

DOCUMENTATION

01 - Décomposition du projet et phases clé	p1
02 - Répartition des tâches entre les membres de l'équipe	p1
03 - Stratégie d'organisation du travail	p2
04 - Gestion du temps et des priorités	p2
05 - Stratégie adoptée pour se documenter	p2
06 - Installation et lancement du projet	p4
07 - Différentes routes utilisées dans l'application	p4

INTRODUCTION

Le projet Hangman Web vise à transformer un jeu de pendu CLI en une application web en Go, en utilisant uniquement des bibliothèques standard. Nous avons décomposé le projet en phases clés : analyse, conception, développement, persistance des données, et tests. Les tâches ont été réparties en fonction des compétences de chaque membre, avec une gestion du temps priorisant les fonctionnalités essentielles.

01 -Décomposition du projet et phases clés

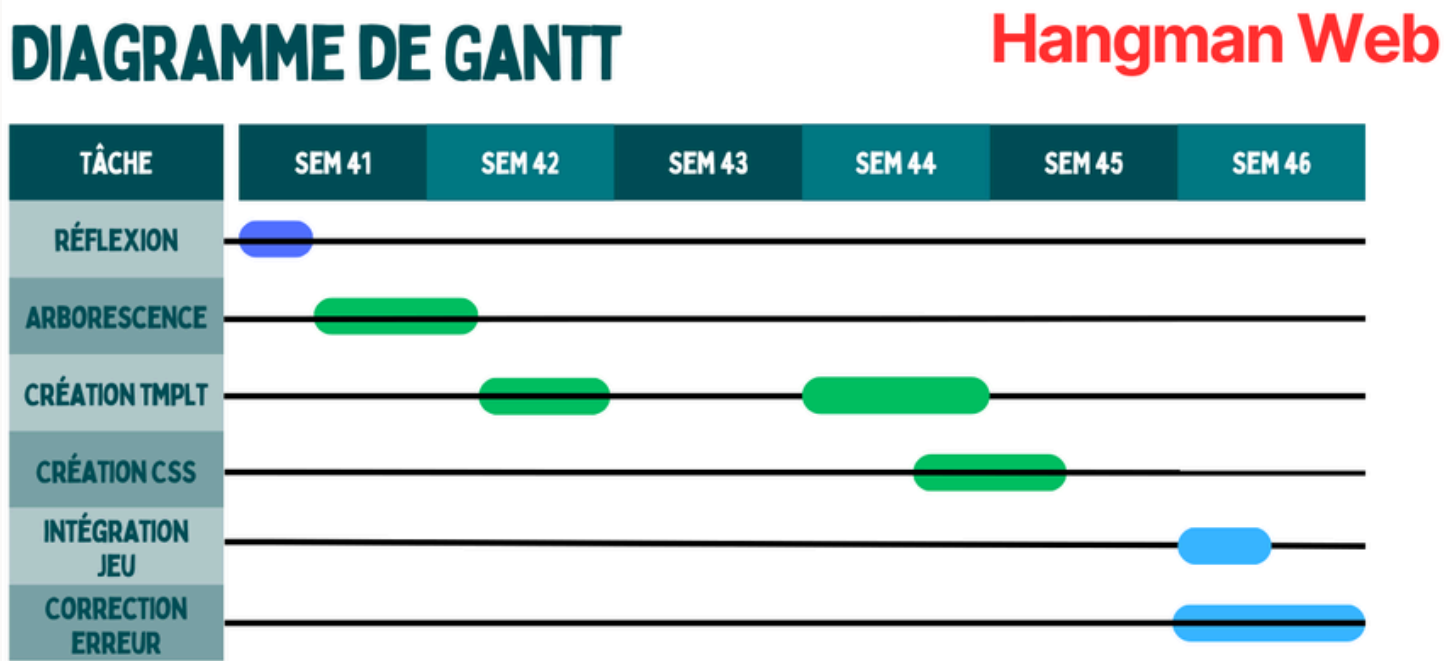
Le projet Hangman Web a été divisé en plusieurs phases. D'abord, une analyse des exigences et la conception de l'architecture du projet. Ensuite, nous avons développé les fonctionnalités principales, telles que le traitement des soumissions de lettres et l'affichage des pages de jeu. Nous avons intégré la persistance des scores et, enfin, réalisé des tests pour vérifier le bon fonctionnement du jeu et corriger les bugs. Cette décomposition a permis une gestion efficace du projet.

