A

Dissertation on

**“GHOSTPLAY”**

By

**Mr. Roshan Pandurang Chothe**

**Seat Number: 183417**

Under the Guidance of

**Asst. Prof. Vijay Rawool**



**Department of Information Technology**

**VIKAS COLLEGE OF ARTS, SCIENCE AND COMMERCE, Vikhroli (East)**

**Mumbai University [2018-2019]**

**CERTIFICATE**

**VIKAS COLLEGE OF ARTS, SCIENCE AND COMMERCE**

**MUMBAI**

**MAHARASHTRA**



This is to certify that the project entitled, **" GHOSTPLAY** ", is bonafied work of **Roshan Pandurang Chothe** bearing Seat.No: (**183417**) submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai

**Internal Guide**  **Coordinator**

**External Examiner**

**Date:**  **College Seal**

**PROFORMA FOR THE APPROVAL PROJECT PROPOSAL**

**PRN No.: …………………… Roll no: \_\_\_\_\_\_\_\_\_\_\_**

**1. Name of the Student *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**2. Title of the Project *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**3. Name of the Guide *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**4. Teaching experience of the Guide *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**5. Is this your first submission? Yes No**

**Signature of the Student Signature of the Guide**

**Date: ………………… Date: …………………….**

**Signature of the Coordinator**

**Date: …………………**

**ACKNOWLEGMENT**

I take this opportunity to thank all those who have contributed in successful completion of this dissertation. I would like to express my sincere thanks to my guide and project co-coordinator Prof. Seema miss who has encouraged me to work on this topic and valuable guidance wherever required. I wish to express thanks to all my friends who helped me directly or indirectly in preparation of this project.

**Mr. Roshan Chothe**

**Abstract**

Now a days the Online Gamers have the problem of staying AFK in a game.

Now, **what does AFK really means?**AFK means to stay away from the keyboard and to convey to your co-player that I have leave the game or I will stay Idle for some while and should not depend on me. So due to this team might lose the game or the player might get banned which is no considerable.

To solve this Problem, we have “**GHOSTPLAY**” which will save gamers from getting banned or losing the game.

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **Sr.**  **No.** | **TOPIC** |
| **01.** | **INTRODUCTION** |
| **02.** | **1.1. Background** |
| **03.** | **1.2. Objectives** |
| **04.** | **1.3. Purpose, Scope, and Applicability** |
| **05.** | **1.3.1. Purpose** |
| **06.** | **1.3.2. Scope** |
| **07.** | **1.3.3. Applicability** |
| **08.** | **1.4. Achievements** |
| **09.** | **1.5. Organization of Report** |
| **10.** | **SURVEY OF TECHNOLOGIES** |
| **11.** | **REQUIREMENTS AND ANALYSIS** |
| **12.** | **3.1. Problem Definition** |
| **13.** | **3.2. Requirements Specification** |
| **14.** | **3.3. Planning and Scheduling** |
| **15.** | **3.4. Software and Hardware Requirements** |
| **16.** | **3.5. Preliminary Product Description** |
| **17.** | **3.6. Conceptual Models** |
| **18.** | **SYSTEM DESIGN** |
| **19.** | **4.1. Basic Modules** |
| **20.** | **4.2. Data Design** |
| **21.** | **4.2.1. Schema Design** |
| **22.** | **4.2.2. Data Integrity and Constraints** |
| **23.** | **4.3. Procedural Design** |
| **24.** | **4.3.1. Logic Diagrams** |
| **25.** | **4.3.2. Data Structures** |
| **26.** | **4.3.3. Algorithms Design** |
| **27.** | **4.4. User interface design** |
| **28.** | **4.5. Security Issues** |
| **29.** | **4.6. Test Cases Design** |

**INTRODUCTION**

* **Background:-**

The Project entitled “**GHOSTPLAY**”. The Main Aim of the project is to control 3D games without any Human Interaction. The Software is best in Use for Gamers. The Software uses the Image Analyzing Techniques to determine and interact with the In-game Environment to find a path out of that situation. As the name suggest “GHOSTPLAY”

It’s like Ghost is playing the Game. It helps Gamers to control the game when they Have to do some work but can’t leave the game. In this situation “GHOSTPLAY” is used.

* **Purpose of the project:-**

The Software uses the Image Analyzing Data stored in the database to help the Program Play the game more Accurate. The User can help the Agent By sending Command from there Android Phone. This Software also provide a1n Advance Personal Assistant that can help Players with the App and can also have conversation as it also works as a chat-bot.

* **Scope of the Project :-**

The Scope of the project is to establish requirements for Desktop Gaming. The main Scope of the Project is to implement**AI** to find as many as possible outcomes to determine the Difficult of the situation and act on the environment by find a perfect way to solve it.

This Software is mostly use my Gamers and Game tester to test the level in there game to check the difficulty of the level.

* **Application:-**

Application defines where the proposed system can be used in our society. Some of them are listed below:-

* **SAY NO TO AFK’s**

This means that by using **GHOSTPLAY** No one will stay AFK every. Nowadays Online games rules are strict and AFK are not tolerated so the player may get banned fora while.

* **TESTING LEVEL IN GAME**

This Feature will help gaming companies to test their game’s level to check ifthe specific level is too hard or too easy to play. And it will also help with finding the bug in the level

* **Video Game Combat AI**

Many contemporary video games fall under the category of action, first person shooter, or adventure. In most of these types of games there is some level of combat that takes place. The AI's ability to be efficient in combat is important in these genres. A common goal today is to make the AI more human, or at least appear so.

* **Uses in games beyond NPCs**

AI developments may play roles in game AI beyond the traditional paradigm of AI controlling NPC behavior.

Player-experience modeling: Discerning the ability and emotional state of the player, so as to tailor the game appropriately. This can include [**dynamic game difficultybalancing**](https://en.wikipedia.org/wiki/Dynamic_game_difficulty_balancing)**,**which consists in adjusting the difficulty in a video game in real-time based on the player's ability.

