

BURGLAR BATTLE DEVELOPMENT DIARY

Week of 23/01/2023

- Worked on a prototype level idea
- Created a *MultiHoldButton* interactable (requires two people to hold in order to achieve an outcome)

Week of 30/01/2023

- Added a *MultiHoldReceiver* script which allows for event firing once both *MultiHoldButtons* have been held
- Made changes to *MultiHold* scripts to adhere to coding style
- Created a *LockableDoor* which is a door that can be opened by pressing interact, however, if held by the player for a certain duration, it will be locked which means that it can no longer be opened with press interact. Then can be held again in order to unlock.

Week of 06/02/2023

- Fixed bugs with lockable door holding and interact.
- Lockable door change: Can no longer be locked when it is open.
- Created a *StunTrap* which stuns the player once they enter the *StunTrap* collision box trigger.
- Created a *ConfusionTrap* which inverts the camera and movement controls once the player enters the collision box trigger.

Week of 13/02/2023

- Both *StunTrap* and *ConfusionTrap* now use a LayerMask as opposed to tags to check if the player has entered to help with performance.
- Created prefabs for the *StunTrap*, *ConfusionTrap*, and *MultiHoldButton*
- *MultiHoldReceiver* now uses UnityEvent instead of C# delegate event in order to make it easier for others to add functions to the event.

Week of 27/02/2023

- Created the *SimonSays* puzzle, where the player has to remember a sequence of buttons presses, exactly the same as the real-world simon says game.
- Created the Memory puzzle which the player has to remember a set of blocks within a grid and input them once they stop flashing after a few times.
- Created a *PillarPuzzle* which is a combination of SimonSays and a memory puzzle combined into one pillar that once completed, invokes an event such as a chest unlock

Week of 06/03/2023

- Created the *ButtonFrenzy* puzzle, where the player has to press a certain number of buttons within a certain time in a room. Having to physically walk around the room to complete it.
- Fixed three *SimonSays* bugs:

- Second to last block would flash to the OFF material if the puzzle finished before the flash blocks coroutine finished.
 - The blocks were pressable even during the coroutine that showed the sequence.
 - If player left the pressure pad during the RestartPuzzle coroutine, then it would finish the coroutine assuming that the player is still inside of the pressure pad.
- Although the intended result should be that the puzzle stops.
- Created *LootWheel* which is a wheel of fortune where the player can press a button to spin a wheel that gives loot randomly based on where it lands.
- Created different variations of the puzzle pillars, EASY, MEDIUM and HARD, increasing size of the grid and sequence lengths of the puzzle on both sides of the pillar.
- Created different variations of the button frenzy puzzle. EASY, MEDIUM and HARD. More buttons depending on the difficulty
- Adjusted ButtonFrenzy to make it clearer when the puzzle is on, button is pressed and puzzle is completed by changing materials based on those states.
- Adjusted ButtonFrenzy to make it easier to implement into a scene, added prefab of it too.

Week of 13/03/2023

- Made *WhackAMole* puzzle. Player has to hit a certain number of blocks which are constantly “disappearing” and “reappearing” in a grid of blocks.
- Created prefab for *WhackAMole*.
- Created prefab for *LootWheel*
- *SimonSays* puzzle can now be reset and started again once it is completed.
- Created a *SimonSaysDoor* prefab, which is a door that opens once the *SimonSays* puzzle is completed.
- Created a *SimonSays* prefab which is a standalone *SimonSays* puzzle that can be placed and hooked up anywhere without being attached to anything within the prefab.
- Added the ability to be able to increment and decrement the sequence length of the *SimonSays* puzzle from another script. (In preparation for allowing this to be increased to make the puzzle harder for when the player is caught).
- Fixed bug with *SimonSaysJailPuzzle* door prefab. (One of the buttons became invisible as the door was sliding into the *SimonSays* once completed.
- Created a jail cell to the Floor2 Room1 prefab which is a box and only allowing the player to escape it by completing the *SimonSays* puzzle.
- Implemented the *ButtonFrenzy* puzzle into the Floor2 Room10 prefab.
- Adjusted the *WhackAMole* puzzle so that there is a min and max length of time that can be adjusted for how long the block is flashing for and how long until the next block flashes up.
- Fixed a *WhackAMole* infinite while loop coroutine related bug.
- Implemented the *WhackAMole* puzzle into Floor2 Room12 prefab.
- Added one of the item pedestal puzzle pieces and loot behind a locked “cage” where it can only be opened once the *WhackAMole* puzzle is completed.
- Fixed unassigned transform error bug on a RotatingDoor in Floor2 Room3 prefab.
- Added doors to Floor2 Room12 prefab and adjusted the wall in Floor2 Hallway6 prefab

Week of 20/03/2023

- *SimonSays* puzzle gets harder once caught, taking into account which team got caught.
- Fixed *SimonSays* bug, out of range error when puzzle incremented.
- Did Phase1 of Floor2 Room13 which involved adding the *LootWheel* interactable into the room and adding a puzzle/riddle door to access it.

