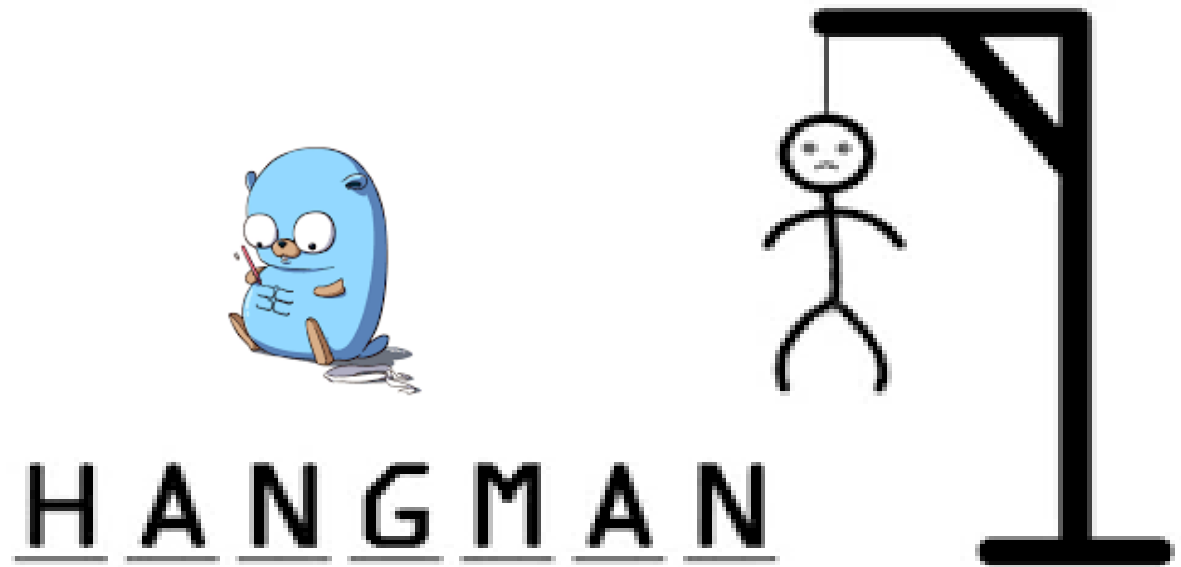


```
Command Prompt - java X + - □ X

=====
HangmanJ v1.0: Main Menu
=====

P - Play Game
M - Manage Words
E - Exit
=====

Choose:
```



# PRESENTATION DE NOTRE PROJET

Kwency  
Esteban  
Nicolas  
Amara

**- RAPPEL DES OBJECTIFS**

**- REPARTITION DES TACHES**

**-EXPLICATION DU CODE**

**-DIFFICULTES RENCONTRE**

**-VOIE D'AMELIORATION**



**RAPPEL  
DES  
OBJECTIFS**

**-CREE UN JEU DU PENDU  
SOUS GOLANG**

**-FINIR DANS LE TEMPS  
IMPARTI**

**-CREE UN REPOSITORY**

**ON AVANCE PLUS VITE TOUT SEUL  
MAIS  
NOUS ALLONS PLUS LOIN  
ENSEMBLE**



## EXPLICATION DU CODE

```
import (  
    "fmt"  
    "io/ioutil"  
    "log"  
    "math/rand"  
    "os"  
    "time"  
)
```

## EXPLICATION DU CODE

```
func GetSystemRandomInt(i int) int {  
    rand.Seed(time.Now().Unix())  
    return rand.Intn(i)  
    //return un nombre aleatoire  
}
```

## EXPLICATION DU CODE

```
func SplitWhiteSpaces(s string) []string {  
    var a []string  
    c := ""  
    v := 0  
    for i := range s {  
        v = i  
    }  
    for j, i := range s {  
        if string(i) == " " || string(i) == "\n" || string(i) == "    " || j == v {  
            if j == v {  
                c += string(s[len(s)-1])  
            }  
            if c != "" {  
                a = append(a, c)  
            }  
            c = ""  
        } else {  
            c = c + string(i)  
        }  
    }  
    return a  
    // append un mot a chaque saut de lignes (trier)  
}
```



