

LE 11 JANVIER 2022

Hangman Web

Projet Hangman version
WEB Ynov 2022



ALEXANDRE



MELVIN

Backend



Hangman WEB

Qu'est-ce que c'est ?

Hangman

On nous a demandé de refaire le jeu du hangman (le jeu du pendu en français) en langage GO et de pouvoir en faire un mini jeu via notre terminal.

Hangman WEB

c'est le même mini jeu sauf qu'ici nous avons utilisé le langage HTML et CSS pour l'utiliser sur des pages web et le rendre plus attrayant.

HTML

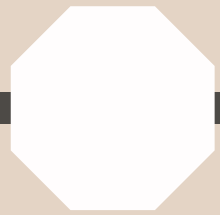


CSS

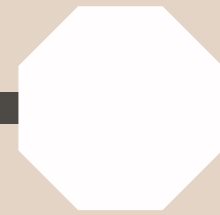


Frontend

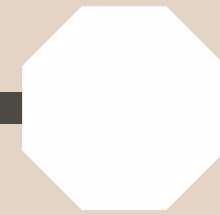
PLAN



Les objectifs



le programme et
c'est fonctions



Bilan

NOS OBJECTIFS

1

Adaptez le code au WEB

2

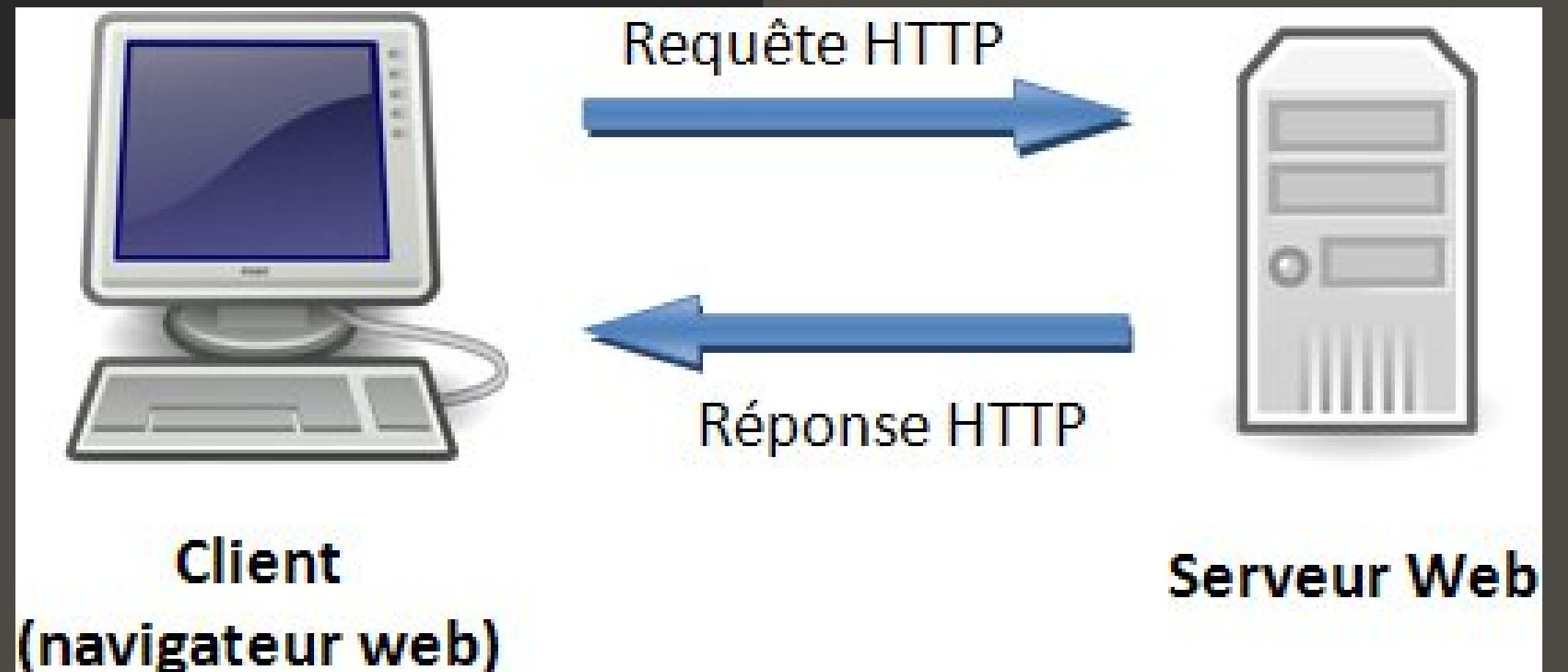
Rendre l'interface esthétique

3

corriger les bugs et erreurs

SCREENSHOT

```
func main() {  
  
    // Add the static path where our server search assets like css/fonts/js/img  
    fs := http.FileServer(http.Dir("assets"))  
    http.Handle("/assets/", http.StripPrefix("/assets/", fs))  
    // name understand by the server      name in my computer  
  
    // Apply a function in this page (don't worry i display every time a html template ^^)  
    http.HandleFunc("/", HomeHandler)  
    http.HandleFunc("/hangman", HangmanHandler)  
  
    // Open the server (let's go)  
    fmt.Println("Open server at http://localhost:8080")  
    log.Fatal(http.ListenAndServe(":8080", nil))  
}
```



SCREENSHOT

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="/assets/css/teststyle.css">
  <title>Document</title>
</head>
<body>
  <center>{{.LifePercent}}</center>
  {{.Hangman}}
  {{.Img}}
  <p>{{.Word}}</p>
  {{.EntryPart}}
</body>
</html>
```

Client

```
func OpenTemplate(fileName string) *template.Template {