* **MUD’s(Multiplayer Real-Time Virtual World)**

Players may run bots to automate laborious tasks: this activity can sometimes make up the bulk of the gameplay. While a prohibited practice in most MUDs, there is an incentive for the player to save his/her time while the bot accumulates resources, such as [**experience**](https://en.wikipedia.org/wiki/Experience_point), for the [**player character**](https://en.wikipedia.org/wiki/Player_character).

* **Dynamic Bot**

Bots dynamically learn the levels and maps as they play. Real Bot, for *Counter-Strike*, is an example. Some bots are designed using both static and dynamic features.

**SURVEY OF TECHNOLOGY**



**PyCharm: -**

**PyCharm** is an [**integrateddevelopment environment**](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [**computerprogramming**](https://en.wikipedia.org/wiki/Computer_programming), specifically for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) language. It is developed by the Czech company **[JetBrains](https://en.wikipedia.org/wiki/JetBrains" \o "JetBrains)**.It provides code analysis, a graphical debugger, an integrated unit tester, integration with [versioncontrolsystems](https://en.wikipedia.org/wiki/Revision_control), and supports web development with [Django](https://en.wikipedia.org/wiki/Django_(web_framework)).

**Features: -**

1. Coding assistance and [**analysis**](https://en.wikipedia.org/wiki/Code_analysis), with [codecompletion](https://en.wikipedia.org/wiki/Autocomplete), syntax and error highlighting, linter integration, and quick fixes

2. Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages

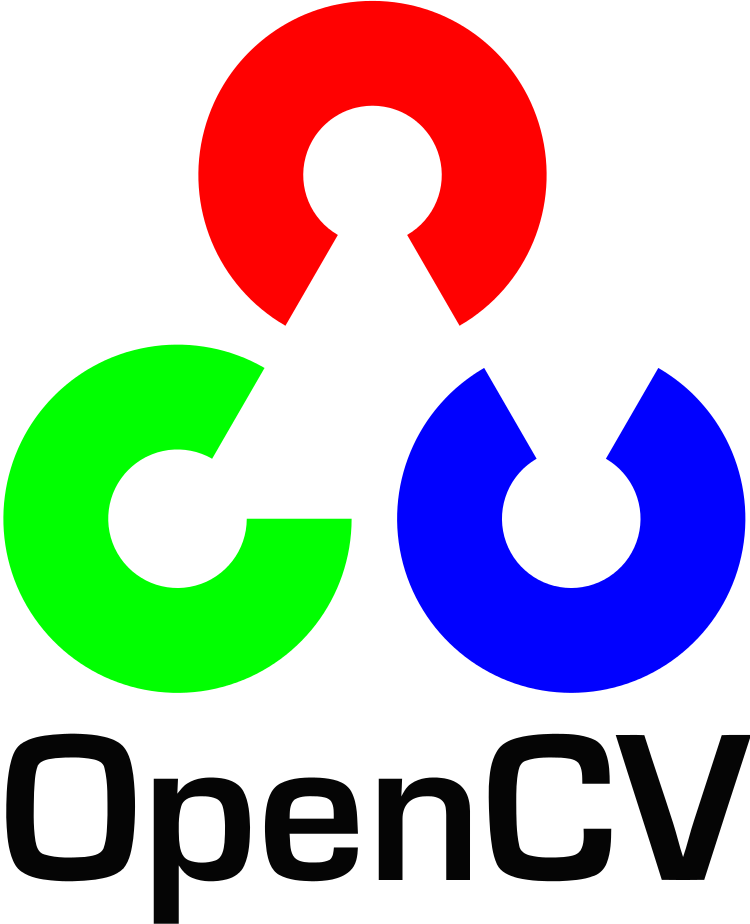
3. Python [refactoring](https://en.wikipedia.org/wiki/Refactoring): including rename, extract method, introduce variable, introduce constant, and pull up, pushdown and others

4. Support for web framework: [**Django**](https://en.wikipedia.org/wiki/Django_(web_framework)), [**web2py**](https://en.wikipedia.org/wiki/Web2py) and [**Flask**](https://en.wikipedia.org/wiki/Flask_(web_framework))

5. Integrated Python [debugger](https://en.wikipedia.org/wiki/Debugger)

6. Integrated [unit testing](https://en.wikipedia.org/wiki/Unit_testing), with line-by-line code coverage

7. [**Google App Engine**](https://en.wikipedia.org/wiki/Google_App_Engine) Python development



**OpenCV: -**

**OpenCV** (*Open Source Computer Vision*) is a [**library of programming functions**](https://en.wikipedia.org/wiki/Library_(computing)) mainly aimed at real-time vision. Originally developed by [Intel](https://en.wikipedia.org/wiki/Intel_Corporation), it was later supported by [Willow Garage](https://en.wikipedia.org/wiki/Willow_Garage) then Itzel (which was later acquired by Intel). The library is [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) and free for use under the [open-source](https://en.wikipedia.org/wiki/Open-source) [BSD license](https://en.wikipedia.org/wiki/BSD_license).

**Application:-**

1. 2D and 3D feature toolkits.
2. Motion understanding
3. Object identification
4. Motion tracking
5. [Augmented reality](https://en.wikipedia.org/wiki/Augmented_reality)
6. [Structure from motion](https://en.wikipedia.org/wiki/Structure_from_motion) (SFM)
7. [Facial recognition system](https://en.wikipedia.org/wiki/Facial_recognition_system)

**Programming Language:-**

OpenCV is written in [**C++**](https://en.wikipedia.org/wiki/C%2B%2B) and its primary interface is In C++, but it still retains a less comprehensive though extensive older [**C interface**](https://en.wikipedia.org/wiki/C_(programming_language)). There are bindings in [**Python**](https://en.wikipedia.org/wiki/Python_(programming_language)), [**Java**](https://en.wikipedia.org/wiki/Java_(programming_language)) and [**MATLAB**](https://en.wikipedia.org/wiki/MATLAB)/[**OCTAVE**](https://en.wikipedia.org/wiki/GNU_Octave).

The API for these interfaces can be found in the online documentation. Wrappers in other languages such as [**C#**](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)), [**Perl**](https://en.wikipedia.org/wiki/Perl), **Haskell** and [**Ruby**](https://en.wikipedia.org/wiki/Ruby_(programming_language)) have been developed to encourage adoption by a wider audience.