## EXPLICATION DU CODE

```
func texte() string {  
    var tab []string  
    file, err := ioutil.ReadFile(os.Args[1])  
    if err != nil {  
        log.Fatal(err)  
    }  
    text := string(file)  
    tab = SplitWhiteSpaces(text)  
    mot := GetSystemRandomInt(len(tab))  
    return tab[mot]  
    //return un mot aleatoire de la liste  
}
```



## EXPLICATION DU CODE

```
func tirret(a string) []string {  
    var tab []string  
    src := len(a) - 1  
    rand.Seed(time.Now().Unix())  
    r := rand.Intn(src)  
    for j, i := range a[:len(a)-1] {  
        if j == r {  
            tab = append(tab, string(i))  
        } else {  
            tab = append(tab, "_")  
        }  
    }  
    return tab  
    //permet de transformer mon mot en _ sauf un seul  
}
```

## EXPLICATION DU CODE

```
func fonc_list(char []string, a string, b string) []string {  
    for n, i := range b {  
        if string(i) == a {  
            char[n] = a  
        }  
    }  
    return char  
    // a chaque bonne lettre de l'utilisateur le programme modifiera la liste  
}
```

## EXPLICATION DU CODE

```
func main() {
    Word := texte()    // texte prend un mot aleatoire et l'attribut a word
    b := (tirret(Word)) // word mot attribuer
    Find := ""
    erreur := 0 // initialisé a 0
    cmp := false // quand l'utilisateur a bon return true quand il a faut return false
    compare := "" //prendre les ellements de la liste et va les comparer a word
    fmt.Print("\n")
    for true { // pour que le programme s'execute jusqu'a l'infini
        for _, i := range b {
            fmt.Print(i)
            fmt.Print(" ") // espaces
        }
        fmt.Print("\n")
        fmt.Print("Taper un mot\n")
        fmt.Scanln(&Find) // ce que l'utilisateur tape sur le clavier

        if Find == Word[:len(Word)-1] {
            fmt.Println("<<<---Bravo tu es le boss métallique--->>>")
            return
        }
    }
}
```



## EXPLICATION DU CODE

```
for _, i := range Word {  
    if string(i) == Find {  
        cmp = true  
    }  
}
```

## EXPLICATION DU CODE

```
if cmp == false {  
    erreur++  
    pendu(erreur)  
    if erreur == 10 {  
        fmt.Print("<<<-----LOOOOOOOOOOSERRRR----->>>")  
        return  
    }  
} else if cmp == true {  
    b = fonc_list(b, Find, Word) // modifie la liste de b à chaque lettre trouvé  
}
```

## EXPLICATION DU CODE

```
for _, i := range b { // transforme le tableau en string pour le comparer a word  
    compare = compare + string(i)  
}
```

```
    if compare == Word[:len(Word)-1] {  
        fmt.Print("Bravo tu as réussi")  
        return  
    }  
    compare = ""  
    cmp = false  
}  
}
```



