



**BINUS UNIVERSITY**

**BINUS INTERNATIONAL**

## Algorithm and Programming

### Final Project Report

**Student Information:**

**Name:** Justin Hadinata

**Student ID:** 2702298236

**Course Code:** 6047001

**Course Name:** Algorithm and Programming

**Class:** L1AC

**Lecturer:** Jude Joseph Lamug Martinez, MCS

**Type of Assignment:** Final Project report

## **A. Introduction**

### **1. Background**

Python is the first programming language I've ever learned; I remember started learning it back in 2021. I've had tons fun with it and back then I also was fond with Video Games, so one of my goals was making a game with Python. But I remember feel pressured with the coding and never actually learned the library needed to do so that is Pygame. This decision gave me an Imposter Syndrome.

So, after some consideration, in order to challenge myself and prove to myself that I can indeed do the stuff I've always wanted, I decided to make a game in Pygame (Also to remove my Imposter Syndrome Feeling). The Game I've decided is a short Visual Novel with some 2D Gameplay to it. At first, I actually wanted to use RenPy Game engine for it, but turns out RenPy uses its own programming language instead of Python (Even though the languages are very similar), So it wouldn't be fit.

### **2. Identified Problems**

There are two problems I wishes to solve by doing this Project. The first one is related to the limitations of mobility and freedom in the Visual Novel games, this limitation heavily impacted the gameplay and results in a repetitive and tedious gameplay that sadly diminish the potential of any Immersive storytelling. I'm aware that most of people that played Visual Novel already know about this fact, and actually played this genre of game for that exact reason.

But there are some groups that actually liked the idea of the genre of Visual Novel and wishes to enjoy it, but get easily appalled by the repetitive gameplay. So, by creating this Visual Novel game that gave the player more mobility and freedom, it aims to captivate, including myself, or groups of people who find the Traditional Visual Novel gameplay boring, with few aspects added to the Traditional genre. The second problem is related to the background.

The realm of programming is vast and, as a result it can be complex and difficult for certain individuals to grasp. This complexity often results in some individuals experiencing an Imposter Syndrome. This issue is also exacerbated by social medias, especially with how many talented programmers there are in the Internet. Therefore, by embarking on this project, it would inspire fellow programmers that we can do anything we aspire our mind too, and in turn also relieves our senses of Imposter Syndrome.

## **B. Project Specification**

### **1. Project Description**

The Game is called Behind Me. It's an extremely simple and short Visual Novel horror game home invasion themed with some 2D game elements to it. The game is about a little boy named Timmy who lives with his parents, and the game takes place at a cabin during a snowstorm. During the night, a stranger invades their house, and Timmy wakes up due to the sound, now you as a player has to play as Timmy and control his action. When running the code, the game starts with a simple menu, that has start button and quit button in it. Clicking the quit button would terminate the program immediately but upon clicking the start button, the game will begin. The game starts by first showing a cinematic scene of a cabin during a winter, it was on a snow storm, the scenes last roughly 5 seconds.

After the cinematic scene is over, a textbox will appear, narrating the story and situation of the game, the player can scheme through the story by pressing the Space bar just like in a visual novel. After some narration and storytelling, the player will be presented by choices and decisions making, and these choices and decision would lead into different path and stories just like in a Traditional Visual Novel. Certain decision would lead the player to a simple 2D side-scrolling mode, where they can explore Timmy's room and can go into different story path depending on an action in the room. Certain decision would lead into a game over scene where the player would be given an option to go back to the main menu, and certain decision would lead into a winning screen with the same go back option. Later in the game there also will be

an enemy whom you have to dodge. The goal of the game is to received the good ending or survive the whole ordeal.

## **2. Libraries / Modules**

- **Pygame**

Pygame is a set of Python modules that is designed for making Video Games. For this particular project, they're used for displaying the images, music, tracking player's input, responsible for the animation, etc.

- **Time**

Time is a python module that provides various time-related functions. In my case, I use the modules to add some timing in the game for functionality reason and dramatic reasons.

- **Sys**

Sys is a module that provides various functions and variables that are used to manipulate different parts of the python runtime environment. In my case, I uses them to terminate the game safely, and efficiently without causing an error or bugs.