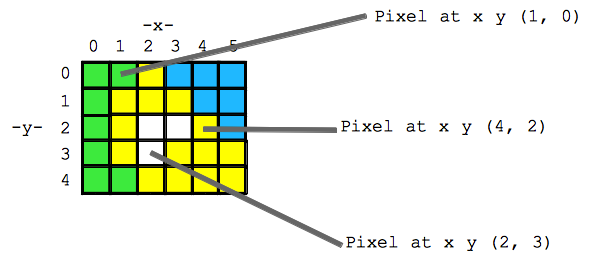
All of the new developments and algorithms in OpenCV are now developed in the C++ interface.

# **Image Processing with OpenCV: -**

Visual information is the most important type of information perceived, processed and interpreted by the human brain. Image processing is a method to perform some operations on an image, in order to extract some useful information from it. An image is nothing more than a two dimensional matrix (3-D in case of colored images) which is defined by the mathematical function f(x, y) where x and y are the two co- ordinates horizontally and vertically. The value of f(x, y) at any point is gives the pixel value at that point of an image, the pixel value describes how bright that pixel is, and/or what color it should be.

For grayscale images the pixel value is a single number that represents the brightness of that pixel, the most common pixel format is the byte image, which is stored as an 8-bit integer giving a range of possible values from 0 to 255. As a convention is taken to be black, and 255 is taken to be white the values in between make up the different shades of gray

To represent color images, separate red, green and blue components must be specified for each pixel (assuming a RGB color model), and so the pixel `value’ becomes a vector of three numbers. Often the three different components are stored as three separate grayscale images known as color planes (one for each of red, green and blue), which have to be recombined when displaying or processing.



Now allow me to introduce the color models formally as follows, a color model is an abstract mathematical model describing the way colors can be represented as tuples of numbers, typically as three or four values or color components. When this model is associated with a precise description of how the components are to be interpreted (viewing conditions, etc.) the resulting set of colors is called color space.

**There are four aspect of Artificial Intelligence**: -

1. Thinking Humanly
2. Thinking Rationally
3. Acting Humanly
4. Acting Rationally

**1**. **Thinking Humanly: -**

Thinking humanly means trying to understand and model how the human mind works. There are (at least) two possible routes that humans use to find the answer to a question: – We reason about it to find the answer. This is called “introspection”. – We conduct experiments to find the answer, drawing upon scientific techniques to conduct controlled experiments and measure change. The field of Cognitive Science focuses on modeling how people think.

**2. Thinking Rationally: -**

Trying to understand how we actually think is one route to AI. But another approach is to model how we should think. The “thinking rationally” approach to AI uses symbolic logic to capture the laws of rational thought as symbols that can be manipulated. Reasoning involves manipulating the symbols according to well-defined rules, kind of like algebra. The result is an idealized model of human reasoning. This approach is attractive to theorists, i.e., modeling how humans should think and reason in an ideal world.

**3. Acting Humanly: -**

Another famous test is called the “Chinese Room” which was proposed by John Searle in a paper published in 1980. Suppose you have a computer in a room that reads Chinese characters as input, follows a program and outputs (other) Chinese characters. Suppose this computer does this so well that it passes the Turing Test (convinces a human Chinese speaker that it is talking to another human Chinese speaker). Does the computer understand Chinese? Suppose Searle is in the room, and he uses a dictionary to translate the input characters from Chinese to English; he then constructs his answer to the question, translates that back into Chinese and delivers the output—does Searle understand Chinese? Of course not. This is Searle’s argument: the computer doesn’t understand it either, because all it is doing is translating words (symbols) from one language (representation) to another.

**4. Acting Rationally: -**

Acting rationally means acting to achieve one’s goals, given one’s beliefs or understanding about the world. An agent is a system that perceives an environment and acts within that environment. An intelligent agent is one that acts rationally with respect to its goals. For example, an agent that is designed to play a game should make moves that increase its chances of winning the game.

When constructing an intelligent agent, emphasis shifts from designing the theoretically best decision-making procedure to designing the best decision-making procedure possible within the circumstances in which the agent is acting.

Logical approaches may be used to help find the best action, but there are also other approaches.

Achieving so-called “perfect rationality”, making ~~the~~ best decision theoretically possible, is not usually possible due to limited resources in a real environment (e.g., time, memory, computational power, uncertainty, etc.).

The trick is to do the best with the information and resources you have. This represents a shift in the field of AI from optimizing (early AI) to satisfying (more recent AI).

**Task Environmentsin Project: -**

1. Partially Observable: -In any FPS Game Environment the Environment is Not Fully Observable as the agent or the player cannot identify the next Level or object. They can observe

Current Environment.

1. Single Agent& Multi Agent : -FPS games can works as both Single Agent & Multi Agent it depend on the way it is played

Some people play with bot which is single agent Environment and in Multiplayer Mode it acts as Multi Agent.

1. Stochastic: -If the next state of the environment cannot be determined by the current state and the action is executed by the agent then the environment is stochastic. Games like Counter Strike have Stochastic.
2. Sequential: -In a Sequential Environment the agent experienceaconnection of Sequential Event. All Games have Sequential Environment as the action performed now affect the next and further event.
3. Dynamic :- If the environment can change while the agent is deliberating then the environment is dynamic for the agent . All 3D games have Dynamic Environment.
4. Discrete: - FPs games have discrete environment has as they can be played for finite time.
5. Known: - This distinction is strictly refer to the agent state of knowledge about the laws of physics. As the FPS games have Known Environment.

**REQUIREMENT AND ANALYSIS**

**PROBLEM DEFINATION**:

In many Multiplayer Games you don’t have the option of pausing the game which gives the player only two option either Complete the Game or to leave the Game in Middle. If player chose to stay AFK (Away From Keyboard) the other player may report the player and then player may also get banned. Which can affect the Player Achievements in the game. So it’s Important for Player to someone play his game when he is AFK. That’s why we are making GhostPlay software to overcome this issue. GhostPlay has ability to handle your game when you are away from the keyboard. It can identify game status and do operations based on the current situation in Real-time.

