# Escape Room Controller

Last Update: 12/19/2016 Rusty (Initial)

07/02/2018 Rusty (Update for guthub commit)

## Summary

At Twisted Room Escapes (TRE) (<https://www.twistedroomescapes.com>) we built our own controller software. We are using a Window Form application.

We have implemented a very simple interface where most of the configuration is driven off the file system (i.e. directories, sub-directories, and files).

The one assumption that we made while developing the controller was the fact they we knew we were going to run the software off of a computer and Duplicate the display onto a HDTV in the escape room. With this in mind, we tried to come up with a design that wouldn't distract game participants. Any character/word typing and movement of the mouse cursor is minimized.

**The design has these basic elements**

* A Count Down timer with start, stop, pause, adjust, end type of controls.
* A Message (notepad) Area where messages are displayed to the game participants.
* The ability to show predefined clues from HTML files.
* The ability to show free form text clues.
* The ability to play user defined sounds and videos.
* The ability to systematically play user defined sounds at minute:second times
* The ability to log game start, end with game master name.

**Some interesting features and setup**

All of the games and files for a game are discovered based on directories, sub-directories, and file types. By default, the hierarchy of the games files are displayed in a tree structure, much like Window's Explorer or Window's File Manager. This allows a game Master to visually see some of the clues, sounds, and videos for a given game if needed. Experience game masters hide the tree view during game play.

Each game has set of "clue" files. The files are user defined (named), but we tend to name them with a numbering scheme or ordering. This allows the tree control (when visible) to display them in order. Our naming convention is usually room number (room within the escape game) "-" clue number. The clue files are HTML files, which allows HTML mark up (bold, underline, color schemes, images, videos, etc.). The clue is displayed in the message area of the controller when you navigate to the file in the tree control and right click and choose the menu item "Show". When the tree control is hidden, the command line interface can be used to display a clue using a "command augment" structure, i.e.

>show 01-01

We use clue files for a consistent game play. Instead of the game master free form texting messages to the participants, we have them display one of many variations of a clue or hint. Depending on the game participants, we may give them a nudge instead of a clue or answer. In any event, the presentation of the clue is predefined and consistent. We don't have issues with wrong verbiage or game master accents or bad articulation, etc. The clue can be digested by the game participants by reading it more than once. We also have clues that are riddles (play on words)... the clue has to be solved to get the clue, so to speak.

The interface also allows the game master to construct and send a free form message as well.

When the message is sent, a user defined sound is played to get the attention of the game participants. Training for the listening and acknowledgement of the sound is done in the introductory spiel (background story) at the start of the game. The game host and participants typically start inside the initial room of the game. The game host goes over the rules and interacts with the game master. The host might say "game master please give us a sample clue" and the clue is displayed and the sound plays. This gives the participants the general idea on how things work.

The game master uses a laminated card or sheet of paper that tracks the progress of the game participants. A dry erase board marker is used to check off milestones of the game. A separate sheet of clue numbers is also used by the game master. Given a puzzle that game participants need help with, there are several clue "file names" that the game master can choose from to send to the game participants.

We have found that there is still a need for bi-directional conversations with the game participants as the game progresses. Usually the participants ask for a hint or clue and the game master responds with a text message. When there is a need for the game master to say something, an amplifier with a press to talk microphone is used. See the Control Room to Escape Room Communications article for more information on this setup.

The controller uses a lot of hot keys (Control-x type of commands) and a tiny command line window at the bottom of the screen. When the game master wants to invoke some sort of command, they hit a Control-Q and this brings up a very tiny modal window at the bottom of the screen. The game master types in commands like

>show clue-01  
>play sound-01  
>video video-01

When the enter key is pressed, the action is taken and the modal window is dismissed. Game participants rarely notice these actions. (Remember, game participants see the same screen as the game master, including mouse movements). We auto hide the task bar at the bottom of the screen to minimize what the game participants see on their HDTV in the room.

The controller has the ability to play sounds and or video at a predefined time. Subdirectories under the root game can be made that represent minutes:seconds. When the controller counts down to the specific minute/second the sound file or video in that directory is played. Several of our escape room games have a suspense aspect. Playing a surprising sound like a dog bark or trumpet startles the participants and adds to the feeling that time is running out.

We use [Dropbox](https://www.dropbox.com/)to hold the configuration for all games. All of the computers are linked to the same Dropbox account. Any computer can be used for any game. You can set the game for a computer from the files system (directory of games). This allows any computer to be quickly used to run a game in the event of a computer failure. Remember, Dropbox keeps a local version of all the files on the local computer hard drive and syncs with the web. New files are sync-ed with all computers linked to that Dropbox account.

