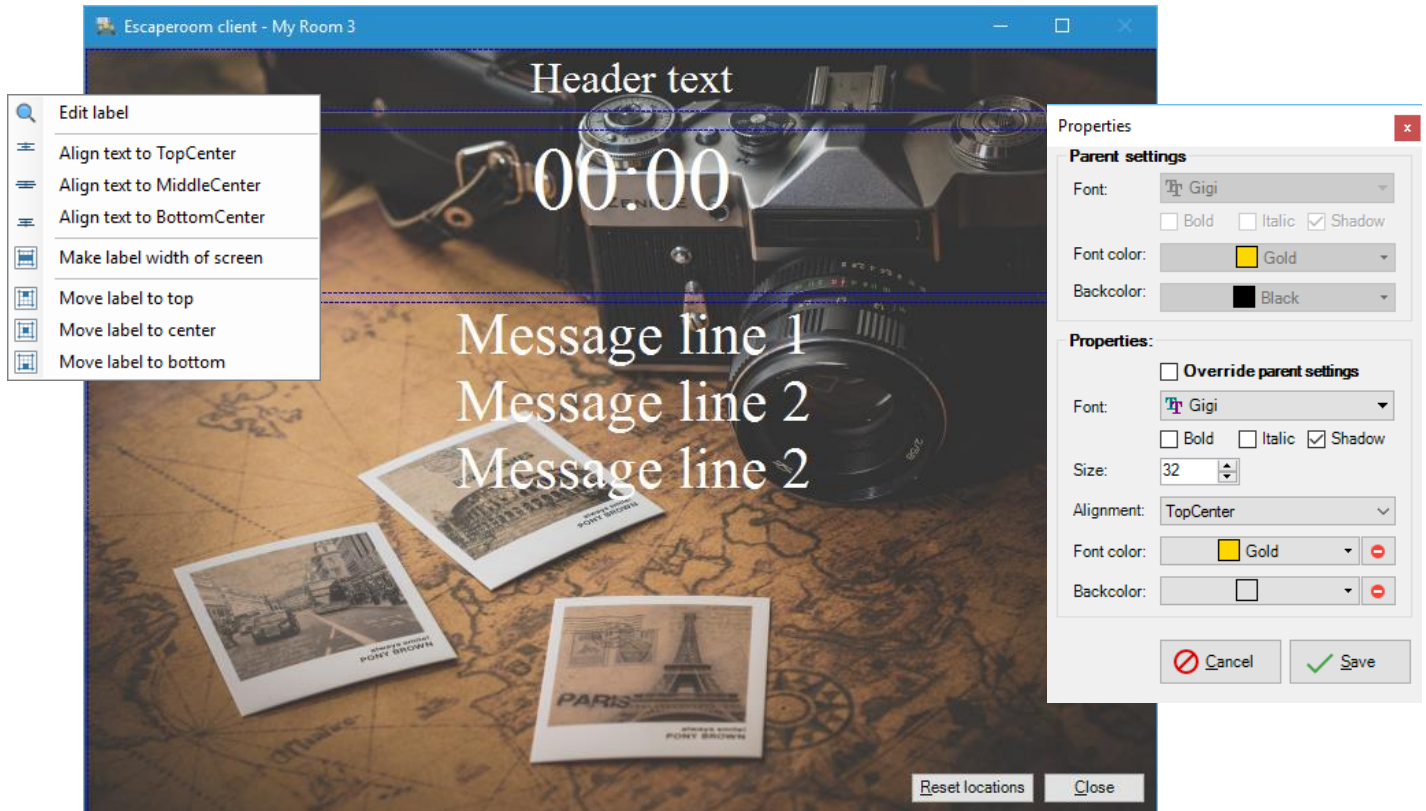
Basic room layout dialog	Details
Header label	This label shows a header caption during the game.
Timer label	This label shows the timer during the game.
Message label	This label shows messages during the game.
Reset locations	Reset the label locations on the screen to default.
Close	Save and close the basic room layout dialog.

Action	Mouse action	Keyboard action
Move label	Drag & drop	Arrow keys
Resize label	Resize using border edges	Increase width: **Home**, decrease width: **End** Increase height: **PgUp**, decrease height: **PgDown**

Context menu	Details
Label properties	Open the label properties dialog
Align label to…	Set the textalignment of the selected label.
Width of screen	Make the label the width of the entire screen.
Move label to…	Move the label to a preset location of the screen.

Properties dialog	Details
Parent settings	Every label adheres to the font settings set in the wizard.
Override parent settings	Enable to override the wizard font settings with custom settings for this label.
Font	Set the font family of the selected label.
Size	Set the fontsize of the selected label.
Alignment	Set the textalignment of the selected label.
Colors	Set the front and back colors of the selected label.

5.7.3. Atmosphere
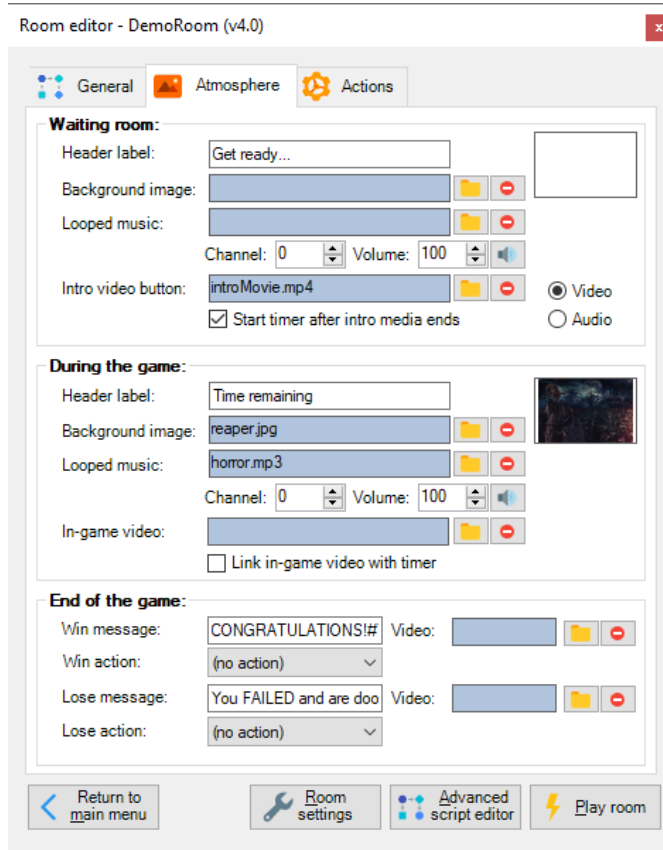
Waiting room	Details
Header label	Set the text of the header label before the game.
Background image	Set the background image before the game.
Looped music, volume, channel	Import audio to play looped in the waiting room. Channel 0 means NO CHANNEL and always play on a separate channel.
Intro video button	Import a video to play when pressing the Intro button.
Start timer after media ends	Automatically start the timer after the intro audio or video ends.

During the game	Details
Header label	Set the text of the header label during the game.
Background image	Set the background image before the game.
Looped music, volume, channel	Import audio to play looped during the game.
In-game video	Import a video to play during the game.
Link in-game video with room timer	Make the video jump along with manual or scripted timer manipulations.

End of the game	Details
Win message	Show a message when the game is won.
Win video	Import a video to play when the game is won.
Win action	Trigger an action upon Win condition.
Lose message	Show a message when the game is lost.
Lose video	Import a video to play when the game is lost.
Lose action	Trigger an action upon Lose condition.