**HAREWARE REQUIREMENTS:**

Processor: Any Processor above 2Ghz (Intel Core i3 4th gen)

Memory: 4gb ram and Above

Hard Disk: 4gb or more

Input device: Standard Keyboard and Mouse

Output device: VGA and High-Resolution Monitor

**SOFTWARE REQUIREMENTS:**

Operating System: Windows Operating System (Windows 8.1 and above)

Technologies: Python3, OpenCV Library, Android.

Database: SQLite

**OPERATING SYSTEM DESCRIPTION:**

**Microsoft Windows:**

Windows is Microsoft's flagship operating System (OS), the de facto standard for home and business computers.The graphical user interface(GUI)-based OS was introduced in 1985 and has been released in many versions since then, as described below. Microsoft got its start with the partnership of Bill Gates and Paul Allen in 1975.

Windows 10 is a series of personal computer operating system produced by Microsoft as part of its Windows NT family of operating systems. It is the successor to Windows 8.1, and was released to manufacturing on July 15, 2015

**Features of Windows:**

* **User interface and desktop**

Windows Provides Graphical User Interface which is very user friendly. The major advantage of GUIs is that they make computer operation more intuitive, and thus easier to learn and use. GUIs generally provide users with immediate, visual feedback about the effect of each action. GUI allows multiple programs and/or instances to be displayed simultaneously.

### **System security**

### Windows 10 incorporates multi-factor authenticator technology based upon standards developed by the FIDO Alliance. The operating system includes improved support for biometric authentication through the Windows Hello platform. Devices with supported cameras allow users to log in with iris or face recognition, Devices with supported readers allow users to log in through fingerprint recognition. Credentials are stored locally and protected using asymmetric **encryption.**

### 

### **Command line**

Command Prompt, also known as cmd.exe or cmd, is the command-line interpreter on Windows. Command Prompt interacts with the user through a command line interface. Command Prompt may take advantage of features available to native programs of its own platform.

### **Multimedia and gaming**

Windows support many different graphical libraries. That’s why windows is very efficient to watch videos, listening music, playing games.

**Why we user Windows Operating System?**

Overall, Windows is the best for gaming. Games are optimized for Windows and their hardware, the software (like Steam) is made for Windows, and the hardware can be tweaked to improve the performance. The software Ghostplay is related to games. Thus, we need to choose windows OS.

**Planning and Scheduling:-**

**Gantt chart:-**

A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time.

On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar, the position and length of reflects the start date, duration and end date of the activity.

This allows you to see at a glance:

What the various activities are?

When each activity begins and ends?

How long each activity is scheduled to last?

Where activities overlap with other activities, and by how much?

The start and end date of the whole project



**PERT Chart:-**

A PERT chart is a project management tool that provides a graphical representation of a project's timeline. PERT, or Program Evaluation Review Technique, breaks down the individual tasks of a project for analysis. Although PERT charts are preferable to Gantt charts because they identify task dependencies, PERT charts are often more difficult to interpret.

Interpreting PERT Charts, A PERT chart is a visual representation of a series of events that must occur within a project’s lifetime. The direction of arrows indicates the flow and sequence of events required for project completion. Dotted activity lines represent dummy activities, which are items located on another PERT path. Numbers and time allotments are assigned and shown inside each vector.

**PERT CHART of GHOSTPLAY**

|  |  |
| --- | --- |
| Scope Details | |
| Start Date | End Date |
| 28-07-2018 | 18-07-2018 |

|  |  |
| --- | --- |
| Problem Analysis | |
| Start Date | End Date |
| 19-08-2018 | 14-09-2018 |

|  |  |
| --- | --- |
| Requirement Analysis | |
| Start Date | End Date |
| 15-09-2018 | 09-10-2018 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| System Architecture | |  | Design | |
| Start Date | End Date | Start Date | End Date |
| 13-11-2018 | 17-12-2018 | 10-10-2018 | 12-11-2018 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Physical Design | |  | Coding | |
| Start Date | End Date | Start Date | End Date |
| 18-12-18 | 24-01-18 | |  |  | | --- | --- | | |  | | --- | | 25-01-18 | | | 7-03-2018 |

|  |  |
| --- | --- |
| Testing | |
| Start Date | End Date |
| 30-04-2018 | |  | | --- | | 3-04-2018 | |

|  |  |
| --- | --- |
| Implementation | |
| Start Date | End Date |
| 08-03-2018 | 29-03-2018 |

**PRELIMINARY PRODUCT DESCRIPTION:**

The GhostPlay will allows user to play games automatically when he or she is not able to play. GhostPlay will help to completing achievements, holding position on particular game situation and to play game for particular period of time.

GhostPlay will automatically detects enemies, Player health, weapon status, and acts differently on different scenarios.

GhostPlay provides users different types of modes to play. In Which user can chose by his preferences.

**SYSTEM DESIGN**

**Basic Modules:**

Our project is divided into several different basic modules. This are as follows,

1. GUI
2. Simple Navigation
3. Advanced Navigation
4. Weapon System Control
5. Enemy Detection
6. GhostAssist Android Remote
7. Mira Voice Assistant