## **3. Game Mechanics overview**

- Player can press space to navigate through the story
- If chooses to investigate, player can move left and right using A, D. Tiny jump using Y, and jump by pressing space.

## **4. Third Party Software and Source**

- **Aseprite**

Aseprite is a program to create pixel arts and animate them. Aseprite is used a lot in the project for the making of the pixel arts and the animation, like the Player and Killers sprite.

- **Canva**

Canva is a free-to-use online graphic design tool, which is heavily used in the project for the Textboxes and decision box.

- **Youtube**

Youtube are a video platform Some of the sound effects are taken from Youtube.com as its source.

## 5. Game Assets

**Textboxes (Source: Made by myself):**



## Algorithm and Programming

DAYS TURN INTO WEEKS, AND WEEKS INTO MONTHS. TIMMY'S DISAPPEARANCE REMAINS A MYSTERY. THE ONCE LIVELY CABIN HOLDS ONLY ECHOES OF LAUGHTER, NOW REPLACED BY AN EERIE SILENCE. THE SMALL TOWN WHISPERS OF AN UNSOLVED CASE—A LITTLE BOY WHO VANISHED WITHOUT A TRACE.

WITH A MIX OF FEAR AND CURIOSITY, TIMMY DECIDES TO INVESTIGATE THE SOURCE OF THE STRANGE NOISE. HE SLIPS OUT OF BED, FEET TOUCHING THE COLD FLOOR. IN THE DIM MOONLIGHT, HE TIPTOES AROUND HIS ROOM, GUIDED BY THE MYSTERIOUS SOUND.

TIMMY FINDS A BLUE PHONE IN THE ROOM. CLICKING ON IT MIGHT REVEAL IMPORTANT INFORMATION OR PROVIDE A WAY TO NAVIGATE THE MYSTERIOUS EVENTS IN THE CABIN.

TIMMY, FEELING A SURGE OF FEAR, DECIDES TO USE THE BLUE PHONE TO CALL FOR HELP. THE POLICE ANSWER, AND TIMMY EXPLAINS THE SITUATION.

AS TIMMY WAITS ANXIOUSLY, HE HEARS DISTANT SIRENS GROWING LOUDER. THE POLICE BURST INTO THE HOUSE, VOICES SHOUTING COMMANDS. TIMMY, RELIEVED, STAYS HIDDEN AS THE AUTHORITIES TAKE CONTROL.

FROM HIS HIDING SPOT, TIMMY HEARS THE COMMOTION AS THE POLICE APPREHEND THE STRANGER THAT CAUSES THE NOISE. THE THREAT IS NEUTRALIZED, AND A SENSE OF SAFETY RETURNS TO THE CABIN.

WITH THE STRANGER BEHIND BARS, TIMMY'S FAMILY IS SAFE. THE POLICE ASSURE THEM, AND THE ONCE TENSE ATMOSPHERE LIFTS. THE ORDEAL IS OVER, AND THEY CAN NOW LIVE HAPPILY EVER AFTER.

DO YOU WANT TO GO  
DOWNSTAIRS?

DESCENDING THE STAIRS, TIMMY FINDS THE FRONT DOOR SLIGHTLY OPEN. THE MYSTERIOUS NOISE GROWS LOUDER.

FEELING A SENSE OF URGENCY, TIMMY DECIDES TO LEAVE THE HOUSE. HE STEPS INTO THE BITING COLD, THE SNOW-COVERED LANDSCAPE STRETCHING BEFORE HIM.

## Algorithm and Programming

TIMMY, DETERMINED, TRUDGES THROUGH THE SNOWY TERRAIN, SEEKING HELP. THE WIND HOWLS, AND THE SNOWFLAKES OBSCURE HIS VISION, MAKING THE JOURNEY MORE CHALLENGING.

AS TIMMY BATTLES THE HARSH ELEMENTS, THE COLD BECOMES TOO MUCH TO BEAR. EXHAUSTED AND FREEZING, HE COLLAPSES IN THE SNOW. THE UNFORGIVING WINTER CLAIMS ANOTHER VICTIM, AND TIMMY'S JOURNEY ENDS TRAGICALLY.