5.7.4. Actions
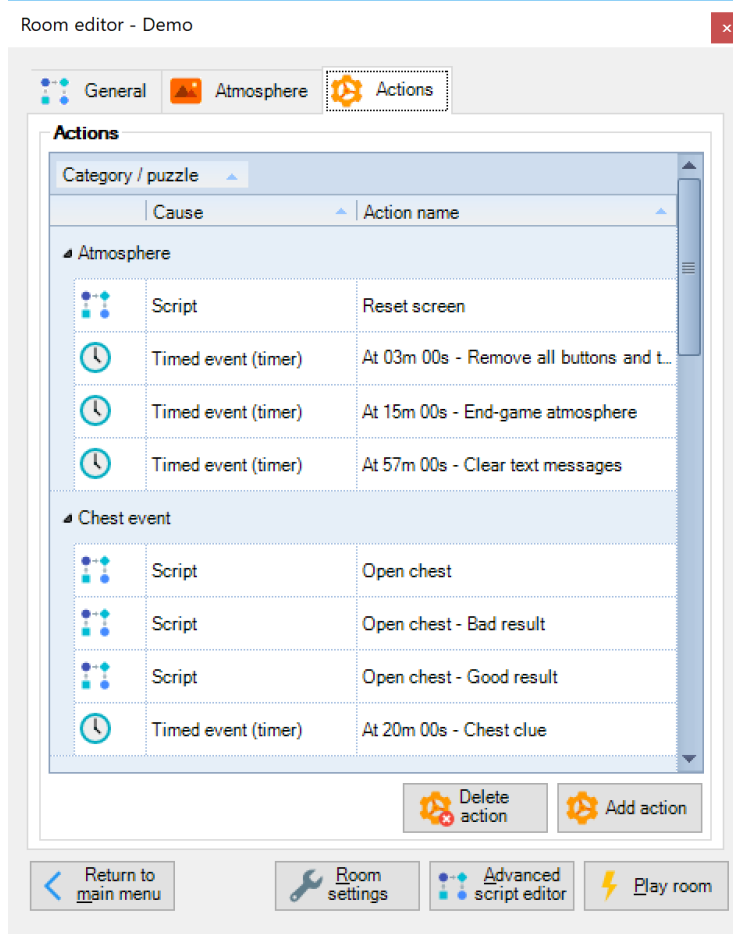
Actions	Details
Actions grid and buttons	Manage various types of actions for this room.

5.8. Action dialog

5.8.1. Cause

General	Details
Action name	Name of the action will also be used as script filename. Example: "Puzzle #1 – Clue #2"
Category/puzzle	Divide your actions into categories. Clues are displayed in categories in the Game Master panel.
GUID	The action GUID is an identifier that can be used for scripting and incoming API calls

Cause	Details
None (stand-alone action, triggered by other actions)	This action may be called by other actions, or external actions via TCP connection.
Clue (manually click on game master panel)	Action is shown as a clues on the game master panel during the game. It must be triggered manually by the game master.
Timed, when timer reaches (min., sec.)	Action is triggered at a specific time in the game.
Timed, based on PC time schedule	Action is triggered at a specific time on the PC clock.
Keypad entry	Action is triggered when a valid keycode has been entered.
Home automation sensor	Action is triggered when a home automation sensor is triggered.

5.8.2. Timed, based on PC time schedule

Timed, based on PC time schedule (optional)	Details
One time	Trigger this action only one time on a specific start date.
Date range	Trigger this action every day in the startdate/enddate daterange, you can also configure daily recurrence.
Monthly	Trigger this action monthly, according to daily recurrence at the proposed start time.
Start time	The start time of the action.
Repeat every hour until	Repeat this action every hour at the same minute as the start time for the rest of the day.

5.8.3. Keypad entry

Keypad settings (optional)	Details
Keypad code	A keypad code that triggers the current action. Please note that you can use an actual keyboard, keypad or barcode scanner to trigger an action like this.
Remove after correct entry	Whether or not a keypad code must be removed after correct entry.
Wrong entry action	Select an action that must be run when a wrong keycode is entered.

5.8.4. Effect

Action	Details
Clear screen	*No action*, *Clear all* or subcomponents on the screen
Background image	The shown image is used as background image.
Stop audio & video	
Play audio file, volume, channel	Play an audio file. Supported filetypes (.mp3, .wav)
Looped music	Stops earlier looped music and starts new looped music.
Font, weight, color	Change font, weight and colors of the main text.
Header text	Set header text.
Message text	Display message text.
Show image	Show an image. Supported filetypes (.jpg, .jpeg, .png, .bmp).
Full-screen image	The shown image is full-screen, overlapping text and timer.
Flickering image, flickers	Make the image flicker a number of times before disappearing.
Play video file	Play a video file. Supported filetypes (.avi, .mp4, .mov)

External messages	Details
GM message	Show a warning message to the Game Master (GM) ony, with an accompanying ALERT sound. Use # as return character.
Print file, target	Print a PDF or Microsoft Word® file on the default printer of the client computer, the server computer or both.

E-mail message	Send an e-mail message with a specific subject to a target e-mail address. Please note that an SMTP server must be configured in the general settings.
Send API call	Runs API call on the SERVER computer. Use ? for params, & as param splitter and %20 for spaces. Example: *http://192.168.100.2:8080/someapi/runaction?param1=0&param2=test%20me* You can do API calls on the Escaperoom Software self-hosted API. Be careful that the API call does not call itself, this will result in an endless loop and freeze the application.
External device message	Send a plain string message to one specific or all configured external devices (TCP, and serial connections are supported). You can use variable names using [brackets]
Home automation action	Trigger a home automation switch.

Special actions	Details
Change timer	Add/remove a number of minutes. 0=no action.
Jump to	Jump to a specific point in time. 0=no action.
Decrease hint counter	Decrease the hint counter variable by 1.
Strike through clue(s)	Strike through this clue and all earlier strikethrough-enabled clues in this category/puzzle (earlier according to alphanumeric sortorder) on the Game Master panel. (only applies if clue is shown on the Game Master panel)
Strike through clue	This clue will be stricken through in the clue selection list.
Statistics: Track this script	A statistics counter keeps track of every time this script was run. See the chapter on *Statistics* in this manual for additional information.
Endgame actions	Set the game status to WIN GAME or LOSE GAME (runs the InstaLose script)

After action	Details
Follow-up after	Follow-up with another action (if configured) after a number of seconds.
Follow-up action	Assign a follow-up action to this action.

5.8.5. Advanced

Advanced	Details
Automatically generate action script (default=ON)	Toggle whether this action script is automatically generated. You can disable this check to manually modify the script. Please note that wizard functions for this script will be disabled when doing so.
Script editor	Shows the current selected script from the script files tree. When making a modification, a * is shown begind the script file in the script files tree. Functions are colored **blue**, strings are colored **red** for easier working. {**purple** text are GUIDs, which are unique identifier codes for other scripts. Hover above a GUID to see the actual name of the script that is referred to. Use the right mousebutton to insert a GUID for any existing script} //Remarked tex is shown **green** and execution of these lines is skipped :autogen: code is **gray**. It is code autogenerated by the wizard and may be overwritten when revisiting wizard mode.
Function glossary tree	Displays all functions. Doubleclick to add a function tot he script editor.
Function glossary info panel	Describes the syntax and details of the function selected in the function glossary tree.