02 - Répartition des tâches entre les membres de l'équipe

Pour assurer une répartition efficace des tâches, nous avons structuré le travail en fonction des compétences spécifiques de chaque membre, adoptant une approche agile pour faciliter l'adaptation et la collaboration. Chaque membre a pris en charge un domaine du projet permettant ainsi un développement en parallèle et une avancée plus rapide. Des réunions courtes et régulières nous ont permis de faire le point sur les progrès, d'ajuster les priorités, et de résoudre les éventuels problèmes. Grâce à cette organisation, chaque membre a pu se concentrer sur son domaine de spécialité, tout en maintenant une coordination et une cohésion globale dans le projet.

03 - Stratégie d'organisation du travail

Afin de pouvoir bien s'organiser pour le projet Romain a fait un diagramme de Gantt



04 - Gestion du temps et des priorités ?

Pour gérer notre temps et nos priorités, nous avons défini les tâches essentielles à accomplir en priorité, comme la logique du jeu et la création des routes.

Nous avons utilisé des sprints courts, avec des réunions régulières pour suivre l'avancement et ajuster les priorités si nécessaire.

Les tâches secondaires, telles que la persistance des scores et l'amélioration de l'interface, ont été réalisées ensuite.

À la fin, nous avons consacré du temps aux tests et corrections pour assurer la stabilité du projet.

Cette organisation nous a permis de respecter les délais et de livrer un projet fonctionnel.

05 - Stratégie adoptée pour se documenter

Pour nous documenter, nous avons consulté des sources en ligne telles que des articles techniques et des vidéos YouTube.

Nous avons également utilisé ChatGPT pour résoudre des problématiques spécifiques et approfondir notre compréhension des outils.

Cela nous a permis d'enrichir nos connaissances et de progresser efficacement dans le projet.

06 - Installation et lancement du projet

Pour installer et lancer le projet Hangman Web, commencez par cloner le dépôt Git à l'aide de la commande suivante :

1: `git clone github.com/RomainD13122/hangman-web-projet`

2: `go run main.go`

3: `http://localhost:8080`

Aucune dépendance externe n'est requise, ce qui permet de démarrer le projet directement avec Go.

07 - Différentes routes utilisées dans l'application

L'application Hangman Web utilise plusieurs routes pour gérer les interactions avec l'utilisateur et afficher les différentes vues :

- 1./ : Route d'accueil, affichant la page d'introduction avec un bouton pour démarrer une nouvelle partie.
- 2./game : Route principale du jeu, où l'utilisateur peut voir l'état du jeu (mots à deviner, lettres déjà proposées, etc.) et soumettre ses lettres.
- 3./game/submit : Route pour traiter la soumission d'une lettre par l'utilisateur et mettre à jour l'état du jeu (nombre de tentatives restantes, mot partiellement découvert, etc.).
- 4./end : Route affichant la page de fin de partie, avec le résultat du jeu (gagné ou perdu) et l'option de recommencer une partie.
- 5./scoreboard : Route pour afficher le tableau des scores, avec la liste des meilleurs scores enregistrés.

Ces routes permettent de distribuer des vues HTML pour l'interface utilisateur et de traiter les données nécessaires à l'avancement du jeu.

08-Conclusion

En conclusion, le projet Hangman Web a été une réussite grâce à une planification soignée, une répartition claire des tâches et une gestion efficace du temps.

Nous avons réussi à transformer un jeu de pendu en ligne de commande en une application web fonctionnelle, respectant les contraintes techniques imposées.

Le processus de développement nous a permis d'apprendre et d'améliorer nos compétences en Go et en développement web.

Les tests et l'optimisation ont garanti une expérience utilisateur fluide.

Ce projet a renforcé notre capacité à travailler en équipe tout en respectant les délais.