H.E.L.P. Game Rules and Mechanics Documentation

Table of Contents

Victory Conditions: Scenario 1

- Escape the Building
- in the process of escaping the building, correctly follow each step along the way in the correct order.

Victory Condition Steps: Scenario 1

1. Once the user smells smoke, use phone object to dial 911.
2. After calling 911, use phone object again to call security.
3. After calling security, proceed to door without running into the smoke (particle emitter)
4. Open door, make way to primary stairway.
5. Open Stairway door, proceed down stairway
6. Exit the building.

Failure Conditions: Scenario 1

- Not calling 911
- Not calling security
- Calling security before calling 911
- Not using the phone object
- Exiting the office before using the phone object
- Moving into or too close to the smoke in the office
- Attempting to use the elevators
- Attempting to go to roof via stairwells
- Using any doors in the stairwell except the starting and exit door
- NO time restraint

Keeping Track of Sequence of Events

- Calling 911: **Boolean Variable** (Phone.911IsCalled)
- Calling Security: **Boolean Variable** (Phone.SecurityIsCalled)
- Exit Office Room: **Boolean Variable** (Door.IsUsed)
- Entering Smoke (Fail Condition): **Failure Method** (Player.onEnterSmoke)
- Elevator Use (Fail Condition): **Failure Method** (Elevator.onUsed)
- Exiting Building: **Boolean Variable** (ExitDoor.IsUsed)
- Going upstairs (Fail Condition): **Trigger Object** (GoingUp.onEnter)
- Misuse of Stairwell doors to re-enter building (Fail Condition): **Failure Method** (StairwellDoorNumber.onUsed)

Checking:

+ Will use Boolean variables and states to keep track of what the player has done and what the player should be doing for testing for failure.

States/Events:

+ The following states will keep track of what point in the game you

are, they are named to describe what the player should be doing.
    + STATE_CALL_911
    + STATE_CALL_SECURITY
    + STATE_EXIT_OFFICE
    + STATE_EXIT_BUILDING
    + STATE_FAIL (general failure at any point in game)
    + STATE_WIN (end of game)
+ The following Boolean's are to check for failure or success, certain ones will only be checked during certain states (above) of the game.
    + **Phone.911IsCalled**
        - Checked during STATE_CALL_911
            - If (Phone.911IsCalled == true) change state = STATE_CALL_SECURITY
        - Checked during STATE_CALL_SECURITY
            - If (Phone.911IsCalled == true) change state = STATE_FAIL
    + **Phone.SecurityIsCalled**
        - Checked during STATE_CALL_911
            - If (Phone.911IsCalled == true) change state = STATE_FAIL
        - Checked during STATE_CALL_SECURITY
            - If (Phone.SecurityIsCalled) change state = STATE_EXIT_OFFICE
    + **Door.IsUsed**
        - Checked during STATE_CALL_911
            - If (Door.IsUsed == true) change state = STATE_FAIL
        - Checked during STATE_CALL_SECURITY
            - If (Door.IsUsed == true) change state = STATE_FAIL
        - Checked during STATE_EXIT_OFFICE
            - If (Door.IsUsed == true) change state = STATE_EXIT_BUILDING
    + **Player.onEnterSmoke**
        - Checked during STATE_CALL_911
            - If (Player.onEnterSmoke == true) change state = STATE_FAIL
        - Checked during STATE_CALL_SECURITY
            - If (Player.onEnterSmoke) change state = STATE_FAIL
        - Checked during STATE_EXIT_OFFICE
            - If (Player.onEnterSmoke) change state = STATE_FAIL
        - Checked during STATE_EXIT_OFFICE
            - If(Player.onEnterSmoke) change state = STATE_FAIL
    + **Elevator.onUsed**
        - Checked during STATE_EXIT_BUILDING
            - If (Elevator.onUsed == true) change state = STATE_FAIL
    + **ExitDoor.IsUsed**
        - Checked during STATE_EXIT_BUILDING
            - If (ExitDoor.IsUsed == true) change state = STATE_WIN
    + **GoingUp.onEnter**
        - Checked during STATE_EXIT_BUILDING
            - If (GoingUp.onEnter == true) change state = STATE_FAIL

+ **StairwellDoorNumber.onUsed**
    - Checked during STATE_EXIT_BUILDING
        - If (StairwellDoorNumber.onUsed == true) change state = STATE_FAIL

Interaction Objects (see pg. 3 for details on interaction)

- Office Phone
- Doors
    - Office door
    - Stairwell doors (starting floor and exit floor)
    - Stairwell doors (in between)
    - Building Exit
- Elevators

Interaction Environment (see pg. 3 for details on interaction)

- Smoke Particle Emitter
- Going up stairwell trigger

Interacting

1. Office Phone
    1. Hovering hand/cross hair over phone should activate the "usable" animation of the hand.
    2. Upon clicking on the phone, a GUI object needs to be created/made visible (2D representation of phone keypad)
    3. Numbers need to be active buttons, as well as Security speed-dial.
    4. Receiver should be "off the hook" already, exposing the receiver button as with a clear function, allowing the user to start over if they accidentally press a wrong number.
    5. A phone dial method should allow up to10 numbers to be entered (after each button is pressed, keep a counter and check if the counter = 10)
    6. Correct dialing of 911 should activate a cut scene sequence of the 911 audio clip after just 3 numbers.
    7. Dialing of other numbers should result in ??? (restart of dial method and warning of incorrect number)
    8. Pressing the speed dial button tagged "Security" should activate a cut scene of the Security audio clip without requiring two more numbers/buttons to be pressed.
    9. Pressing the Security button *before* dialing 911 should result in activation of failure method.
    10. After activating an audio clip/cut scene, the GUI object should be destroyed/made invisible.
2. Doors
    1. Hovering hand/cross hair over any door should activate the hand's "usable" animation.
    2. Clicking on any door should activate its Open animation, always opening away from the user.
    3. Office Door Failure Condition: Clicking on the office door before taking the appropriate preceding actions (using phone for 911 and security) should result in the activation of

failure method.
4. Building Exit Door Success Condition: Clicking of the building exit door should result in the activation of success method.
5. Stairwell Doors Failure Condition: Clicking on any door in the stairwell that is not on the starting floor or exit floor should result in the activation of failure method.