5.9. Advanced script editor
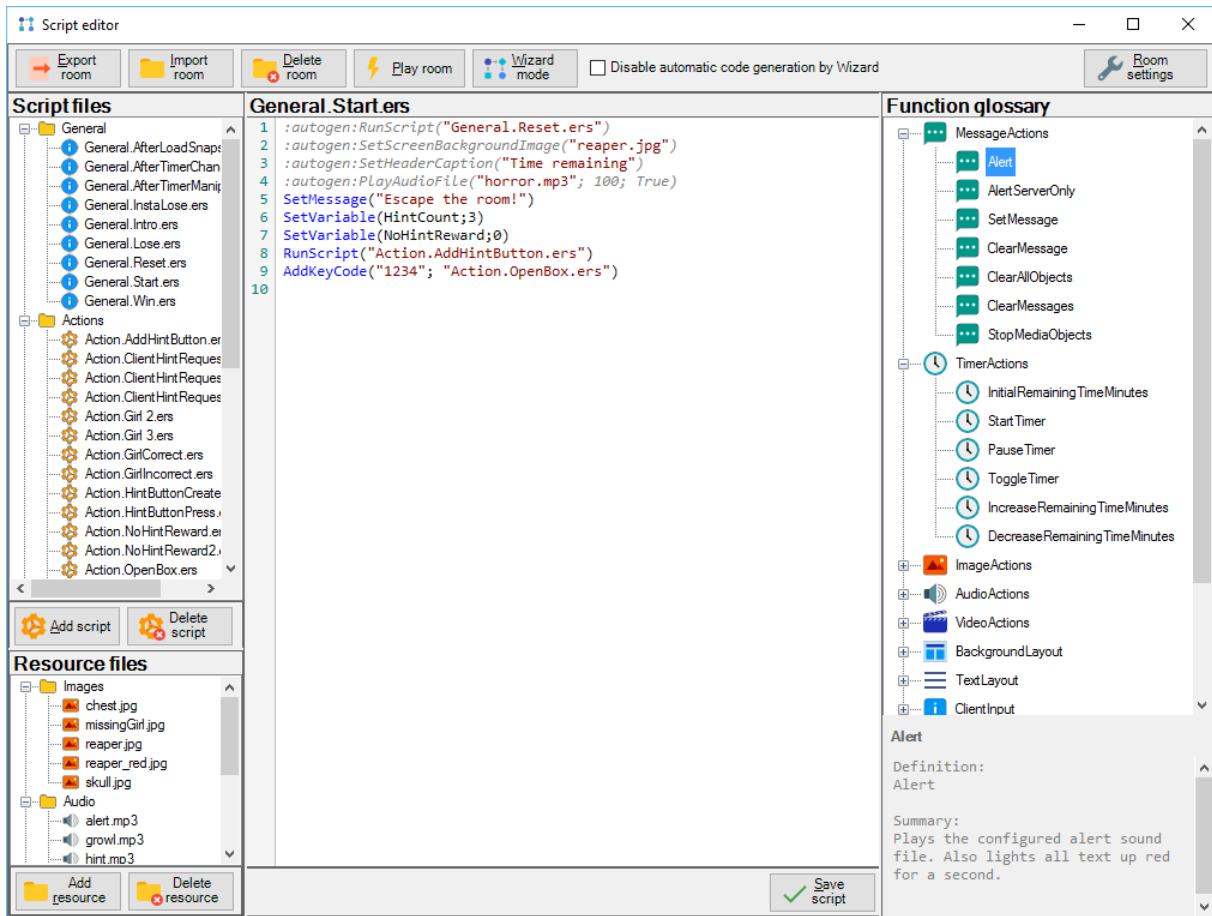
Room management panel	Details
Export room	Export the room folder to a ZIP file in the room zip folder (see the configuration screen).
Import room	Import a room ZIP file from the room zip folder (see the configuration screen).
Delete room	Delete the current room. You will be asked if you wish to create a new, empty room. The New room wizard will appear.
Play room	Start the room in DEBUG mode for easy testing.
Wizard mode	Quickly switch between the script editor and the wizard. Please note that the wizard autogenerates codes and usually, but not always respects manually added code.
Disable automatic code generation by wizard	Check this box to make sure the editor always starts in script editor mode instead of wizard mode.
Room settings	Open the room settings screen.

Files pane (left)	Details
Script files tree (general, actions, timed events, manual actions)	Shows all script files in the room folder.
New script button	Create a new script. You will be asked what type of script you wish to add. Please note that you cannot add general scripts, as they are a fixed set of scripts.
Delete script button	Delete the selected script from the room folder. Please note that you cannot delete general scripts, as they are a fixed set of scripts.

Rename script	When using the right mouse button on a scriptfile, you can rename it.
Resource files tree (images, audio, video, other)	Displays and categorises all resource files in the room folder. The right mousebutton als be used to access add/delete and previewing of the resource files. Please note that you can also add resources by dragging files from your Windows explorer onto this tree.
Add resource	Add a resource tot the room folder.
Delete resource	Delete the selected resource from the room folder.

Script editor panel (center, right)	Details
Script editor	Shows the current selected script from the script files tree. When making a modification, a * is shown begind the script file in the script files tree. Functions are colored **blue**, strings are colored **red** for easier working. {**purple** text are GUIDs, which are unique identifier codes for other scripts. Hover above a GUID to see the actual name of the script that is referred to. Use the right mousebutton to insert a GUID for any existing script} //Remarked tex is shown **green** and execution of these lines is skipped :autogen: code is **gray**. It is code autogenerated by the wizard and may be overwritten when revisiting wizard mode.
Save script button	Save the current opened script.
Function glossary tree	Displays all functions. Doubleclick to add a function tot he script editor.
Function glossary info panel	Describes the syntax and details of the function selected in the function glossary tree.

Add script	Details
Action	Add a generic action that can be kicked off by other scripts or triggers.
Timed event	Add a script that will fire at a specific time in the game (based on minutes and seconds).
Manual (clue)	Add a manual (clue) script that will show in the clue selector on the server screen.

5.10. In-room client screen

The client screen is *stay on top*.
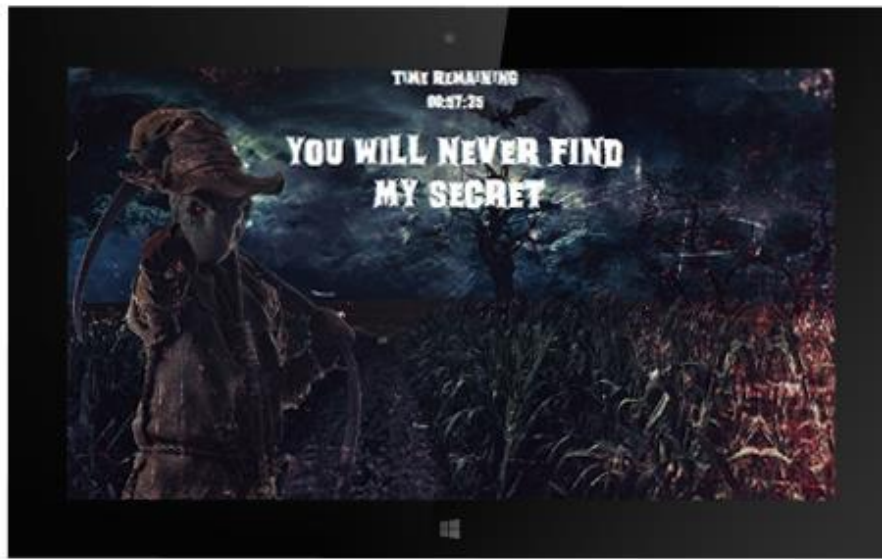
Interaction possibilities	Details
Keyboard - F10 key	Toggle logfile visibility.
Keyboard - F11 key	Toggle normal/locked window.
Keyboard - F12 key	Close the client with the F12-key.
Touchscreen buttons	You can configure touchscreen buttons and associated scripts in the editor.
Keyboard/keypad input or USB barcode scanning	You can configure codes and associated scripts in the editor. Just enter the code in the keyboard and press ENTER. Please note that a USB barcodescanner behaves just as a keyboard and should give a RETURN-character (ENTER key) at the end of scanning each barcode.