The system doesn’t require internet connectivity. It is advantageous for the file sync-ing of Dropbox, but not needed. The system works even with internet disruptions.

We don't rely on a server or a database or special license software or anything fancy. We keep things simple (KISS). There are enough technical challenges aside from the controller that need attention and babysitting. We didn't want to have to deal with any technical issues in regards to the controller.

We allow many of our game participants to continue playing after time runs out, if there isn't a booking back to back. We have a separate timer that increments, keeping track of the overage time. When the game master ends the game, all timers stop and a screen shot of the screen is taken and saved into a predefined directory inside the game directory structure. This allows us to post to Facebook or merge into group photos.

A gamer log is appended to for each game. It logs the start and end time of the games and the game master that over saw the game. This allows us to keep track of what game master ran which game for payroll and other auditing needs.

## Sample Screen Shots

Figure 1-1 depicts the initial screen of the controller. You can see the tree menu that normally gets hidden when the game starts.

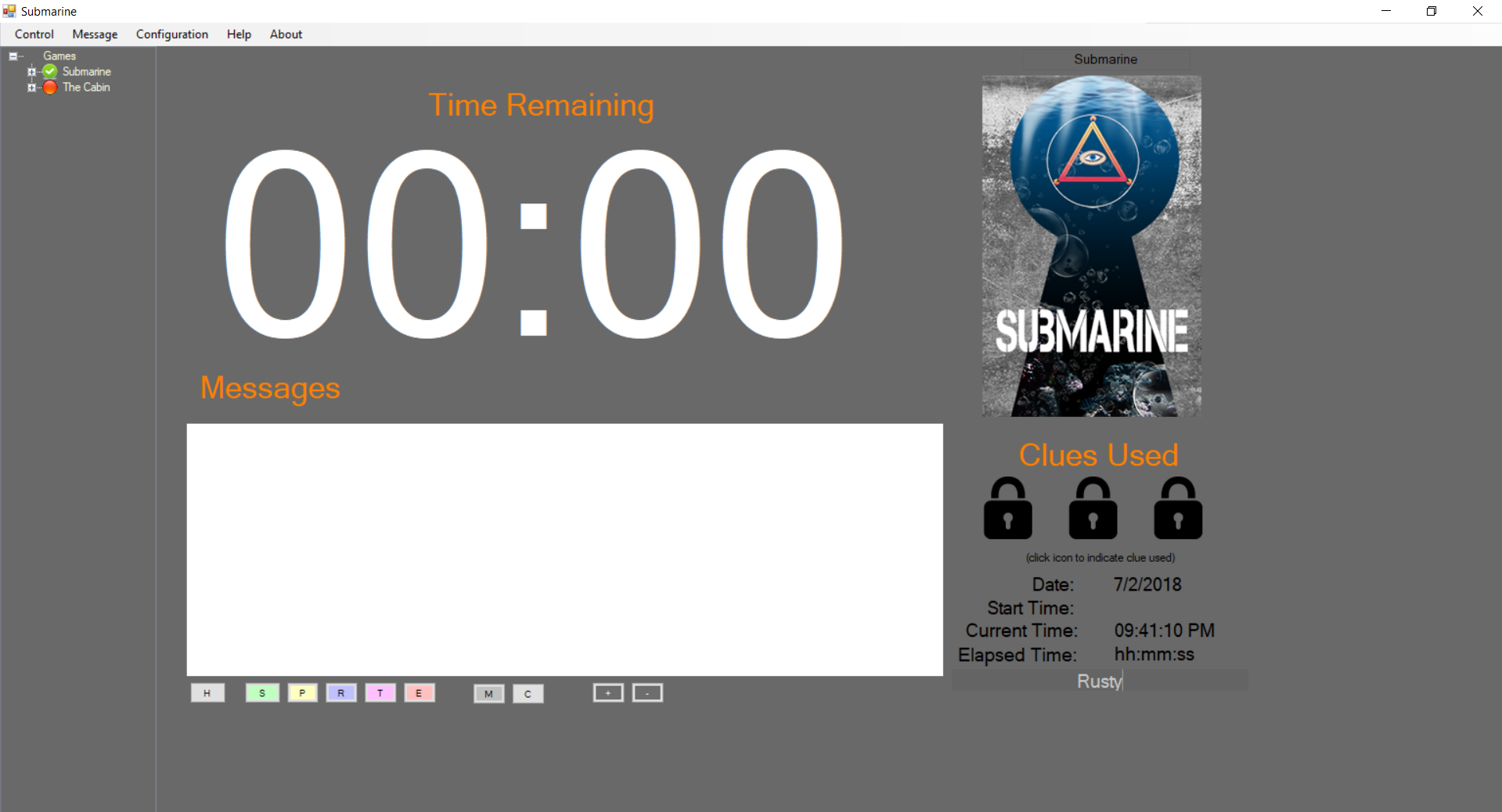


Figure 1-1

Figure 1-2 depicts the countdown timer invoked, after the game is started.

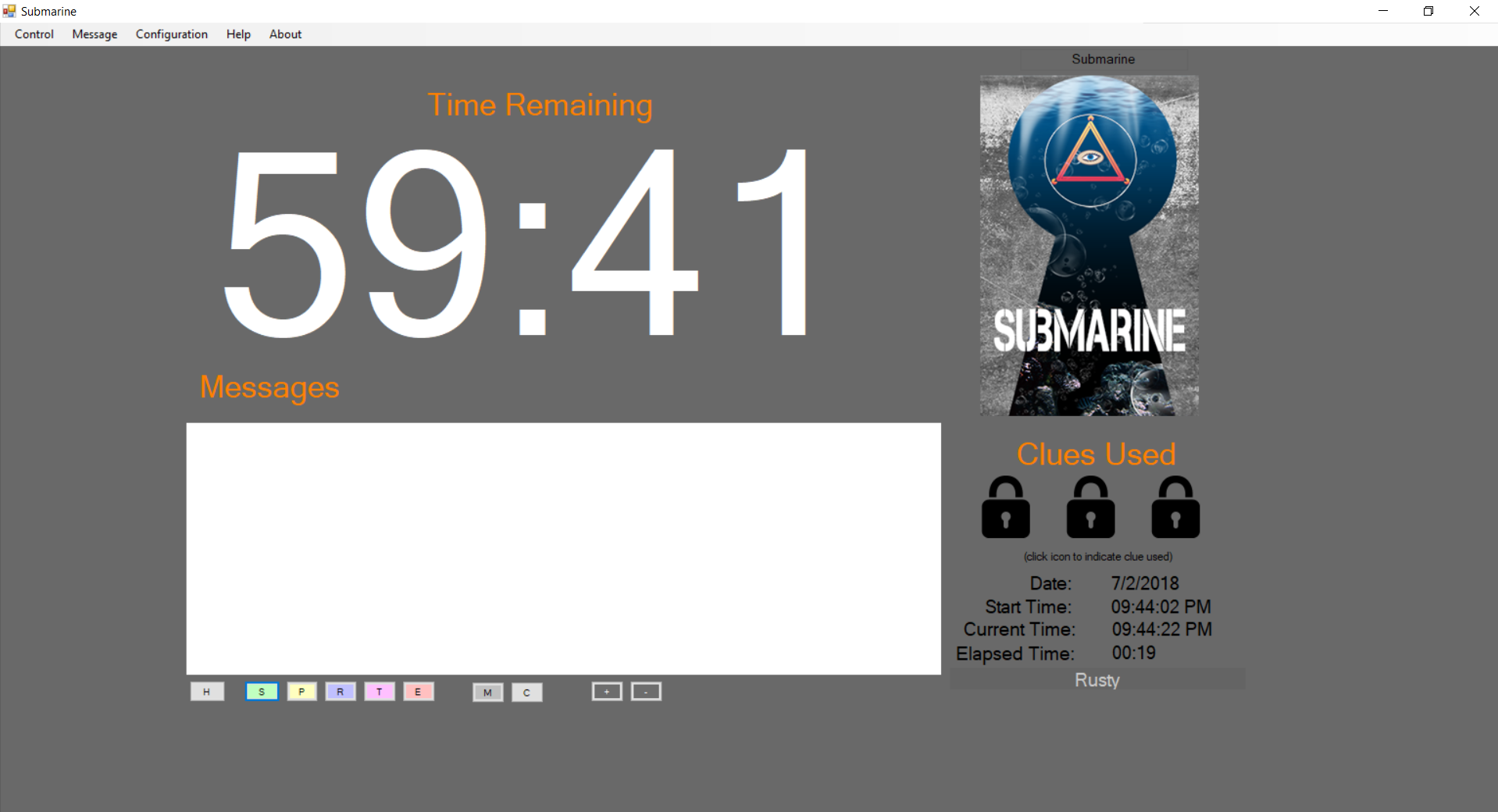


Figure 1-2

Figure 1-3 depicts the countdown timer when time has expired and the secondary timer is invoked.