    tmpl, err := template.ParseFiles(fmt.Sprintf("./assets/templates/%s.html", fileName))
    if err != nil {
        fmt.Println(err.Error())
    }
    return tmpl
}
```

SCREENSHOT

```
func HangmanHandler(w http.ResponseWriter, r *http.Request) {  
    // Error 404 detection  
    if r.URL.Path != "/hangman" {  
        ErrorHandler(w, r, http.StatusNotFound)  
        return  
    }  
  
    // Get back the letter enter and test if she's valid  
    letterGuessed := ""  
    if r.FormValue("test") != "" {  
        letterGuessed = r.FormValue("test")  
        if !hangman.IsValidEntry(letterGuessed) {  
            hangman.GuessingLetter(letterGuessed)  
            data.Word = hangman.HideWord(hangman.WordChosen, hangman.LettersAlreadyAppeared)  
            if data.Word == oldWord {  
                data.Attempts--  
                data.Hangman = fmt.Sprintf("<img class=\"hangman\" src=\"/assets/img/hangman%d.png\"></img>", 10-data.Attempts)  
                data.LifePercent = fmt.Sprintf("<div class=\"bar\"><div class=\"percentage has-tip\" style=\"width: %d%%\"></div></div>", dat  
            }  
        }  
    }  
}
```

Serveur

PROBLÈME RENCONTRER

1

l'adaptation vers

WEB

une légère difficulté au départ du projet
pour le porter vers une page web

2

Gestion des erreurs

crée les pages correspondant au
erreur diverse du WEB pour
l'utilisateur

3

gestion des

chemins du projet

gérer les différent fichier et repertoire
du projet pour sont bon
fonctionnement

ENRICHISSEMENT PERSONNEL

- (Melvin) apprendre avec alexandre sa vision de la programmation
- (Alexandre) decouverte de la programmation back-end dans le web
- gérer les fichier entre back-end et front-end pour les faire fonctionner en harmonie

**MERCI POUR
VOTRE ECOUTE**



ALEXANDRE



MELVIN