5.11. Limiting access on the client computer

Windows 8.x and 10 support Windows kiosk-mode. You can thoroughly lock the client screen computer and limit user access on the computer by running the application in kiosk mode. You do need a Pro version of Windows for this feature though.

Other tricks are to hide the taskbar, the charms panel in Windows 8.x or the Action Center in Windows 10.

Microsoft offers a number of solutions on their website.

5.12. Importing a room on a dedicated client computer

Use the main menu to import and configure a room when using a dedicated CLIENT computer.

- Make sure both CLIENT and SERVER run the same version of Escape Room Software

- Make sure both CLIENT and SERVER computers have power saving settings and screensavers disabled

- When using different fonts, make sure all CLIENT and SERVER computers have the proper fonts installed

6. Scripting guide

6.1. General

Every script that is executed runs all actions in chronological order. There is no pausing between actions.

6.2. Script types

The following script types are available:

Name	Purpose	Examples of use
General	General scripts are run at fixed core-events in the game.	See the next paragraph for more information.
Actions	These scripts are run only when referenced by another script.	Create reusable scripts that can be executed from multiple places. Also, a variable comparison could cause a specific action script to run.
Timed events	These scripts run automatically at the specific minute that you choose.	Increase tension by changing images and music in the last 10 minutes of play.
Manual actions	These scripts are available for the game master to run manually by doubleclicking on them.	Use manual actions for prepared clues or effects to influence the room ambiance.
Home automation scripts	These scripts are run when specific home automation triggers are fired.	Such scripts are fired when a home automation event is fired.

6.3. General script definitions

The general scripts have the following purpose:

Name	Event	Details
Reset	Automatically executed when the room is loaded, OR when the game master presses the RESET button.	Here you usually define the fonts, font sizes, colors, room timer value (default is 60 minutes), etc.
Intro	Executed when the game master presses the INTRO button.	Used for intro movies.
Start	Executed when the game master presses the START button.	
AfterTimerChange	Always executed after timer has changed	Use the ! variable to access the current time.
AfterTimerManipulation	Executed after time has been manipulated by manually pressing a + or - button on the server screen.	Use the ! variable to access the current time. You could use this to match up a playing video with the room timer.
AfterManualMessage	Executed after the GameMaster (GM) manually sends a message using the button on the GM panel.	
Win	Executed when the game master presses the WIN button.	
Lose	Executed when the game timer expires.	
InstaLose	Executed when the game master presses the LOSE button.	Per default the InstaLose script automatically runs the normal General.Lose script.

6.4. Examples

1. How to create a room with a red background and white text? (simple)

Modify script: General.Reset.ers
```
SetScreenBackColor("red")
SetFontColor("white")
```

---

2. How to create a room with music that changes when there are 5 minutes remaining? (simple)

Modify script: General.Start.ers
```
PlayAudioFile("audiofile_normal.mp3"; 100; true)
```

Add timed script at 5 minutes: Timed.005.ers
```
PlayAudioFile("audiofile_scary.mp3"; 100; true)
```

---

3. How to create a room with a quiz using buttons that allows players to win time? (advanced)

Modify script: General.Start.ers
```
SetVariable(QuizAnswered;0)
```

Add timed script at 45 minutes: Timed.045.ers
```
SetImage("MysteriousPictureOfAGirl.jpg")
SetMessage("A girl from the village vanished#in this house..."; "white")
```

Add timed script at 44 minutes: Timed.044.ers
```
SetMessage("#Her#name#was#Emily"; "gray")
ClearMessages(10)
```

Add timed script at 5 minutes: Timed.005.ers
```
CompareVariable(QuizAnswered;=;0;"Action.Quiz.ers";"")
```

Add action script: Action.Quiz.ers
```
SetMessage("What was the name of the#missing girl?"; "yellow")
AddButton("Answer1"; "Rachel"; ""; C-300; 300; 250; 100; "GUID of Action GirlIncorrect")
AddButton("Answer2"; "Max"; ""; C+50; 300; 250; 100; "GUID of Action GirlIncorrect")
AddButton("Answer3"; "Emily"; ""; C-300; 500; 250; 100; "GUID of Action GirlCorrect")
AddButton("Answer4"; "Chloe"; ""; C+50; 500; 250; 100; "GUID of Action GirlIncorrect")
```

Add action script: Action.GirlCorrect.ers
```
SetVariable(QuizAnswered;1)
RemoveAllButtons
SetMessage("Correct!##You receive 3 minutes bonus time!"; "limegreen")
IncreaseRemainingTimeMinutes(3)
ClearMessages(10)
```

Add action script: Action.GirlIncorrect.ers
```
SetVariable(QuizAnswered;1)
RemoveAllButtons
SetMessage("Alas, the answer is incorrect.")
ClearMessages(10)
```

Please import the included **SampleRoom** file to check out more examples.

7. Function glossary

7.1. MessageActions

Name	Sample code	Details
Alert	Alert	Plays the configured alert sound. Also lights all text up red for a second.
AlertServerOnly(string Message)	AlertServerOnly("Notification message here!")	Shows a message in a red color on the server screen only, to inform the game master that something has happend or indicate that a manual action is required. Click on the label to hide it again. Also note that [variables] are supported. Also plays the configured alert sound, on the server even though it may be muted. If no special server sound is configured, the normal alert sound is played.
SetMessage(string Message; optional string ColorName)	SetMessage("Hello world!") SetMessage("Colored message here!"; "Red") SetMessage("This is a variable value: [variableName]")	Sends a text message. Use the color parameter to indicate a color. Use the # character for a line-break. Also note that [variables] are supported. Search the internet for .NET colornames table.
ClearMessage	ClearMessage	Clears current shown message (if any).
ClearAllObjects(optional int Seconds)	ClearAllObjects(0)	Stops all playing media objects. Optionally after a number of seconds. Also clears message and any images.
ClearMessages(optional int Seconds)	ClearMessages(0)	Stops all playing media objects. Optionally after a number of seconds. Also clears message and any images.
StopMediaObjects(optional int Seconds)	StopMediaObjects(0)	Clears message and image objects. Optionally after a number of seconds.

7.2. TimerActions

Name	Sample code	Details
InitialRemainingTimeMinutes(int Minutes)	InitialRemainingTimeMinutes(60)	Sets initial remaining time for the room to a number of minutes.
StartTimer	StartTimer	Starts the timer.
PauseTimer	PauseTimer	Pauses the timer.
ToggleTimer	ToggleTimer	Toggles timer between started and paused.
IncreaseRemainingTimeMinutes(int Minutes)	IncreaseRemainingTimeMinutes(0)	Increases game timer with a number of minutes.
DecreaseRemainingTimeMinutes(int Minutes)	DecreaseRemainingTimeMinutes(0)	Decreases game timer with a number of minutes.
JumpToTimer(int Minutes)	JumpToTimer(0)	Jump the timer to a specific number of minutes.
LoopTimer	LoopTimer	Reset the time elapsed to 0 after timer finishes, without losing the game. This allows for an infinite cyclic strategy.

7.3. ImageActions

Name	Sample code	Details
SetImage(string FileName; bool FullScreen)	SetImage("imagefile.jpg"; False)	Loads and shows image. Use True or False to indicate whether the image should be shown full screen. Supported filetypes (.jpg, .jpeg, .png, .bmp)
FlickerImage(int NumberOfFlickers)	FlickerImage(100)	Flickers the loaded image a number of times for a scary effect.
ClearImage	ClearImage	Hides current shown image (if any).
SetImageLocationSize(int Left; int Top; int Width; int Height)	SetImageLocationSize(10; 10; 10; 10)	Sets screen location and size for the IMAGE box. Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Use % character for width and height to make this object a specific percentage of the screen size, or to place the top or left at a specific percentage of the screen.