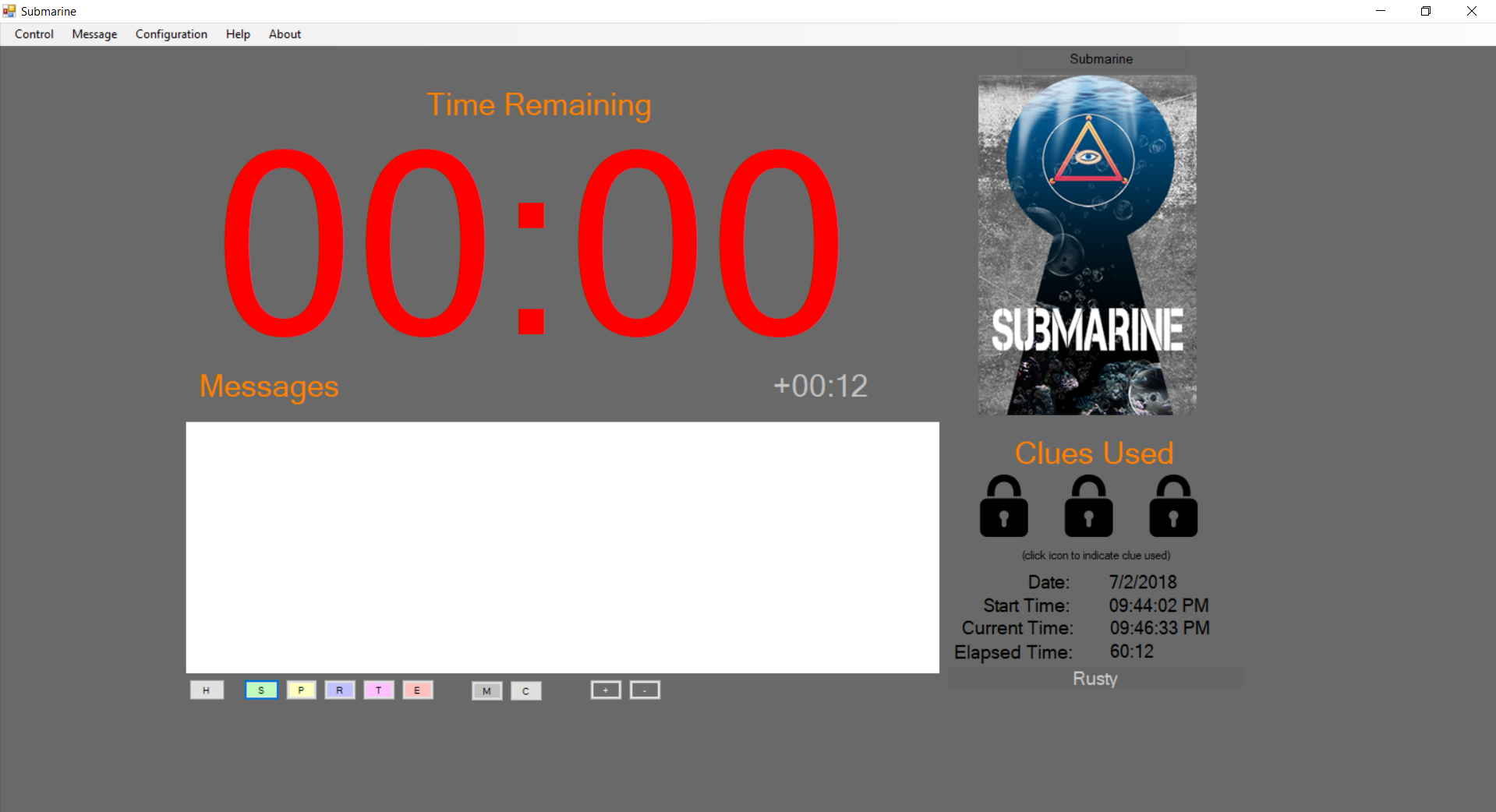
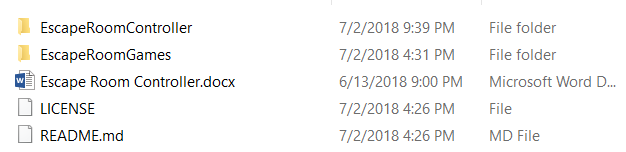


