**Introduction:** SYM DET is an immersive detective adventure where you embark on the trail of a mysterious murder. Joe, a police officer, is assigned to California to unravel the mystery behind his sister's death. However, Joe's journey is plagued by a formidable obstacle – the presence of "False Witnesses." As he battles his own inner demons, Joe must navigate through a web of deceptive statements and misleading information provided by these false witnesses. With bouts of attacks that blur the line between reality and hallucinations, Joe's ability to separate truth from lies becomes paramount. He must carefully analyze their statements, uncovering the hidden clues amidst the deceptive web in order to access crucial evidence and follow the correct leads. In his quest for the truth, Joe must maintain control over himself while thoroughly investigating every corner to unearth the secrets behind the murder.

Incorporating the obstacle within the introduction adds depth to the storyline, highlighting the challenges Joe faces and emphasizing the importance of careful analysis and discernment throughout the game.

**SYM DET Documentation**

**Overview:**

SYM DET is an immersive detective adventure game where players take on the role of Joe, a police officer, as he unravels the mystery behind his sister's tragic death. The game combines elements of deduction, puzzle-solving, and exploration in a gripping storyline filled with suspense and intrigue.

**Key Features:**

* Engaging Storyline: Immerse yourself in a captivating narrative as you follow Joe's journey to uncover the truth behind his sister's death.
* Atmospheric Setting: Explore meticulously crafted environments that reflect the dark and mysterious atmosphere of the game.
* Challenging Obstacles: Encounter deceptive false witnesses and solve intricate puzzles to gather crucial evidence and progress in the investigation.
* Psychological Twist: Navigate Joe's inner turmoil as he battles his own demons, with bouts of attacks that blur the line between reality and hallucinations.
* Intuitive Controls: Seamlessly control Joe's movements and interactions using keyboard and mouse inputs.

**Gameplay:**

* Investigate Crime Scenes: Search for clues, examine evidence, and reconstruct the crime scene to piece together the events leading to the murder.
* Interact with Characters: Engage in conversations with various characters, including suspects and witnesses, to gather information and uncover their hidden motives.
* Analyze Statements: Navigate the web of deception created by false witnesses, carefully analyze their statements, and separate truth from lies to progress in the investigation.
* Solve Puzzles: Encounter challenging puzzles that require logical thinking, observation, and deduction to unlock new leads and unveil the truth.
* Manage Joe's Mental State: Deal with Joe's psychological state and the impact it has on his ability to investigate, controlling his attacks and maintaining a clear mind.
* Progression and Upgrades: Unlock new abilities, gather evidence, and earn experience points to enhance Joe's investigative skills and unlock additional story elements.

**Game Mechanics:**

**Player Movement:**

* Players can control Joe's movement using keyboard inputs (WASD or arrow keys).
* Smooth character movement with acceleration and deceleration.

**Jumping:**

* Joe can perform standard jumps to navigate platforms and avoid obstacles.
* Controllable jump height allows for precise platforming.
* Players have control over the height of Joe's jumps, allowing for precise platforming.

**Shooting:**

* Joe can equip and use a firearm to defend himself and eliminate enemies.
* Shooting mechanics include firing, and reloading.

**Enemies and Traps:**

* Encounter various enemy types, including melee enemies and ranged enemies, each with unique attack patterns and behaviors.
* Enemies may exhibit patrol behavior, moving left and right, and attacking when they encounter Joe.

**Multiple Levels:**

* Explore different levels or stages with distinct environments and challenges.

**Camera Movement:**

* Dynamic camera movement follows Joe's character, keeping him in view at all times.
* Camera transitions smoothly between rooms or areas to provide a seamless gameplay experience.

**Health System:**

* Joe has a health bar that indicates his current health status.
* Collectible health items can be found throughout the levels to heal Joe and restore his health.

**Music and Sounds:**

* Immersive sound effects for player actions, enemy actions, traps, and background music.
* Dynamic audio enhances the atmosphere and gameplay experience.

**Checkpoint System and Respawn:**

* Progress is saved through a checkpoint system that allows players to respawn at the last reached checkpoint upon death.

**Game Over Scenario and Screen:**

* A game over condition is triggered upon the player's failure to complete a level or upon losing all health.
* A game over screen provides options such as retrying the level or returning to the main menu.

**Pause System and Menu:**

* Players can pause the game, which brings up a pause menu.
* The pause menu includes options such as adjusting sound and music volume.

**Narrative and Dialogue Mechanics:**

• A robust narrative component is managed through a unique messaging system, where messages are revealed to the player in a typewriter-like fashion.

• Player input controls the reveal speed of the text, as well as the progression to the next message, offering an interactive narrative experience.

• Once all messages or dialogues are explored in a level, the game transitions to the next stage, suggesting a narrative-driven progression.

A screenshot of a video game

Description automatically generated

A screenshot of a video game

Description automatically generated

A screen shot of a computer program

Description automatically generated with low confidence

The enemy regularly scans the area for players within a certain radius. The opponent begins a shooting timer when the player is found, and when it is ready, it fires a bullet in that direction. The opponent has ammunition, a fire rate, and sound effects when shooting. It also uses the IDamagable interface, which gives it the ability to receive and do damage. When damaged the opponent records the harm and displays a pain animation. The opponent is eliminated when its health is zero. The general enemy behavior provided by this code includes detection, shooting, and damage management.

A screen shot of a computer program

Description automatically generated with low confidence

This interface includes actions for damage events and death, properties for health and default health, a boolean property indicating whether the object is alive, a transform property for position information. It also provides a method for damage on the object, which reduces its health, triggers damage events, and checks if the object is killed. The interface also offers ways to deal with death, damage received, and damage given. The IDamagable interface standardizes the behavior and functionality related to damage and health for game objects that implement it.

A screen shot of a computer program

Description automatically generated with low confidence

Several parameters, including ammo capacity, fire rate, and shooting cooldown, are initialized. When the conditions are met, shooting is permitted after handling the rotation of the gun based on the mouse position in the Update() method. The shooting cooldown is controlled by the ShootTimer() function. The firing action is handled by the Shoot() method, which also plays sound effects, reduces ammo, and spawns projectiles with the proper velocity. Overall, the code allows a gun to rotate in the direction of the mouse position and fire shots with controlled timing and ammo consumption.

A picture containing text, screenshot, software, multimedia software

Description automatically generated

This C# script for Unity defines a class Bullet, which represents a bullet in a game. If the collided object has a component implementing the IDamagable` interface, the bullet will call the doDamage method of that component, passing in a damage amount of 36 and the IDamagable component of the bullet's owner, representing the source of the damage which is either enemy or our player. After inflicting damage, the bullet object destroys itself to prevent further collisions, thus simulating the bullet's disappearance upon impact.

A screenshot of a computer program

Description automatically generated with medium confidence

The script displays messages character-by-character at a rate specified by time\_per\_character. After a delay of 2.5 seconds from the start of the scene, the message box animation plays, and the message starts to reveal( fade in fade out). The Update method checks whether the message is still revealing and, if so, updates the text display. If the user presses the Space key, the script either finishes revealing the current message or, if the message has finished revealing, it moves to the next message or fades out the scene and loads the next level, if all messages have been revealed.