7.4. AudioActions

Name	Sample code	Details
PlayAudioFile(string FileName; int Volume; bool Loop; optional int Channel)	PlayAudioFile("audiofile.mp3"; 100; False)	Plays audio file at a specified volume percentage. Use True or False to indicate whether the audio should be played in a loop. Multiple files can be played simultaneously. Optionally use different channels to organise audio. Supported filetypes (.mp3, .wav)
PlayAudioFileAndStartTimer(string FileName; int Volume; bool Loop; optional int Channel)	PlayAudioFileAndStartTimer ("audiofile.mp3"; 100; False)	Plays audio file at a specified volume percentage and starts the timer upon finishing. Supported filetypes (.mp3, .wav)
ChangeAudioVolume(int Volume; optional int Channel)	ChangeAudioVolume(100)	Plays a running audio file at a specified volume percentage. Optionally use different channels to organise audio.
StopAllAudio(optional int Channel)	StopAllAudio	Stops all playing audio objects. Optionally choose a specific channel to stop audio on.
SetAlertFile(string FileName; optional int Volume)	SetAlertFile("alert.mp3"; 100)	Set audio file for playing alerts.
SetServerAlertFile(string FileName; optional int Volume)	SetServerAlertFile("serverAlert.mp3"; 100)	Override alert audio file on the server.

7.5. VideoActions

Name	Sample code	Details
PlayVideoFile(string FileName; bool Loop)	PlayVideoFile("videofile.avi"; False)	Plays video file. Use True or False to indicate whether the video should be played in a loop. Only one file can be played simultaneously. Supported filetypes (.avi, .mp4, .mov)
PlayVideoFileAndStartTimer(string FileName; bool Loop)	PlayVideoFileAndStartTimer("videofile.avi"; False)	Same as PlayVideoFile, but also starts timer upon finishing. Used as followup for the intro button.
SetVideoFileTimer(int Seconds) SetVideoFileTimer(special function)	SetVideoFileTimer(!)	Set the video timer to a specific time in seconds. Use the ! variable to match up with the current timer value.
OffsetVideoFileTimer(int OffsetInSeconds)	OffsetVideoFileTimer(0)	Offset the video timer with a specific time in seconds. You can use both positive and negative values.
PauseVideoFile	PauseVideoFile	Pause current playing video file (if any).
ResumeVideoFile	ResumeVideoFile	Resume current playing video file (if any).
StopVideoFile	StopVideoFile	Stop and hide current playing video file (if any).
TimedStopVideoFile(int Seconds)	TimedStopVideoFile(10)	Stop and hide current playing video file (if any) after a number of seconds.
SetVideoLocationSize(int Left; int Top; int Width; int Height)	SetVideoLocationSize(10; 10; 10; 10)	Sets screen location and size for the VIDEO player. Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Use % character for width and height to make this object a specific percentage of the screen size, or to place the top or left at a specific percentage of the screen.

7.6. BackgroundLayout

Name	Sample code	Details
SetScreenBackColor(string ColorName)	SetScreenBackColor("Black")	Sets the screen backcolor by color name. The default value is "black". Search the internet for the .NET color names table.
SetScreenBackgroundImage(string FileName)	SetScreenBackgroundImage("image.jpg")	Loads and shows the lower backgroundimage. Make sure the background image matches the client screen resolution to avoid black bars on top or on the sides. Supported filetypes (.jpg, .jpeg, .png, .bmp)
ClearScreenBackgroundImage	ClearScreenBackgroundImage	Clears the lower background image of the screen.
SetClientScreenPower(bool)	SetClientScreenPower(False)	Turns client screen(s) on/off. Does not work when running client and server on the same computer.

7.7. TextLayout

Name	Sample code	Details
SetFontFamily(string FontFamily)	SetFontFamily("Times New Roman")	Changes the font of the HEADER and MESSAGE and TIMER labels and BUTTONS. Make sure that the chosen font is installed on the server and on all client computers. The default value is "Times New Roman".
SetFontBold(bool)	SetFontBold(True)	Sets the font.Bold property of all labels and buttons.
SetFontItalic(bool)	SetFontItalic(True)	Sets the font.Italic property of all labels and buttons.
SetFontShadow(bool)	SetFontShadow(True)	Sets the font.Shadow property of all labels and buttons.
SetFontColor(string ForeColorName; string BackColorName)	SetFontColor("white"; "transparent")	Sets the font forecolor and backcolor for all labels and buttons by color name. The default value is "white". Search the internet for .NET colornames table.
SetHeaderCaption(string Caption)	SetHeaderCaption("Time remaining...")	Sets the value for the HEADER label. Useful for language translations. Also note that [variables] are supported.
SetHeaderFontSize(int Size)	SetHeaderFontSize(24)	Sets font size of the HEADER label. The default value is 24.
SetHeaderFont(string FontFamily; int Size; bool Bold; bool Italic; bool DropShadow; string ForeColorName; string BackColorName)	SetHeaderFont("Times New Roman"; 24; False; False; False; "white"; "transparent")	Sets font family, size, bold, italic, shadow and colors of the HEADER label. The default size value is 24.
SetHeaderLocationSize (int Left; int Top; int Width; int Height; string TextAlignment)	SetHeaderLocationSize(10; 10; 10; 10; Center)	Sets screen Location, Size and TextAlignment for the HEADER label. TextAlignment can be either Center, Left or Right. You can also use TopLeft, TopCenter, TopRight, MiddleLeft, MiddleCenter, MiddleRight, BottomLeft, BottomCenter and BottomRight for more specificplacement in the label. Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Use % character for width and height to make this object a specific

		percentage of the screen size, or to place the top or left at a specific percentage of the screen.
`TimerVisible(bool)`	`TimerVisible(False)`	Shows or hides the TIMER label. You may want to use this when using a videofile that has the timer incorporated.
`TimerFormat(string Format)`	`TimerFormat("HHMMSS")`	Use the followinf masks: - HHMMSS = 00:00:00 (hours, minutes, seconds) - MMSS = 00:00 (minutes, seconds)
`SetTimerFontSize(int Size)`	`SetTimerFontSize(32)`	Sets font size of the TIMER label. The default value is 32.
`SetTimerFont(string FontFamily; int Size; bool Bold; bool Italic; bool DropShadow; string ForeColorName; string BackColorName)`	`SetTimerFont("Times New Roman"; 32; False; False; False; "white"; "transparent")`	Sets font family, size, bold, italic, shadow and colors of the TIMER label. The default size value is 32.
`SetTimerLocationSize( int Left; int Top; int Width; int Height; string TextAlignment)`	`SetTimerLocationSize(10; 10; 10; 10; Center)`	Sets screen Location, Size and TextAlignment for the TIMER label. TextAlignment can be either Center, Left or Right. You can also use TopLeft, TopCenter, TopRight, MiddleLeft, MiddleCenter, MiddleRight, BottomLeft, BottomCenter and BottomRight for more specificplacement in the label. Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Use % character for width and height to make this object a specific percentage of the screen size, or to place the top or left at a specific percentage of the screen.
`SetSubtimerFontSize(int Size)`	`SetSubtimerFontSize(32)`	Sets font size of the SUBTIMER label. The default value is 32.
`SetSubtimerFont(string FontFamily; int Size; bool Bold; bool Italic; bool DropShadow; string ForeColorName; string BackColorName)`	`SetSubtimerFont("Times New Roman"; 32; False; False; False; "white"; "transparent")`	Sets font family, size, bold, italic, shadow and colors of the SUBTIMER label. The default size value is 32.
`SetSubtimerLocationSize(int Left; int Top; int Width; int Height; string TextAlignment)`	`SetSubtimerLocationSize(10; 10; 10; 10; Center)`	Sets screen Location, Size and TextAlignment for the SUBTIMER label. TextAlignment can be either Center, Left or Right. You can also use TopLeft, TopCenter, TopRight, MiddleLeft,