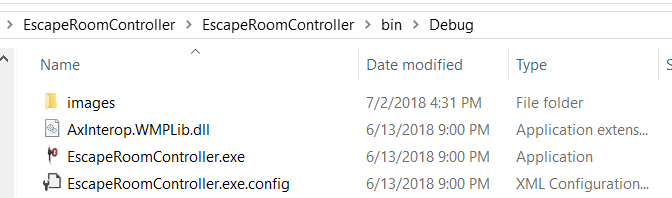
Figure 1-3

## Launching the Controller

The easiest way to get started is to download or clone the GitHub repository (<https://github.com/twistedroomescapes/EscapeRoomController> ), unzipping the files if needed. There are two main directories as seen below.



The **EscapeRoomController** directory has all of the source code files and a recent build (.exe) in the debug subdirectory several levels down.



You will need to update the config file if you want to run the controller directly from this directory. Edit the **EscapeRoomController.exe.config** file and add the directory path to the **EscapeRoomGames** directory. Figure 1-4 depicts a sample directory value for the “GamesDirectory” settings value.

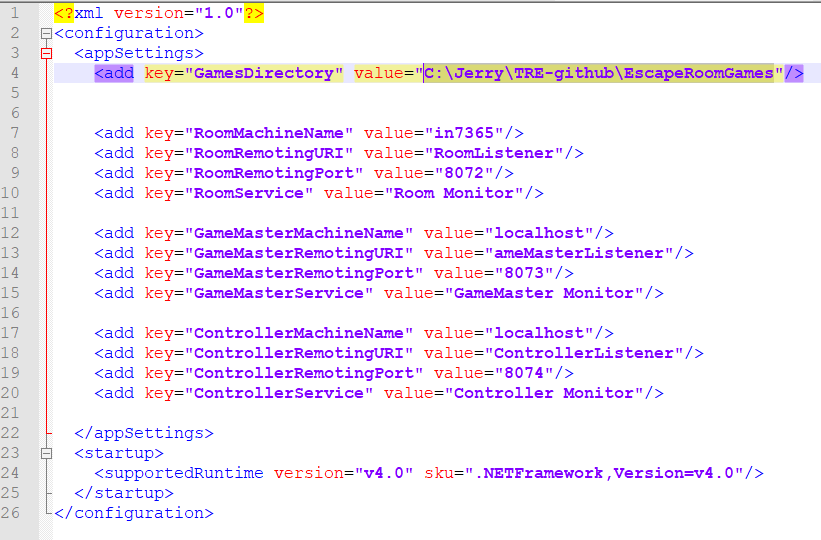


Figure 1-4

The EscapeRoomGames directory contains directories of the games and the clues, sounds, videos, etc for each game. There are a couple of pre-defined games in this directory structure. The structure contents contains a sample set of files and directories. The sample set does not contain a complete set to protect our (TRE) actual game clue and media file contents.

Launch the **EscapeRoomController.exe** executable to launch the controller.

**TODO:** Create a window’s installer to install the software.

# Features and Usage

## Countdown Timer

The countdown timer is preset to 60 minutes. When started, the countdown will be displayed on the main window of the controller. The timer can be paused and restarted. If needed, the timer can be set to a specific minute value.

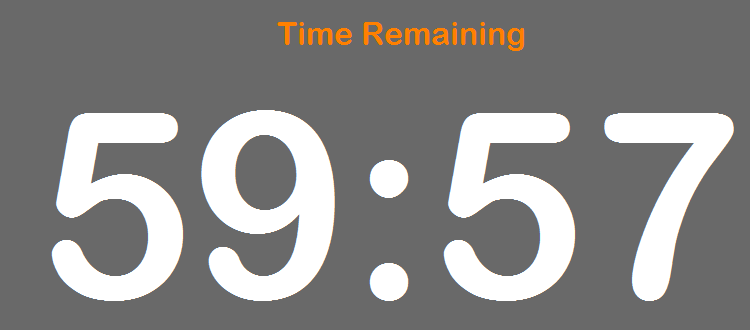


Figure n-n

When the timer starts, the displayed timer font color is green and turns white once the first second tick occurs.

For most of the duration of the timer countdown, the font will be white.