* **GUI**: Graphical User Interface is the first module we saw in our application. It provides user the way to manage different types of settings. User have options to start GhostPlay when the user login into Windows. User can manage the training data which is collected while user plays game with GhostPlay in training mode. User can manage different Hotkeys which used in game to start different modes of GhostPlay. There is Connectivity tab in which user can connect their GhostAssist (Android App). The Credits tab shows some information about developers.
* **Simple Navigation**: In the game, When start any mode from GhostPlay, the AI will start to move in his appropriate direction. Simple Navigation will navigate Player where it detects the desirable path. It also tries to check is the movement of AI got stuck in some situation and try to solve that situation.
* **Advanced Navigation**: Advanced Navigation is based on Training Mode of GhostPlay. GhostPlay Training Mode can used by the users to give Application freedom to collect data from game environment. When Training mode starts, it collects different data from game as images and store it in particular directory in proper sorted manner on Hard Drive. The advantage if Advanced Navigation is the AI can find desirable path more easily with the help of stored Data.
* **Weapon System Control**: The important aspect of games is the weapon choice. Player Strategy is mostly based on what weapon is choose on what situation. For eg. Reload a particular weapon on safe area, change weapon to secondary weapon on the front of enemies to skip weapon reloading. These things can be handles by Weapon System Control.
* **Enemy Detection**: Enemy Detection takes place in game. It detects enemies using colors and give coordinates of screen where the enemy situated at. Then AI performs further logic to fire weapon on enemy and tries to kill him.
* **GhostAssist Android Remote**: This Android Application will help user to give GhostPlay AI command even when user away from computer with the help of Android device and Wi-Fi connection. User can give different command over the WIFI to handle the behavior of AI.
* **Mira Voice Assistant**: Mira voice assistant is the chatbot which can give commands to GhostPlay with the help of chats. Mira Assistant use Google API to recognize words of voice and Assistant’s logic will decide which command needs to send to GhostPlay.

**Data Design:**

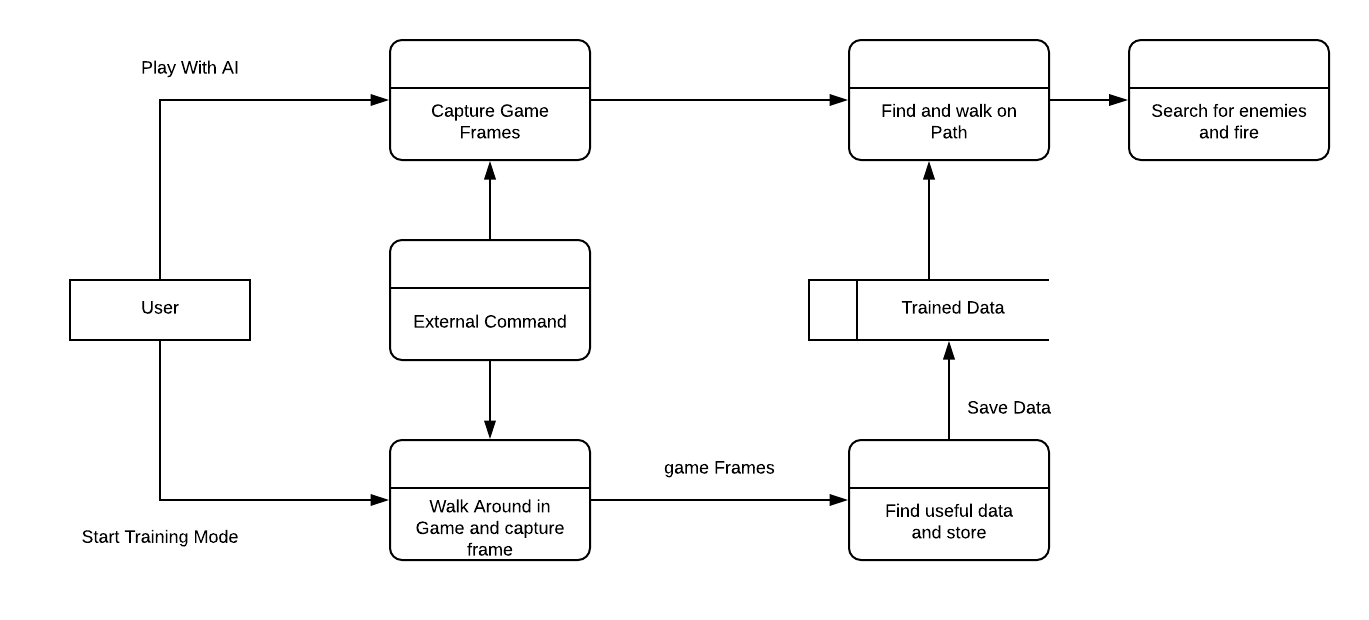
The Flow of data is simple in our Application which is shown below in Data Flow Diagram. 

Figure 1: Data Flow Diagram

**Logic Diagrams:**

The flow of our program is shown in following Control Flow Diagram.