		MiddleCenter, MiddleRight, BottomLeft, BottomCenter and BottomRight for more specificplacement in the label. Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Use % character for width and height to make this object a specific percentage of the screen size, or to place the top or left at a specific percentage of the screen.
SetMessageFontSize(int Size)	SetMessageFontSize(48)	Sets font size of the MESSAGE label. The default value is 48.
SetMessageFont(string FontFamily; int Size; bool Bold; bool Italic; bool DropShadow; string ForeColorName; string BackColorName)	SetMessageFont("Times New Roman"; 48; False; False; False; "white"; "transparent")	Sets font family, size, bold, italic, shadow and colors of the MESSAGE label. The default size value is 48.
SetMessageLocationSize(int Left; int Top; int Width; int Height; string TextAlignment)	SetMessageLocationSize(10; 10; 10; 10; Center)	Sets screen Location, Size and TextAlignment for the MESSAGE label. TextAlignment can be either Center, Left or Right. You can also use TopLeft, TopCenter, TopRight, MiddleLeft, MiddleCenter, MiddleRight, BottomLeft, BottomCenter and BottomRight for more specificplacement in the label. Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Use % character for width and height to make this object a specific percentage of the screen size, or to place the top or left at a specific percentage of the screen.
SetAlertServerOnlyLocationSize(int Left; int Top; int Width; int Height; string TextAlignment)	SetAlertServerOnlyLocationSize(10; 10; 10; 10; Center)	Sets screen Location, Size and TextAlignment for the ALERT SERVER ONLY label. TextAlignment can be either Center, Left or Right. You can also use TopLeft, TopCenter, TopRight, MiddleLeft, MiddleCenter, MiddleRight, BottomLeft, BottomCenter and BottomRight for more specificplacement in the label.

	Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Use % character for width and height to make this object a specific percentage of the screen size, or to place the top or left at a specific percentage of the screen.

7.8. ClientInput

Name	Sample code	Details
AddButton(string ButtonName; string Caption; string ImageFileName; int Left; int Top; int Width; int Height; string ScriptGUID)	AddButton("NewButton"; "Caption"; "image.jpg (leave blank if not needed)"; 10; 10; 250; 100; "scriptGUID")	Adds a button to the client screen that kicks off a script when pressed. An example of useful interaction is for example a HINT button. Use the C (=Center) character to offset from center (example: C-100 or C+100) for left and top variables. Also note that [variables] are supported for the caption. Use % character for width and height to make this object a specific percentage of the screen size, or to place the top or left at a specific percentage of the screen.
RemoveButton(string ButtonName)	RemoveButton("Button1")	Removes specific button from the client screen.
RemoveAllButtons	RemoveAllButtons	Removes all buttons from the client screen.
SetButtonFont(string FontFamily; int Size; bool Bold; bool Italic; bool DropShadow; string ForeColorName; string BackColorName)	SetButtonFont("Times New Roman"; 24; False; False; False; "white"; "transparent")	Sets font family, size, bold, italic, shadow and colors for any BUTTONS plased subsequently. The default size value is 24.
AddKeyCode(string KeyCode; string ScriptGUID)	AddKeyCode("12345"; "scriptGUID")	Players can use a numpad or a barcodescanner attached to the client to enter codes or scan codes clues, triggering a server-side script. A barcode scanner automatically enters the code and sends the ENTER key, just like you would using a manual keyboard. Some barcode scanners are configured to also send a checkdigit at the end, so be sure to test for this. Please note that QR code scanning using webcam is not supported.
RemoveKeyCode(string KeyCode)	RemoveKeyCode("12345")	Removes a specific keycode.
RemoveAllKeyCodes	RemoveAllKeyCodes	Removes all keycodes.
SetKeyCodeNotFoundScript(string ScriptGUID)	SetKeyCodeNotFoundScript ("12345"; "scriptGUID")	Set a script to be fired when a wrong keycode is entered.
ClearKeyCodeNotFoundScript	ClearKeyCodeNotFoundScript	Clears the script to be fired when a wrong keycode is entered.

7.9. Scripting

Name	Sample code	Details
RunScript(string ScriptGUID)	RunScript("scriptGUID")	Runs another script. The other script is finished before the running script continues.
RunTimedScript(string TimerName; string ScriptGUID; int Seconds)	RunTimedScript("TimerName"; "scriptGUID"; 0)	Runs another script after a number of seconds. The running script continues in the mean time. Please note that you can run multiple scripts like this simultaneously.
StopTimedScript(string TimerName)	StopTimedScript("TimerName")	Stops the running a RunTimedScript action.
StopAllTimedScripts	StopAllTimedScripts	Stops the running all RunTimedScript actions.
WinRoom	WinRoom	Finish the game – Win status
LoseRoom	LoseRoom	Finish the game – Lose status

7.10.　Connectivity

Name	Sample code	Details
ShellOnServer(string FileName; optional string Arguments)	ShellOnServer("c:\notepad.exe"; /p)	Runs an external command on the SERVER computer. Optionally add a commandline parameter.
ShellOnClient(string FileName; optional string Arguments)	ShellOnClient("c:\notepad.exe"; /p)	Runs an external command on the CLIENT computer. Optionally add a commandline parameter.
BroadcastSimpleCommand(string)	BroadcastSimpleCommand("TEST1234")	Broadcast a simple command over the configured main port. This can be used to notify other applications or devices listening on the same port.
BroadcastExternalControllerCommand(optional int ID; string value)	BroadcastExternalControllerCommand("TEST1234") or BroadcastExternalControllerCommand(2, "TEST1234")	Broadcast a simple command over the CONFIGURED PORT for all or a specific EXTERNAL DEVICE (low traffic, recommended when using external controllers). This can be used to notify external controller devices listening on the assigned External Controller TCP or serial port. You can use variable names using [brackets].
APICall(string URL)	APICall("http://192.168.100.2:8080/someapi/runaction?param1=0&param2=test%20me")	Runs API call on the SERVER computer. Use ? for params, & as param splitter and %20 for spaces.
PrintDocumentOnServer(string FileName)	PrintDocumentOnServer("FileName.pdf")	Print a document on the client computer default printer. Supported filetypes (.pdf)
PrintDocumentOnClient(string FileName)	PrintDocumentOnClient("FileName.pdf")	Print a document on the client computer default printer. Supported filetypes (.pdf)
SendSMTPEmailOnServer(string EmailAddress; string Subject; string Body; optional string AttachmentFileName)	SendSMTPEmailOnServer("info@info.com"; "Subject line"; "Body lines")	Send an email from the server computer. Please note that you must configure an SMTP server in the settings.
ShutDownClientComputer	ShutDownClientComputer	Turns client computer off. Does not work when running client and server on the same computer.
RestartClientComputer	RestartClientComputer	Restarts client computer. Does not work when running client and server on the same computer.

7.11. HomeAutomation

Name	Sample code	Details
HaSetSwitch(string DeviceID; bool State)	HaSetSwitch("1234"); True)	Turn a Home Automation switch on or off.

Please note that home automation integration must be enabled in the room settings.