## EXPLICATION DU CODE

```
func pendu(nbr int) {  
    if nbr == 1 {  
        fmt.Println("    ")  
        fmt.Println("    ")  
        fmt.Println("    ")  
        fmt.Println("    ")  
        fmt.Println("    ")  
        fmt.Println("    ")  
        fmt.Println("=====")  
    } else if nbr == 2 {  
        fmt.Println("    ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")  
    } else if nbr == 3 {  
        fmt.Println(" +---+")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")
```

```
    } else if nbr == 4 {  
        fmt.Println(" +---+")  
        fmt.Println(" | | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")  
    } else if nbr == 5 {  
        fmt.Println(" +---+")  
        fmt.Println(" | | ")  
        fmt.Println(" 0 | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")  
    } else if nbr == 6 {  
        fmt.Println(" +---+")  
        fmt.Println(" | | ")  
        fmt.Println(" 0 | ")  
        fmt.Println(" | | ")  
        fmt.Println(" | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")
```

```
    } else if nbr == 7 {  
        fmt.Println(" +---+")  
        fmt.Println(" | | ")  
        fmt.Println(" 0 | ")  
        fmt.Println(" /| | ")  
        fmt.Println(" | | ")  
        fmt.Println("=====")  
    } else if nbr == 8 {  
        fmt.Println(" +---+")  
        fmt.Println(" | | ")  
        fmt.Println(" 0 | ")  
        fmt.Println(" /|\ | ")  
        fmt.Println(" | | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")  
    } else if nbr == 9 {  
        fmt.Println(" +---+")  
        fmt.Println(" | | ")  
        fmt.Println(" 0 | ")  
        fmt.Println(" /|\ | ")  
        fmt.Println(" / \ | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")  
    } else if nbr == 10 {  
        fmt.Println(" +---+")  
        fmt.Println(" | | ")  
        fmt.Println(" 0 | ")  
        fmt.Println(" /|\ | ")  
        fmt.Println(" / \ | ")  
        fmt.Println(" | ")  
        fmt.Println("=====")  
        fmt.Println("<<<<<<---ahaha t'as perdu tocard--->>>>>>")  
    }  
}
```

**DIFFICULTES  
RENCONTRES**

**-DIFFICULTES DE  
COMPREHENSION  
DES CONSIGNES**

**-REALISATION DES  
TACHES**

**-DIFFICULTE DU  
PROGRAMME**



## **VOIE D'AMELIORATION**

**-POUVOIR AVOIR ACCES A  
NOTRE JEU DEPUIS LE WEB**

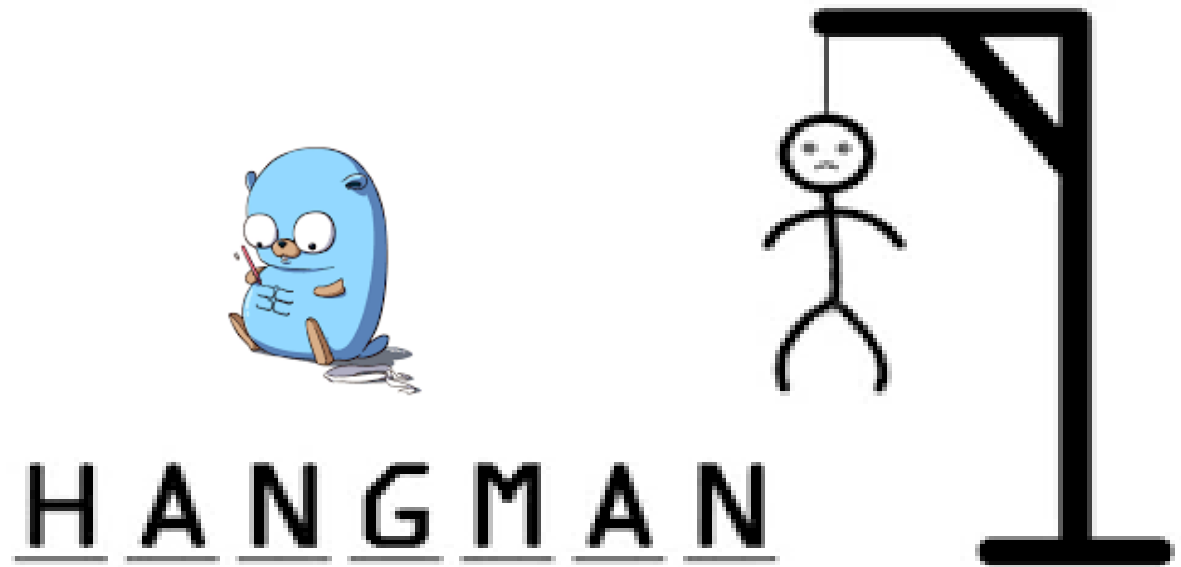


```
Command Prompt - java X + - □ X

=====
HangmanJ v1.0: Main Menu
=====

P - Play Game
M - Manage Words
E - Exit
=====

Choose:
```



# CONCLUSION

Kwency  
Esteban  
Nicolas  
Amara