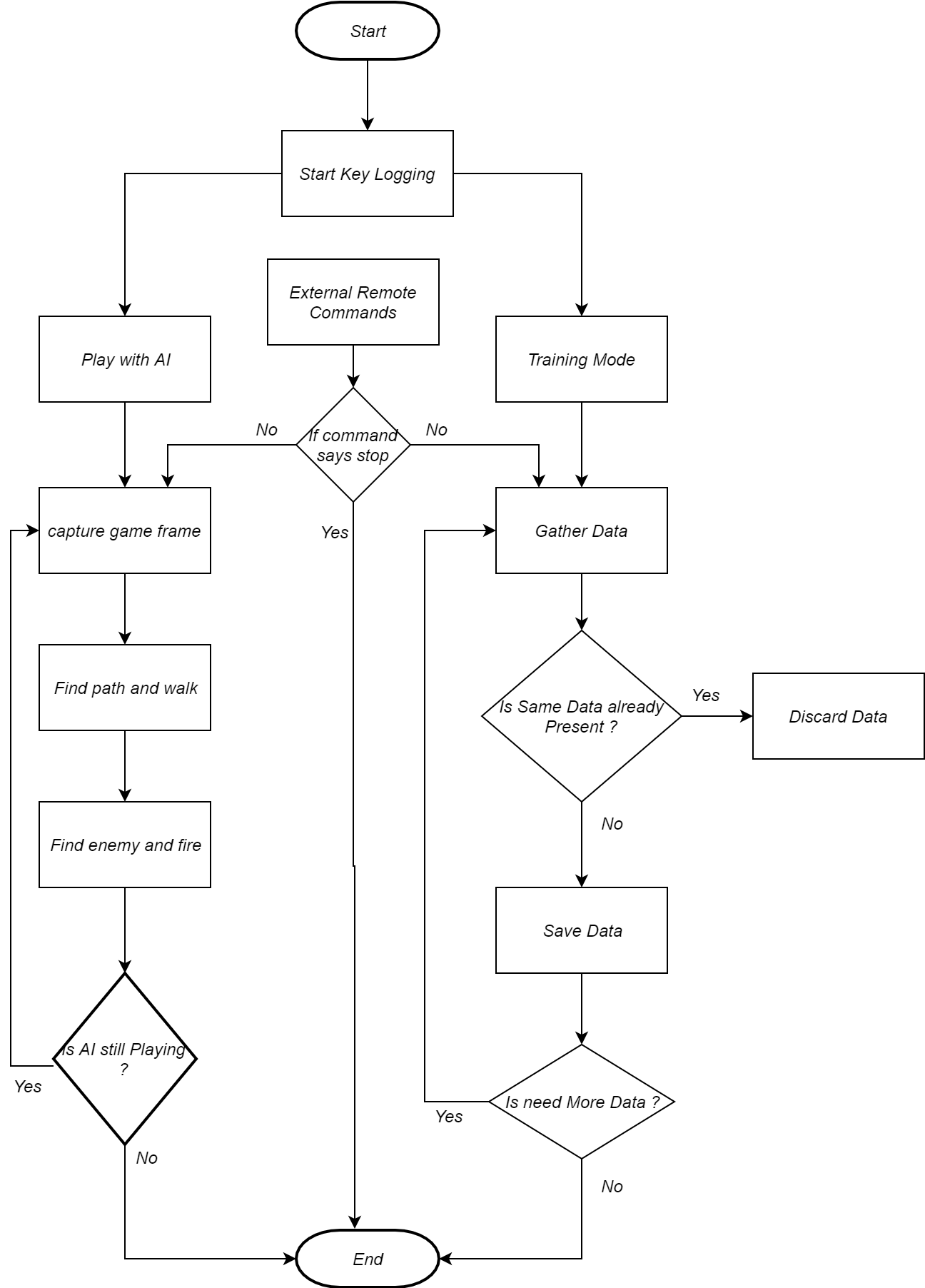


Figure 2:Control Flow Diagram

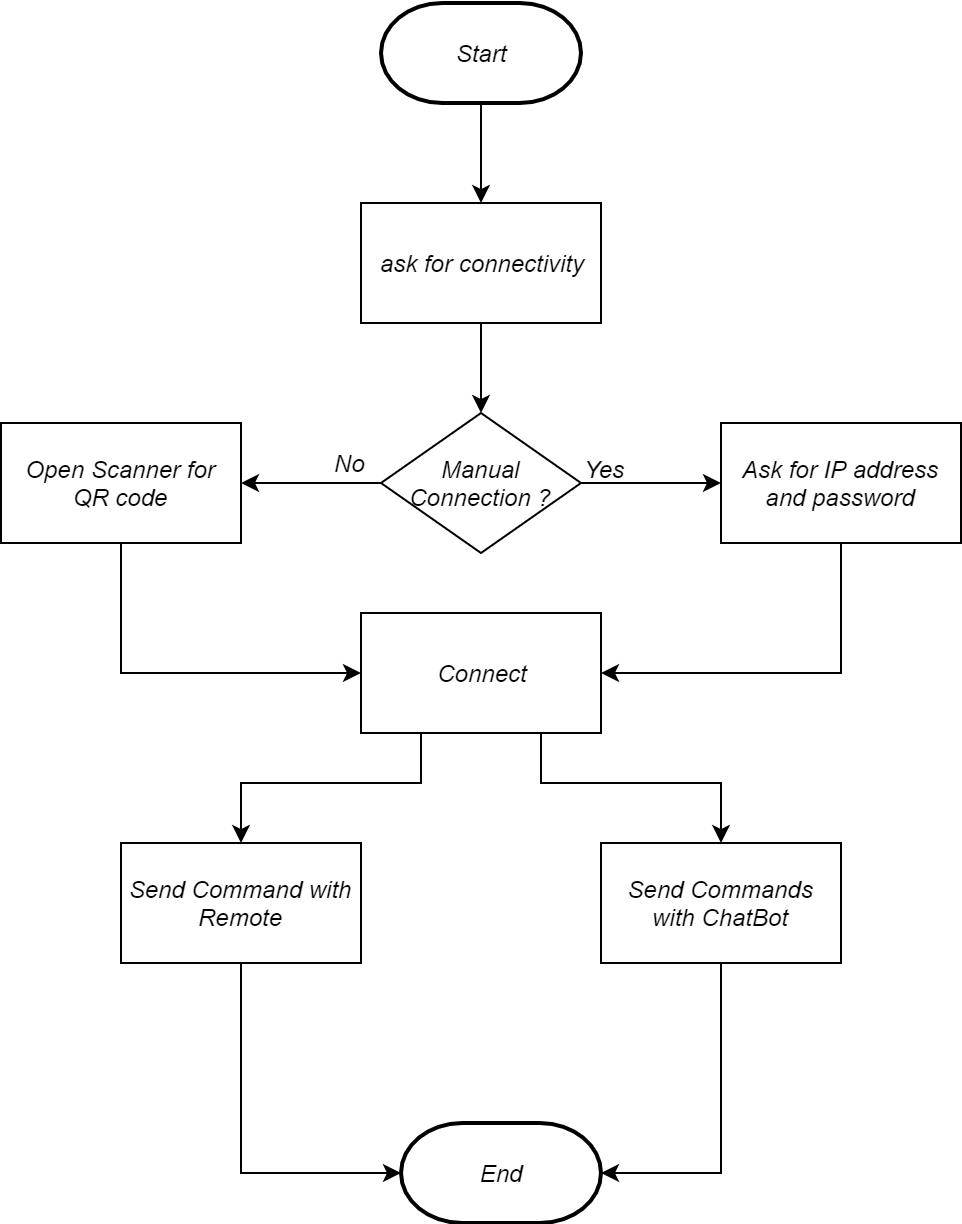


Figure 3:Control Flow Diagram of GhostAssist

**The Process Diagram of Software:**

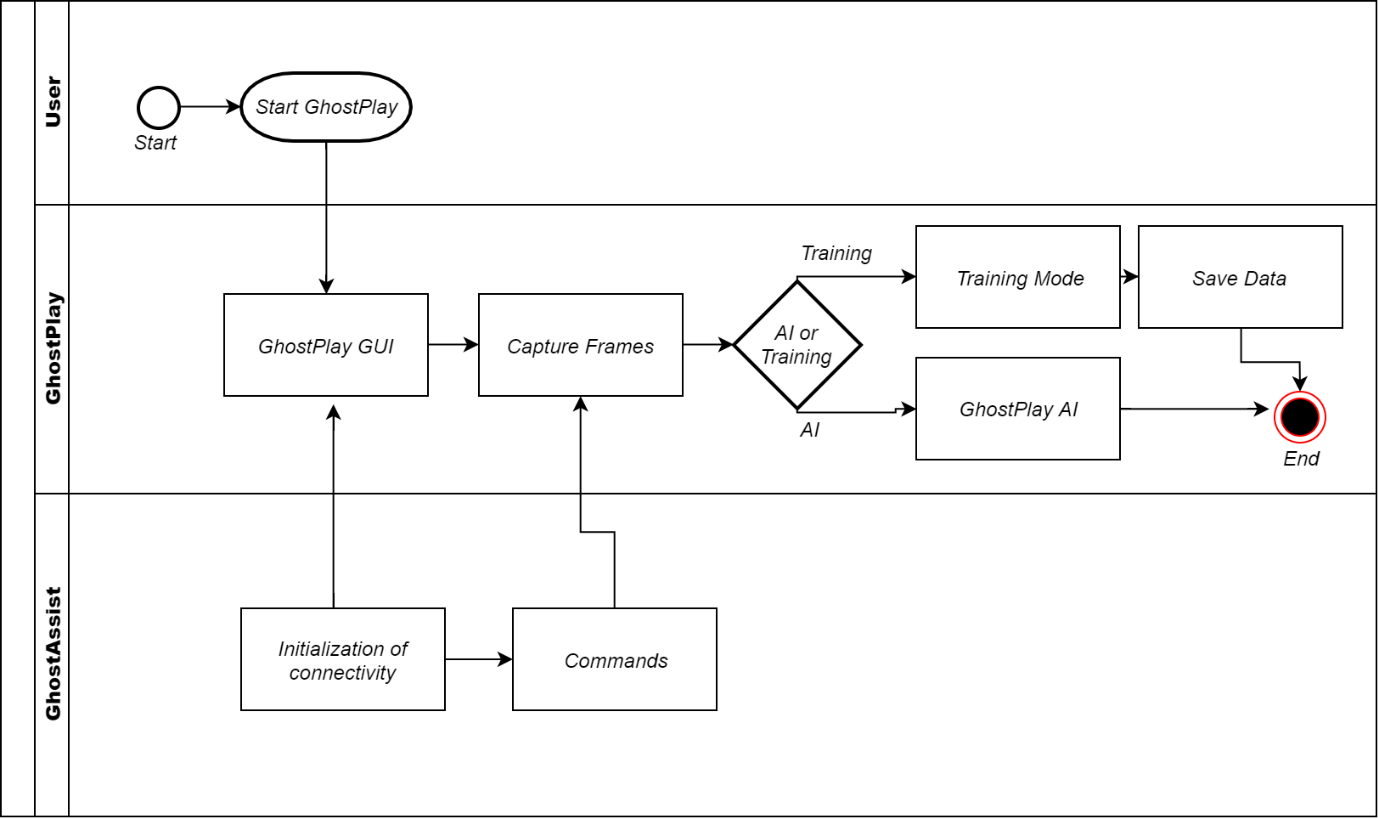


Figure 4:Process Diagram

**Algorithm Designs**:

In GhostPlay, the main Input data is the frames i.e. Sequence of images captured in real-time game environment. The frames are processed further to simplify to detect particular things of games. There are some other inputs taken from user to start and stop different modes of GhostPlay. Some other inputs are taken from GUI to modify the behavior of GhostPlay with Operating System.

The following used to initialize GhostPlay Software:

Step 1: Start GhostPlay GUI

Step 2: Minimize GUI

Step 3: Start Game

Step 4: Start AI using hotkeys

GhostPlay Uses further algorithm to find path to walk on the ground in the **game:**

Step 1: Capture Screen Frame

Step 2: Filter Captured Image

Step 3: Send it to identify path

Step 4: Check if path is available to walk, If yes then go forward by auto pressing key “W”

Step 5: If path is not available, check where to turn. Get current mouse position and move mouse according to the turn direction.

Step 6: If Ai Identifies particular number of frames are similar to each other then it detects that Player is stuck at some position. Then auto press ”S” for some milliseconds and turn accordingly.

Step 7: Release Frame

Step 8: Check if user pressed hotkey. If yes then end the screen capturing. If not then go to Step 1

**Training mode uses following algorithm:**

Step 1: Capture Screen Frame

Step 1: Walk for particular distance and capture frame

Step 2: Walk Again for particular distance and capture frame

Step 3: If all frames are different then walk back for same particular distance and capture particular part of screen.

Step 4: Check if similar captured part of screen is already present in data or not, if present then discard the captured part, or save in data

Step 5: If user pressed hotkey then stops capturing.