7.12.   Variables

Name	Sample code	Details
SetVariable(variableName; var Value)	SetVariable(variableName; 0)	Sets a variable in memory. A variant can be either a *string* or an *integer* value.
IncreaseVariable(variableName; int Value)	IncreaseVariable(variableName; 1)	Increases an integer variable with a specific value.
DecreaseVariable(variableName; int Value)	DecreaseVariable(variableName; 1)	Decreases an integer variable with a specific value.
CompareVariable(variableName; operator; variableName; string ScriptGUIDWhenTrue; string ScriptGUIDWhenFalse)	CompareVariable(variableName;=;variableName;"scriptGUIDWhenTrue";"scriptGUIDWhenFalse")	Compares two variables. Assign scripts to run in case of true and false. Leave empty to not run a script. Use =, <>, < or > as operators for comparison. Use the ? character as a variable for a random result of 0 or 1. Use the ! character to get the elapsed time in seconds.
SaveVariableToSessionStats(variableName; int CustomFieldSlot)	SaveVariableToSessionStats(variableName; 1)	Save a variable to one of the 9 custom fields in the current session statistics record. Please note that there are 1-9 custom field slots available.
SaveScriptCountToSessionStats	SaveScriptCountToSessionStats	Keep track of how many times this script was run.
StrikeThroughClues	StrikeThroughClues	Strike through this clue and all earlier strike through-enabled clues in this category/puzzle on the Game Master panel.

Please note that you can place variables within [and] in message strings.

7.13.   Reserved

Name	Sample code	Details
Unknown	n/a	Reserved function, do not use.
RestartClient	n/a	Reserved function, do not use.
AutomaticTimerChange	n/a	Reserved function, do not use.
PacketDropped	n/a	Reserved function, do not use.
UpdateClientVariables	n/a	Reserved function, do not use.
AskServerApplicationVersion	n/a	Reserved function, do not use.
ReturnServerApplicationVersion	n/a	Reserved function, do not use.
AskServerRoomLastChangedDate	n/a	Reserved function, do not use.
ReturnServerRoomLastChangedDate	n/a	Reserved function, do not use.

7.14.   System variables

These are the system variables available in scripting mode:

Variable	Valuetype	Details	Can be used in functions:
!	int	Seconds elapsed	CompareVariable, SetVideoFileTimer
?	int	Random number 1 or 0	CompareVariable
&	string	Current language ISO	CompareVariable
[ ]	string	Use brackets to show a variable value	All string functions; SetMessage, etc.
[LastKeyCode]	string	Displays the last keycode entered by the players	All string functions; SetMessage, etc.
^	string	Time taken in minutes, based on timerformat (HH:MM:SS or MM:SS)	All string functions; SetMessage, etc.
#	String	Return character	SetMessage

7.15.   Incoming messages from external devices

If you send message from another device (TCP, UDP, Serial) using the proper syntax, the server runs a script of your choosing.

Syntax example: *@scriptGUID*

The @ character indicates that a script should be run, followed by the GUID of the script. This allows for external applications or devices to kick off scripts within the Escape Room Software server.

An external controller can add a variable by placing it between brackets after the normal syntax.

Syntax example: *@scriptGUID[variableName;Value]*

Before the script is executed on the server, first the variable of that name is set to the proposed value. If the variable does not yet exist, it is created.

8. NET colors overview

8.1. Color chart

AliceBlue	DarkTurquoise	LightSeaGreen	PapayaWhip
AntiqueWhite	DarkViolet	LightSkyBlue	PeachPuff
Aqua	DeepPink	LightSlateGray	Peru
Aquamarine	DeepSkyBlue	LightSteelBlue	Pink
Azure	DimGray	LightYellow	Plum
Beige	DodgerBlue	Lime	PowderBlue
Bisque	Firebrick	LimeGreen	Purple
Black	FloralWhite	Linen	Red
BlanchedAlmond	ForestGreen	Magenta	RosyBrown
Blue	Fuchsia	Maroon	RoyalBlue
BlueViolet	Gainsboro	MediumAquamarine	SaddleBrown
Brown	GhostWhite	MediumBlue	Salmon
BurlyWood	Gold	MediumOrchid	SandyBrown
CadetBlue	Goldenrod	MediumPurple	SeaGreen
Chartreuse	Gray	MediumSeaGreen	SeaShell
Chocolate	Green	MediumSlateBlue	Sienna
Coral	GreenYellow	MediumSpringGreen	Silver
CornflowerBlue	Honeydew	MediumTurquoise	SkyBlue
Cornsilk	HotPink	MediumVioletRed	SlateBlue
Crimson	IndianRed	MidnightBlue	SlateGray
Cyan	Indigo	MintCream	Snow
DarkBlue	Ivory	MistyRose	SpringGreen
DarkCyan	Khaki	Moccasin	SteelBlue
DarkGoldenrod	Lavender	NavajoWhite	Tan
DarkGray	LavenderBlush	Navy	Teal
DarkGreen	LawnGreen	OldLace	Thistle
DarkKhaki	LemonChiffon	Olive	Tomato
DarkMagenta	LightBlue	OliveDrab	Transparent
DarkOliveGreen	LightCoral	Orange	Turquoise
DarkOrange	LightCyan	OrangeRed	Violet
DarkOrchid	LightGoldenrodYellow	Orchid	Wheat
DarkRed	LightGray	PaleGoldenrod	White
DarkSalmon	LightGreen	PaleGreen	WhiteSmoke
DarkSeaGreen	LightPink	PaleTurquoise	Yellow
DarkSlateBlue	LightSalmon	PaleVioletRed	YellowGreen
DarkSlateGray			

9. How to connect to a micro controller

9.1. Arduino

9.1.1. API connection

9.1.1.1. Send API call from Arduino to ERS Server

```
#include <WiFi.h> // Use <ESP8266WiFi.h> for ESP8266

const char* ssid = "YOUR_SSID";     // Replace with your network SSID
const char* password = "YOUR_PASSWORD"; // Replace with your network password

WiFiClient client;

void setup() {
  Serial.begin(115200);
  delay(10);

  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  Serial.print("Connecting to WiFi");

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("\nConnected to WiFi");
}

void loop() {
  String ipAddress = "127.0.0.1"; // Change as needed; localhost may not work; use actual IP address when running locally
  int port = 8080; // Change as needed
  String roomID = ""; // Set the roomID if necessary
  String action = "{846e92d0-299c-454b-a799-3b4227ddb862}"; // Set the action as needed
  String url = "http://" + ipAddress + ":" + String(port) + "/ersapi/runaction?roomID=" + roomID + "&action=" + action;

  if (client.connect(ipAddress.c_str(), port)) {
    Serial.println("Connected to server");

    // Send HTTP GET request
    client.print(String("GET ") + url + " HTTP/1.1\r\n" +
           "Host: " + ipAddress + "\r\n" +
           "Connection: close\r\n\r\n");

    // Wait for server response
    while (client.connected() || client.available()) {
      if (client.available()) {
        String line = client.readStringUntil('\n');
        Serial.println(line);
      }
    }

    client.stop();
    Serial.println("Disconnected from server");
  } else {
    Serial.println("Connection failed");
  }
  delay(10000); // Wait 10 seconds before next request
}
```

9.1.1.2. Send API call from ERS Server to Arduino

```
#include <WiFi.h> // Use <ESP8266WiFi.h> for ESP8266
#include <WebServer.h>

// Wi-Fi credentials
const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";

// Configurable IP address and port
IPAddress localIP(192, 168, 1, 100); // Set your fixed IP address
const int port = 80; // Set your desired port

WebServer server(port);

void setup() {
  Serial.begin(115200);

  // Connect to Wi-Fi
  WiFi.mode(WIFI_STA);
  WiFi.config(localIP, WiFi.gatewayIP(), WiFi.subnetMask());
  WiFi.begin(ssid, password);

  // Wait for connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("Connected to WiFi");
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());

  // Define the API route
  server.on("/Ping", HTTP_POST, handlePing);

  // Start the server
  server.begin();
  Serial.println("Server started");
}

void loop() {
  server.handleClient();
}

void handlePing() {
  if (server.hasArg("plain")) {
    String message = server.arg("plain");
    Serial.print("Received: ");
    Serial.println(message);
    server.send(200, "text/plain", "Pong");
  } else {
    server.send(400, "text/plain", "Bad Request");
  }
}
```