- Added review notes to the *GuardManager.cs* script.

Week of 27/03/2023

- Fixed issue with some puzzles where they would not work due to sound not being added in inspector.
- Fixed jail door in Floor1 as there were changes made to the *RotatingDoor* script.
- Added SFX to *SimonSays* on block press (correct and incorrect sounds).
- Added SFX to *WhackAMole* and *ButtonFrenzy* puzzles.
- Adjusted the material on *SimonSays* to make it more clear what blocks are being shown during the sequence flash.
- Added puzzle writeup for *SimonSays*, *ButtonFrenzy*, *WhackAMole*, and *LootWheel* in the Game Design Document in the GitHub wiki, explaining how to play and implement each puzzle into a scene. Also what each inspector value does.

Week of 03/04/2023

- Fixed the *SimonSays* puzzle in the jail cell being set for the wrong team for Team B.
- Adjusted the jail cells in floor2 so that they can be openable using a button press from the outside instead of needing to do the *SimonSays* puzzle.
- *SimonSays* by default is now easier, starts off with a shorter sequence and the top end sequence length is now shorter.
- Adjusted *SimonSays* so that it no longer has different rounds and you only need to complete one "round" of memorisation to open the door.
- *SimonSays* now uses all black blocks as opposed to different coloured blocks, and the flashing colour is white to ensure it's clear what is happening.
- *SimonSays* increased the size of the buttons to make it easier for controller.

Week of 10/04/2023

- *WhackAMole* increased the size of buttons. Removed some of the buttons to make it more visible which are turned on.
- *ButtonFrenzy* is now more reliant on two players by having less time to complete the puzzle and there being a clear divide between two sections that need to be completed. Size of the buttons has been increased.
- *SimonSays* made the buttons even larger to make it easier for controllers.
- Moved props inside of the Floor1 jail cells in order to not interfere with the *SimonSays* puzzle.
- Fixed Floor2 Room12, as it was reverted due to merge.
- Hallway6, adjusted barrel as it was clipping through a wall.
- Room10, fixed the *ButtonFrenzy* puzzle.

Week of 17/04/2023

- *LootWheel* is now faster and doesn't take as long to wait for the prize.
- *SimonSays* completely changed so now you don't have to enter a pressure pad you just have to press a button in order to start it/show the sequence.
- Added the previous *SimonSays* changes to Floor2 and Floor1
- Fixed missing material in *ButtonWall* prefab.

Week of 24/04/2023

- *SimonSays*: added rumble on correct button press
- *WhackAMole*: added rumble on correct button press
- *ButtonFrenzy*: It was previously using `Rumble()` function on correct button pressed, from another person, however, due to the TDD, I changed it to `RumblePulse()`
- *InteractionButton*: Changed from `Rumble()` to `RumblePulse()` due to same reason as above.
- *SimonSays* in Floor1 and Floor2, changed the button interaction text to align with the puzzle, changing it to "Show Sequence".
- Floor1 Jail, added lighting to showcase where the *SimonSays* puzzle is as it is very dark.
- Floor2 Room2, fixed the gangway buttons as there was no audio implemented in the inspector, so added audio to the button presses.
- Floor2 Room2, fixed flag not dropping down on third gangway button press.

Week of 01/05/2023

- *WhackAMole*: puzzle now doesn't require player to stand on a layermask/pressure pad, it runs all the time. This is due to a bug where the layermask would not work on certain players/team.
- *SpinWheel*: Made it so that the wheel doesn't spin for as long.
- *RiddleDoor*: Fixed bug where there was still an invisible collider on the interactables that you can spin, even once completed the door.
- Floor2Room9, Floor1Room5A, Floor1Room5B: Fixed soft lock issues by adjusting prop rotations and positions. As there were issues where players could jump into a specific section and not be able to get out of there.
- Changed a lot of environment objects layers from *Default* to *Environment* as there were issues where because you aren't able to jump on top of those objects, it would be sometimes very slow to get out of it.