When the timer is paused, the displayed timer font color is yellow to indicate it is paused.

When the timer countdown reaches 15 minutes or less, the displayed timer font color is red to indicate time is running out.

## Timer Controls

Some of timer controls are accessed from the main menu under the [Control] menu. Choose [Timer] and select the appropriate action as seen in Figure 2-1.

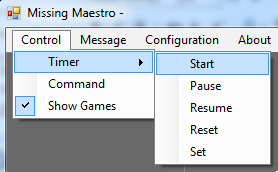


Figure 2-1

All of the controls have short-cut keys to allow the Game Master to start/pause/resume without using the menus and mouse. The short cut keys are as follows:

Control-S -🡪 Start

Control-P -🡪 Pause

Control-R -🡪 Resume

Control-T -🡪 Reset

Control-E -🡪 End

**TODO:** Note, when some controls have focus the Control-P, for example passing through the command to the windows O/S and a printer command is invoked. Still trying to figure out how to account for this. Use the menu item to pause or the Control Buttons described later to pause if a Control-key doesn’t behave as expected.

If you just want to reset the time to 60:00, us the [Reset] menu option, shortcut, or control button.

If you need to set the timer to a specific Minute:Second time, use the {Set] menu option. Another dialog will pop up for entry of a new time. See Figure 2-2.

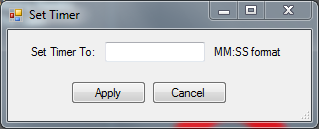


Figure 2-2

When the timer counts down to 00:00, another small timer is started and displayed just below the main timer to indicate time “overage”. See Figure 2-3. At TRE, we commonly let the participants finish the game, if there is room in the schedule to allow for it.



Figure 3-3

## Ending a Game

The game master has to end the game manually. Since play can extend past the timer hitting 00:00, ending the game is a manual operation. To end a game, press Control-E or use the [E] Control Button.

When game master “ends” the game, all timers are stopped, a screen capture image of the controller is saved in a sub-directory called “ScreenCapture”. A log file is also appended to, to capture the details of the game played.

Figure 2-4 depicts the ScreenCapture directory with some .jpg files. The directory is created as a sub-directory in the game directory. The screen capture file name is date stamped.

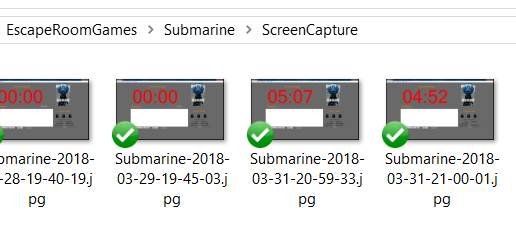


Figure 2-4

A sample of the log file can been seem in Figure 2-5. The log file is named “GameLog.txt” and is created if it doesn’t already exist. The file is appended to when the game is ended. The file is created in the same “ScreenCapture” directory as the screen capture.

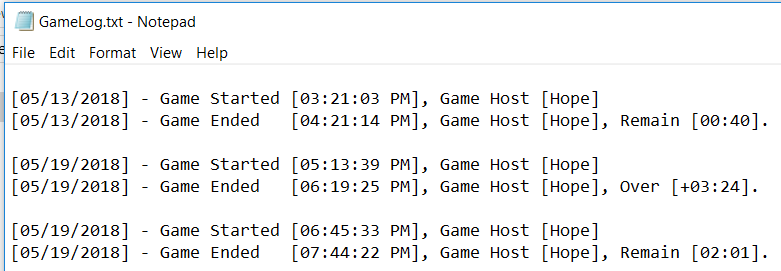


Figure 2-5

**TODO:** log file format needs to be something a little more friendlily to other systems, i.e. JSON, XML, CVS.

## Control Buttons

Just below the Messages HTML Control (text box) are some buttons to help control the game. The buttons can be hidden if desired by using the Control -> Show Buttons menu item.



H – Hides the Game Tree view

S – Start Timer

P – Pause Timer

R – Resume Timer

T – Reset timer

E – End Game

M – User message

C – Clear message

“+” – Time up (60 seconds)

“-“ – Time down (60 seconds)

## Game Tree

A game tree view (hierarchical collection of labeled items) is displayed on the left hand side of the controller. The game tree is built from the file contents (and hierarchy) found in the game folder. The game folder is configurable.

The controller reads the folder structure of this game folder (root directory) and looks for file naming conventions to build the images, clues, videos, and sounds for a particular game. The naming of the folder equates to the game name seen in a tree node. A game can be configured by dropping files in the top level folder.