Step 6: Else Step 1 again until particular amount of data is collected

**Graphical User Interface:**

These are the screenshots of GhostPlay GUI:

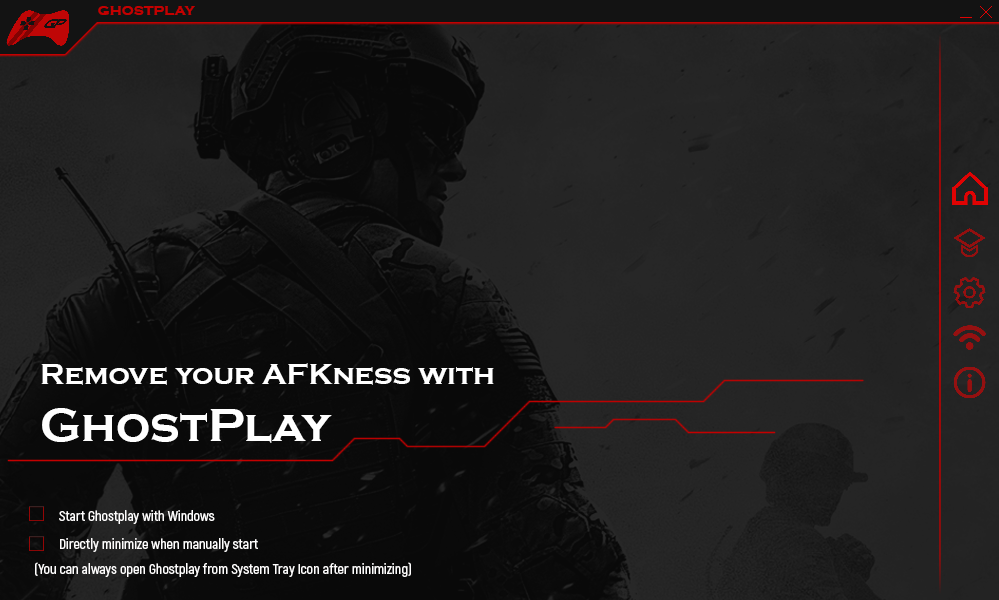


Figure 5: Home Widget

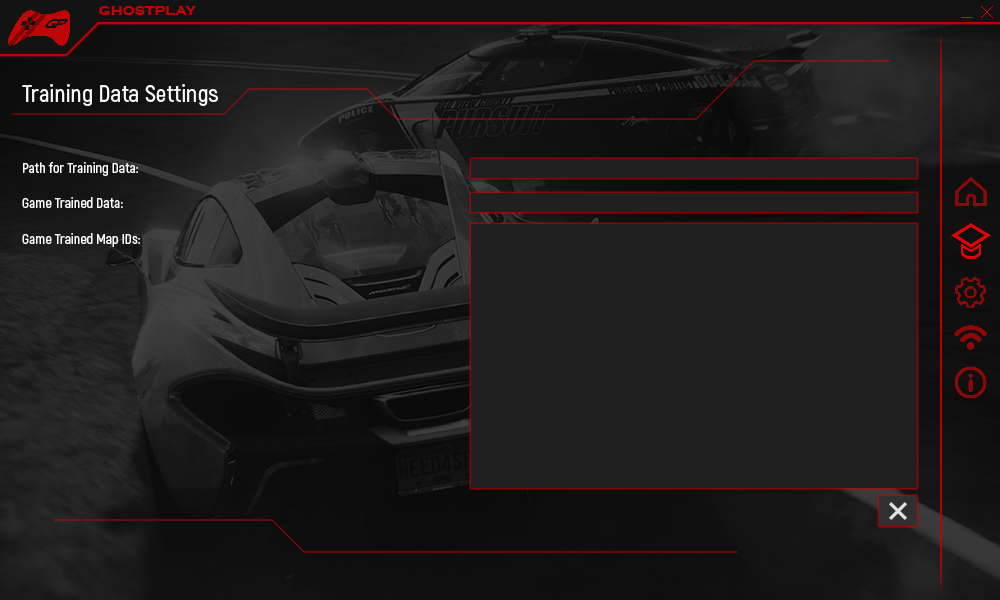


Figure 6: Training Widget

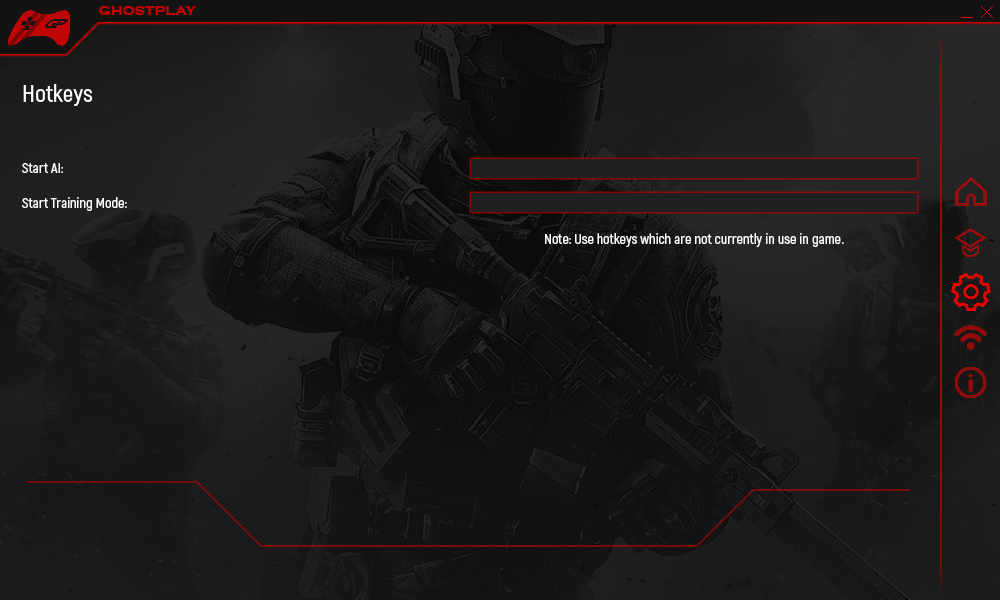
Figure 7: Settings Widget



Figure 8: Connectivity Widget

GUI is a small user interactive piece of software where user can tweak settings about GhostPlay.

The values of the GUI elements are fetched from SQLite database.

**Security Issues :**

The main problem with these kinds of software interaction in game is the Cheat Security provided by the developers of the games. When some kind of software tries to interfere with system files of software, it is identifying as the cheating in the game and player gets banned by the server.

There are several anti-cheat software’s used by game:

* PunkBuster
* BattleEye Anti-Cheat
* Denuvo Anti-Piracy
* Valve Anti-Cheat

To overcome this issue, GhostPlay don’t interfere with game’s system files. GhostPlay uses screen capturing to watch the environment of the game and interact in the game with keyboard and mouse simulation. By this way, game files don’t get modified and player stays safe from banning.