IGNORING THE OPTION TO LEAVE, TIMMY BRAVELY DECIDES TO INVESTIGATE THE MYSTERIOUS NOISE. THE DARKNESS SEEMS TO ENVELOP HIM AS HE VENTURES FURTHER INTO THE HOUSE, GUIDED ONLY BY THE UNSETTLING SOUND.

FOLLOWING THE NOISE LEADS TIMMY TO A LOCKED ROOM. THE MUFFLED SOUNDS PERSIST, RAISING TENSION

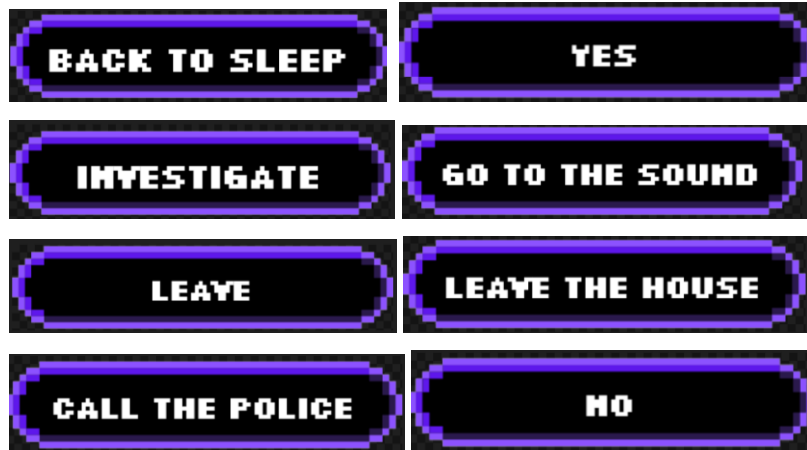
OPTING TO UNLOCK THE DOOR, TIMMY DISCOVERS THE SOURCE OF THE NOISE—AN INTRUDER RUMMAGING THROUGH THE FAMILY'S BELONGINGS. SHOCK AND FEAR GRIP HIM AS THE INTRUDER BECOMES AWARE OF TIMMY'S PRESENCE.

TIMMY, QUICK ON HIS FEET, MANAGES TO DODGE THE ADVANCING KILLER. THE INTRUDER STUMBLES FORWARD, CRASHING INTO A SOLID OBJECT, RENDERING THEM UNCONSCIOUS.

WITH THE STRANGER INCAPACITATED, TIMMY SEIZES THE OPPORTUNITY TO CALL THE POLICE. SIRENS WAIL IN THE DISTANCE AS AUTHORITIES ARRIVE TO HANDLE THE SITUATION. THE DANGER IS AVERTED, AND TIMMY, THOUGH SHAKEN, SADLY THE KILLER HAS ALREADY KILLED THE DOG.

DESPITE TIMMY'S EFFORTS, THE KILLER'S STRIKE LANDS. THE ROOM ECHOES WITH THE SOUND OF THE IMPACT AS TIMMY CRUMPLES TO THE FLOOR.

IN THE FINAL MOMENTS OF CONSCIOUSNESS, TIMMY'S VISION BLURS. THE INTRUDER, HAVING SUCCEEDED, LEAVES THE ROOM. THE GAME CONCLUDES WITH A TRAGIC ENDING AS TIMMY'S JOURNEY COMES TO A DEVASTATING END.