A sample top level directory is shown below.

…\Documents\Dropbox\TwistedRoomEscapes\EscapeRoomGames

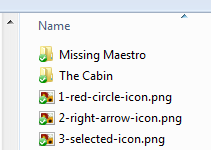


Figure 3-1

When the Controller is launched, it will create top level “Game” folder for the top level folders within the root directory. The Controller will remember the computer’s last game and set that game as the select game.

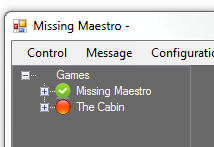


Figure 3-2

The tree view can be hidden from view by using the menu option [Show Games] from the Control menu bar. See Figure 3-3. It will be advantageous to hide the tree view when the game is in progress.

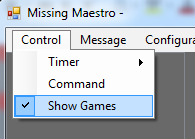


Figure 3-3

**Note:** For our setup at TRE, the game folder is located in DropBox. Every computer has DropBox installed and is connected to the same account. DropBox is installed locally so that any files updated in the master folder is synchronized to all other computers. This allows all computers to run with the same configuration. Every computer, should be dedicated to a different game. The Controller will remember, once set, the last game played.

## Selecting a Game

A game is selected by using the context menu on the game node of the tree. Right click on a node and select the [Set] menu option. The icon next to the game node will change from RED to GREEN. See Figure n-n. Once a game is set, it will be remembered next time the Controller is launched. (Setting is stored in the window’s registry).

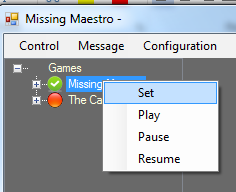
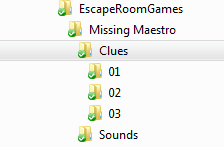


Figure 3-4

## Clues

Any number of clues can be configured for a game. Within the folder structure beneath every game folder is a “Clue” folder. This folder contains sub folders. Each sub folder contains 1 to n .HTML files.

The sub folder name can be anything, but names will be displayed within the tree view within the Controller and will be ordered sequentially. So there should be some thought into naming conventions. It is suggested to name the sub folders with a number, representing a room number of the room or puzzle the clue pertains to. i.e. directories 01, 02, 03 represent clue directories.



Contents of the clue files are read and displayed in an HTML browser control within the main window of the controller. See Figure 3-5



Figure 3-5

Contents of the clue files can have HTML markup to allow for a stylish presentation of the clue. If there are image references in the HTML markup, prefix them with "{$Base)” and make sure the image file is in the clue directory. i.e

<img src="{$Base}\myimage.png">

This allows the images to be found by the HTML browser control.

The clue directories will placed under the game folder of the tree view within the Controller. See Figure 3-6 below where the directory names are 01, 02, 03.

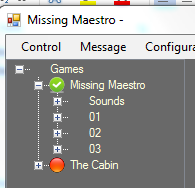


Figure 3-6

The clue files contained within each sub folder are .html files. Only files with a file extension of .html will be displayed in the tree structure. Figures 3-7 and 3-8 show both a screen shot of the directory with files and the tree view control within the controller.

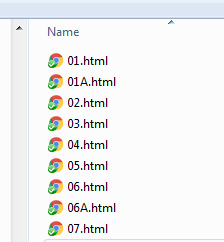


Figure 3-7

There is a context menu on each of the clue nodes in clue folder. Right click on a node you will see a [Show] menu item. Click on this to “show” the contents of the HTML file in the game controller.

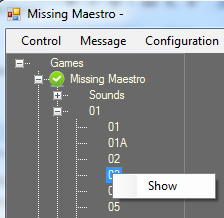


Figure 3-8

A clue can also be shown using the command line interface.

*To use the command line interface, see the section named “Command Line Interface”. The command line interface is almost exclusively used during game play to minimize what the game participants see on their HDTV monitor/screen.*

## Clue Sounds

A sound can be configured after a clue is displayed. This sound gives the participants an audio indication that a new message has been sent to the monitor in the room. By default, if there is a .wav file in the clue sub folder (folder with the .html files), this sound file will be played when the clue is displayed. If no .wav file is found in this folder, the parent folder is checked for a .wav file and that file is used.

**Note:** If there are more than one .wav file in the directory, only the first one found will be used.

## Adhoc Sounds

The controller has the ability to show and play adhoc sounds.