9.1.2. TCP ethernet connection (not recommended, please use API)

```
// Escaperoom Software Arduino demo project TCP.Ethernet
// Version 1.6 (8/15/2017)
// Sample compiled from various sources found on www.arduino.cc
// Modified by Escaperoom Software

#include <Ethernet.h>
#include <SPI.h>

// please configure these settings
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; // MAC address (get from sticker on device, or make up your own: must
be unique in the network)
byte ipLocal[] = { 192, 168, 100, 129 }; // local IP (must be unique in the network)
byte ipServer[] = { 192, 168, 100, 185 }; // server IP
unsigned int port = 13809; // room TCP port (the accompanying demo room sample uses TCP port 13009) Please do not
confuse the TCP port and sample with the UDP port and sample
bool isDebugMode = true;

// internal variables (don't change)
bool isConnecting = false;
bool isConnected = false;
String readLine = "";
String commandSplitter = "_NEXT_ERSCOMMAND_";
char commandSplitterChar[] = "_NEXT_ERSCOMMAND_";
bool testMessageSent = false;
EthernetClient client;

void setup()
{
}

void reConnect()
{
  isConnecting=true;
  Ethernet.begin(mac, ipLocal);
  Serial.begin(9600);
  delay(1000);
  serialPrint("Connecting...");

  if (client.connect(ipServer, port)) {
    client.flush();
    serialPrint("Connected.");
    isConnected = true;
  } else {
    serialPrint("Connection failed. Retry.");
  }
  isConnecting=false;
}

void loop()
{
  if (!isConnected) {
    if (!isConnecting) {
      reConnect();
    }
  } else {
    if (client.available()>0){
      clientReadIncomingMessage();
      serverWriteOutgoingMessage();
    }
  }
```

```
  if (!isConnecting) {
   if (!client.connected()) {
     serialPrint("Disconnecting. Retry.");
     client.stop();
     isConnected = false;
   }
  }
}

void clientReadIncomingMessage()
{
 char thisChar = client.read();
 readLine += String(thisChar);

 if (readLine.endsWith(commandSplitter)) {
   client.flush();
   serialPrint("Line received from server:");
   processIncomingMessage(readLine);
   readLine="";
 }
}

void processIncomingMessage(String im)
{
 String value = im;

 if (isDebugMode) {
   displayIncomingMessage(value);
 }

 // PROCESS YOUR INCOMING MESSAGES HERE

}

void displayIncomingMessage(String im)
{
 serialPrint(im);
}

void serverWriteOutgoingMessage()
{

 // SEND YOUR OUTGOING MESSAGES HERE

 if (!testMessageSent) {     |
   serialPrint("Sending test message...");
   client.write(prepMessage("@Action.ServerTest.ers")); // WRITE AN @ACTION NAME TO THE SERVER TO EXECUTE IT
   testMessageSent=true;
 }
}

void serialPrint(String serialMessage)
{
 Serial.print(serialMessage);
 Serial.print('\n');
 Serial.flush();
}

char *prepMessage(char result[]){
 strcat(result, commandSplitterChar);
 return result;
}
```

9.1.3. TCP serial connection (not recommended, please use API)

```
// Escaperoom Software Arduino demo project Serial
// Version 1.0 (8/22/2017)

// Sample compiled from various sources found on www.arduino.cc
// Modified by Escaperoom Software

// internal variables (don't change)
String readLine = "";
String commandSplitter = "_NEXT_ERSCOMMAND_";
bool testMessageSent = false;

void setup()
{
  reconnect();
}

void reconnect()
{
  Serial.begin(9600);
  while (!Serial) {}
}

void loop()
{
  //Send test message
  if (!testMessageSent) {
    serverWriteOutgoingMessage("@Action.ServerTest.ers"); // WRITE AN @ACTION NAME TO THE SERVER TO EXECUTE IT
    testMessageSent = true;
  }
  clientReadIncomingMessage();
  delay(25);
}

void serverWriteOutgoingMessage(String outgoingMessage)
{
  Serial.println(prepMessage(outgoingMessage));
  Serial.flush();
}

String prepMessage(String result){
  result = result + commandSplitter;
  return result;
}

void clientReadIncomingMessage()
{
  readLine += Serial.readString();
  Serial.flush();

  if (readLine.length()>0) {
    if (readLine.indexOf(commandSplitter)>0) {
      serverWriteOutgoingMessage("@Action.MicrocontrollerResponse.ers"); //Return an action to indicate the message
was received by the Microcontroller
    }
    readLine = "";
  }
}
```

9.2. Raspberry PI

9.2.1. Send API call from Raspberry PI to ERS Server

The below will create a Python script that does an API call to ERS Server. Start ERS Server in Self-hosted API mode (see earlier chapter on this subject).

1. In Raspberry PI Console, install Requests Library
   Command: **pip install requests**

   Raspberry PI 4 and higher may require you to install components in a separate environment for security reasons.

2. Write Python script sending an API call to ERS Server.
   Code example:

```
url = fhttp://IP ADDRESS:8080/ersapi/runaction?roomID=&action=GUID
response = requests.get(url)
if response.status_code == 200:
    print("Success:", response.text)
else:
    print("Error:", response.status_code)
```

9.2.2. Send API call from ERS Server to Raspberry PI

The below will create a Python script that runs a self-hosted API on the Raspberry PI which can then be called from ERS Server.

1. In Raspberry PI Console, install Flask.
   Command: **pip install Flask**

   Raspberry PI 4 and higher may require you to install components in a separate environment for security reasons.

2. Write Python script running self-hosted API that can receive calls from external programs (such as the ERS Server). The API hosts a function called "ping" which will return the message "pong".
   Code example:

```
from flask import Flask
app = Flask(__name__)
@app.route('/ping', methods=['GET'])
def ping():
    return "Pong!", 200
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080, debug=True)
```

9.2.3. Automatically start Raspberry PI software upon startup

You may want the Raspberry PI Python program to automatically when the Raspberry Pi starts:

1. Create a new file üsing the the Console:

sudo nano /etc/systemd/system/flaskapp.service

2. Make the file similar to this:

[Unit]
Description=Flask App
After=network.target

[Service]
User=pi
WorkingDirectory=/home/pi/path/to/your/app >> actual app location
ExecStart=/usr/bin/python3 /home/pi/path/to/your/app/app.py >> actual app location
Restart=always

[Install]
WantedBy=multi-user.target

3. Now run these commands to enable the program as a startup service`

sudo systemctl daemon-reload
sudo systemctl enable flaskapp.service
sudo systemctl start flaskapp.service

10.   Frequently Asked Questions (FAQ)

10.1.   How can I release my license key for migration to a new computer?

Please perform the following steps:

1. Make sure you have stored your license key somewhere (or check the license key e-mail that you have received upon purchase)
2. Go the the Settings screen, License tab
3. Press the "Release License" button
   a. Your license key is now no longer associated with the local software
   b. The license key is released in the online license library
4. You can now use the license key on a different computer

10.1.1. I cannot recover my license key

If your are unable to access the computer that has the license key, please contact support@escaperoombuilder.com and add your ordernumber and the license key from the original e-mail that you received on purchase.

10.2.   Is there an Intercom/Camera feature

No. Both features are security intensive and network intensive. You should use a separate intercom/camera solution.