3. Elevator
   1. Hovering hand/cross hair over elevators should activate the hand's "usable" animation.
   2. The use of the elevators at any time during the mission should result in failure method.
4. Smoke Particle Emitter
   1. Should have no collision box (be enterable)
   2. Either the emitter object itself or trigger object should activate failure method should the player enter the immediate vicinity of the smoke.
   3. Moving near the smoke should cause the hand to become red, signaling the proximity of fire.
5. Going up Stairwell Trigger
   1. Passing into this trigger should activate failure method.

Triggers:

Objects with triggers include:

- Phone
- Door to exit office
- Door to exit building
- Stairwell doors
- Smoke
- Elevators

Trigger actions/spaces

- **Phone**
  - When in cubicle in area in front of phone, a little wider than phone an icon or button will appear in the corner of the screen allowing the user to press a button to make the phone display on screen usable to dial with.
- **Door to exit office/building/stairwell doors**
  - When standing in front of door in an area as wide as door and sticking out the same amount as the width, user will be able to click on the door and therefore have it open (outwards).
- **Smoke**
  - Smoke will have a trigger box slightly smaller than the area the smoke actually covers so that the game can make sure the player is completely inside the area before setting off a kill trigger.

GUI/Objects in View

- Hand object
  - This will need to be created as a "gun" since torque makes use of that instead of a true "hand"
  - At times that cut scenes show the actual avatar (phone calling,

complete mission), the “hand” will need to be unequipped so as not to show up being held by the avatar.
- The hand will need the following states:
  - **Normal** (unequipped or otherwise not visible): When the player is not facing any usable objects, near fire, or in a cut scene
  - **Object Usable** (equipped or visible and reaching towards usable object): When the player has a usable object in his/her cross hair.
  - **Near Fire (equipped or visible and turning an increasing color of red): When the player is near fire, the hand should be visible and turn red at intervals in relation to proximity of fire.**
  - **Unequipped/3rd Person Cut Scene(unequipped so as not to show up held by avatar): When the player is in a cut scene.**
- Cross hair
  - A cross hair should be created and placed in the center of the view so as to make activating usable objects easier.
  - Needs two states:
    - **Visible: While player is active and moving through the mission.**
    - **Invisible: While the player is in a cut scene or using the Phone Object.**
- Fire Indicator
  - A “fire indicator” should be placed in the bottom left corner to act as a warning for proximity to fire along with hand
  - Needs three states:
    - **No Fire: While the player is not near fire, it should be static and inanimate.**
    - **Near Fire: While the player is getting closer, it should flash as a warning.**
    - **Invisible:**During cut scenes
- Phone
  - A phone GUI element should appear in the center of the screen when the phone object is activated
  - Will need to have active buttons as the keypad for: 1, 2, 3, 4, 5, 6, 7, 8, 9, *, 0, #, and Security Speed Dial
  - the normal mouse cursor will need to be activated, allowing the player to press the buttons
  - Needs two states:
    - **Visible: After the player has activated the phone object**
    - **Invisible: Before and After the player has used the phone object, once the player has dialed 3 numbers or chosen the security speed dial button, and during the phone cut scenes.**

Cut-scenes

1. **Game Beginning/Camera Fly-in**
   1. Fade from black, camera positioned outside of building facing towards window
   2. Move camera in towards window
   3. Go into the building, move camera towards cubicle, keeping the camera rotated towards the player.

4. Move camera to about 3rd person perspective
5. Zoom into players head
6. Switch to 1st person view

2. **911 phone call**
   1. Make phone GUI element disappear
   2. Set camera in front of player, facing them
   3. Begin phone call animation (player with phone/hand up to his ear), begin 911 phone call sound clip
   4. After phone call sound clip has ended, set camera back to first person, set animation back to normal.
3. **Security phone call**
   1. same process as above, apply appropriate sound clip.

Non-Player Controlled Events

1. Beginning Cut scene: triggered when game is started
2. Fire/Smoke in office: triggered by a timer X seconds after game has begun
3. Firefighters running into building: triggered when player exits bottom of stairwell
4. Fire alarm activation: triggered X seconds after security call has ended

Sounds

1. **Phone beep:** each time a button on the phone is pressed, execute phone beep once.
2. **911 Clip:** after calling 911, execute this once.
3. **Security Clip:** after calling security, execute this once.
4. **Fire alarm:** -5 seconds after security clip ends, execute this to loop for the rest of the scenario.
5. **Footsteps:** Loop while moving.
6. **Fire Engine noises:** loop while player is in proximity of building exit doors.
7. **Success music:** loop after player has exited building and view has faded out.
8. **Failure music:** loop after player has failed mission and view has faded out.
9. **Door open:** if added, execute once each time a closed door is interacted with.
10. **Door close:** if added execute once each time an opened door is interacted with.