Within the game folder structure, a sub folder name “Sounds” can contain .wav files that will be displayed within the tree structure of the controller. These registered sounds can be played using the context menu on the node within the tree or from the command line interface.

*To use the command line interface, see the section named “Command Line Interface”. The command line interface is almost exclusively used during game play to minimize what the game participants see on their HDTV monitor/screen.*

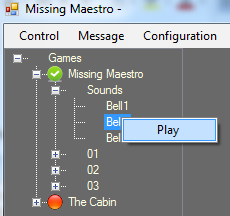


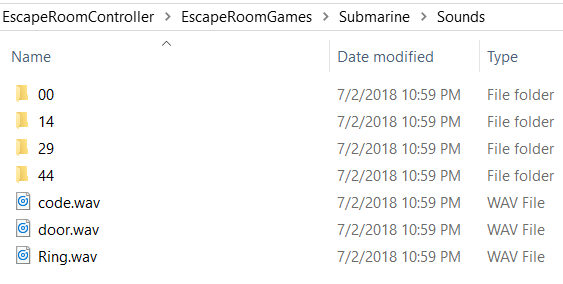
Figure 3-9

## Time Triggered Sounds

The controller has the ability to triggered a sound at a given minute within the game.

In the Sounds directory, you can create sub-directories that represent time at the MINUTE granularity. A single sound file (.wav file) can be placed in the directory and the controller will play that sound file when the countdown timer reaches that minute. A special directory of 99 can be used to play a sound when the game is ended.

Here are some samples.



00 – Play sound when time expires in the game.

14 – Play sound when there is approx.. 14 (14:59) minutes left in the game.

29 - Play sound when there is approx.. 29 (30:59) minutes left in the game.

44 – Play sound when there is approx.. 45 (44:59) minutes left in the game.

99 – Play sound when the game ends.

## Background Music

When the game is started, the controller will look for a .mp3 file in the game directory and play that sound file throughout the game. The sound file doesn’t not loop. The sound file needs to be 60 minutes or longer in length.

## Adhoc Videos

The controller has the ability to show and play adhoc videos.

Within the game folder structure, a sub folder name “Video” can contain .mp4 files that will be displayed within the tree structure of the controller. These registered videos can be played using the context menu on the node within the tree or from the command line interface.

*To use the command line interface, see the section named “Command Line Interface”. The command line interface is almost exclusively used during game play to minimize what the game participants see on their HDTV monitor/screen.*

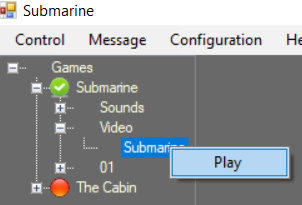


Figure 3-10

## Command Line Interface

There is a "command" menu option under the "Control" top level menu bar of the Controller. See Figure 4-1

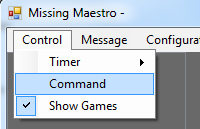


Figure 4-1

When clicked, a small window is presented. See Figure 4-2

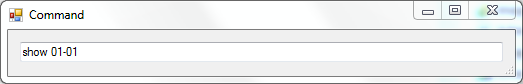


Figure 4-2

You can type commands in this window. Hitting the [Enter] key will invoke the command.

Valid commands are

show directory-file

play file

video file

All are designed to assume file extensions.

To show clue **01.html** in sub folder Clues/01, you would type the command **show 01-01** and press enter.

To play sound **sound1.wav** in the sub folder Sounds, you would type the command **play sound1** and press enter.

To play video **video1.wav** in the sub folder Video, you would type the command **video video1** and press enter.

If successful, the action is done and the window closes. If there is an error, you should get an error message.

You can use the short-cut **Control-Q** to get to the command window. This should allow the game master to hide the tree control and type Control-Q and show a clue or play a sound with minimal distraction on the shared screen.

**Note:** Before using the command line interface, you need to make sure you set the "game" context. The 'set' command from the context menu on the game node.

## Game Image

The game image can be changed by placing a file named “{name} Game.png” in the directory of the game where {name} is the directory name. So for the game “The Cabin”, the file name would be “The Cabin Game.png”. This has to be a .png file and should have the dimensions of 255 x 382 or something close.

By default, the TRE logo is displayed if the game image cannot be found. This image is in the Games Directory under the file name “Twisted Room Escapes Game.png”.

**To Be Continued - WIP**

# Tips

If you need to convert audio files from .mp3 to .wav files, search Google for online conversion tools.

# Troubleshooting