### 2D Game Assets:

#### Player (Source: Made by myself):

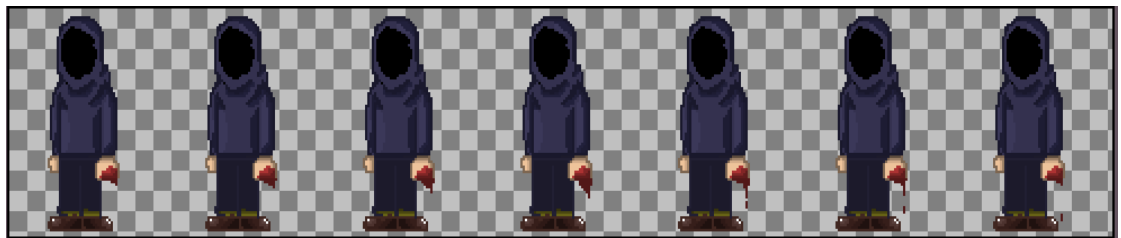
PlayerJUMP.png

Player1.png

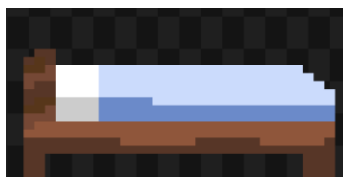
player2.png



#### Killer (Source: Made by myself):



#### Bed, Ground, and Table (Source: Made by myself):



#### Phone (Source: )





**Intro Snowing Cinematic Scene Gif (Source:**

**<https://www.deviantart.com/ccelestiiall/art/Cabin-in-the-Snow-723308283>)**



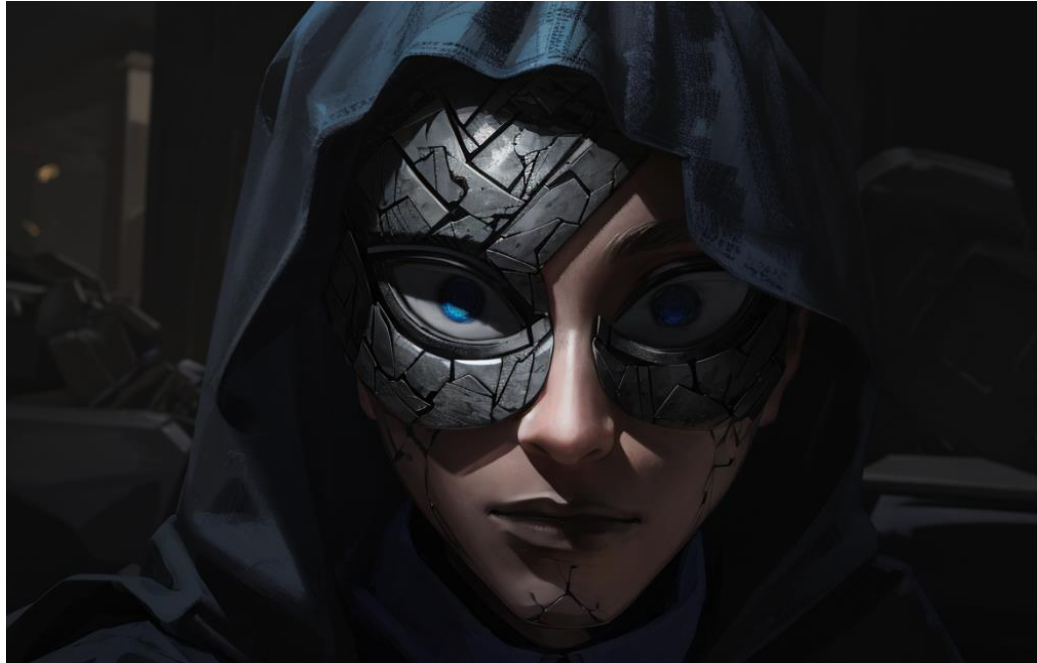
**Game Title (Source: Made by Myself)**



**Home Screen (Source: From Myself, Using AI)**



**Killer Screen (Source: From Myself, Using AI)**



## B. Algorithm

For my Project, I only have one Python file, that is named Main.py:

### The Essential & Main Algorithm

Main.py

```
import pygame
from sys import exit
import time
```

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

```
class Player(pygame.sprite.Sprite): ...
```

We define a Class called Player using the Pygame OOP (Sprite)

We can called the Pygame Sprite through the syntax -> pygame.sprite.Sprite inside the class argument.

```
def __init__(self):
    super().__init__()
    Player_standing_1 = pygame.image.load('player/Player1.png').convert_alpha()
    Player_standing_2 = pygame.image.load('player/Player2.png').convert_alpha()
```

We first use the `pygame.image.load` syntax to display the images for our player, because our player has an Animation of 2 frame, we load 2 images and store them in a variable. They're in the `def __init__(self):` because, this module is where we want to load all of our assets, and `super()` so we can use the variable in any other modules

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.

The code starts by Importing the the Pygame, Sys and getting , and Time Module.