

# PYTHON PROJECT

## HANGMAN

**G. Madhavi(mentor)-16WH1A1225**

**D. N. V. Kiranmai-12WH1A1217**

**K. Sanjana-16WH1A1231**

**P. Swetha-16WH1A1252**

**INTRODUCTION:**

Hangman is a classic word game in which you must guess as many secret words as you can before chance runs out.

**PROJECT SCOPE:**

This project is useful in the gaming sector where people would love to play new , innovative and creative games in their daily life.

**OBJECTIVE:**

The objective of this game is to provide interaction between player and the computer in an efficient way. Any time, the player can play the game without any complex gaming issues of the user interface.

**INFRASTRUCTURE:****HARDWARE REQUIREMENTS:**

- ▶ Operating System: Windows 10
- ▶ Processor: intel i5 7<sup>th</sup> Gen
- ▶ Hard Disk: Minimum 32 GB
- ▶ RAM: Minimum 1 GB
- ▶ Sound Effects: Sound Card and Speakers

**SOFTWARE REQUIREMENTS:****1. Supported OS:**

- ▶ Windows
- ▶ Mac Os

**2. Software Tools Used:**

- ▶ Python -3.6 Version
- ▶ Pygame -1.9 Version

**SOFTWARE REQUIREMENT ANALYSIS:*****INDIVIDUAL PHASES OF THE PROJECT:***

There are 2 modules,

1. Player
2. Hangman(Computer Response)

**1. PLAYER:**

The Player tries to guess what Hangman thinks one letter at a time. If the word does not contain the suggested letter, the player's noose comes a step down. If the player guesses the correct word, the player beats Hangman and he wins the game one round at a time.

**2. HANGMAN:**

Hangman thinks of a word and draws a number of dashes equivalent to the number of letters in the word. If the player suggests a letter that occurs in the word, then the Hangman fills the blanks with that letter in the right places. If the word contains the suggested letter, the Hangman's noose comes a step down. If the player does not guess the correct word, then the Hangman beats the player and the Hangman wins the Game.

**FUNCTIONAL REQUIREMENTS:****1. Player Module:****▶ Choose to Play the Game:****Description:**

Player can play the game until he fails to guess the correct word.

**Dependencies:**

Only letters are allowed to be typed using keyboard.

**Ranking:**

Essential

**2. Hangman(Computer Response) Module:****▶ Check whether the letter is correct:****Description:**

Hangman checks letters entered by the player to be correct or not.

**Dependencies:**

Optional

**Ranking:**

Essential

**NON-FUNCTIONAL REQUIREMENTS:**

This section describes the non-functional requirements which includes the performance requirements of the project.

**PERFORMANCE REQUIREMENTS:**

The Performance Requirements refers to static numerical requirements placed on the interaction between the users and the software.

**1. Interface Loading Time:****Description:**

The loading time of each interface should be consistent. Each interface should load in less than 30 seconds.

**Constraints:**

The requirement can be dependent on the user's computer.

**Ranking:**

Desirable

**2. Maintainability:****Description:**

The game shall be designed with the view that bugs may need fixing, for future optimizations and for adding extra functionalities.

**Constraints:**

Inexperience of team members in design.

**Ranking:**

Essential

**3. Extendibility:****Description:**

The game shall be designed with the view that optional functional requirements will be implemented at a later date.

**Constraints:**

The current set of tasks are not independent of each other.

**Ranking:**

Desirable

**4. Portability:****Description:**

The game shall be designed in such a way to make it platform independent.

**Constraints:**

None

**Ranking:**

Essential

**5. Security:****Description:**

The game shall make sure that correct interfaces and functionalities are available according to the player that is using it.

**Constraints:**

The Player is recognized by the game.

**Ranking:**

Essential