

Document complémentaire au diaporama

Diapo 1 : Bonjour, dire le sujet, présentation de l'équipe, la durée du projet

Diapo 2 : présentation du sommaire

Diapo 3 : Lire la définition + l'expliquer avec nos propres mots "permutation de sons dans une phrase, donnant une nouvelle phrase. Cette nouvelle phrase est souvent vulgaire et est cachée par la phrase de départ" -> donner exemple

Diapo 4 : dire les objectifs et les détailler : dire que c'est à but pédagogique, que le site doit être ergonomique et simple d'utilisation.

Diapo 5 : dire les langages + expliquer pourquoi : objectifs de résultats rapides en python qu'on connaît plutôt que d'apprendre le JS et attendre pour des résultats. Dire les langages pour le site qui sont logiques

Diapo 6 : re-citer les langages en disant pourquoi nous les avons utilisés

Diapo 7 : Introduction de la méthode de travail : dire que nous avons réfléchi à une stratégie pour résoudre les problèmes posés. Dire que ces 3 points ont été essentiels pour le travail.

Diapo 8 : important de bien définir les objectifs avec un délai de résolution. 3 objectifs seulement car une fois ces objectifs faits, le reste serait du recopiage. 1er objectif sous 1 mois car ça va être la base des prochains algorithmes mais assez de temps pour chercher des informations sur les contrepèteries et des dictionnaires. Réalisé une permutation de sons -> on approche du but en Python donc le réalisé à mi-projet est un bon timing. Enfin un site vitrine afin de pouvoir présenter quelque chose de concret avec comme délai la 1ere soutenance également. Ces délais ont été fixé avec l'idée de "ça doit être fait pour X". Le fait de diviser le projet en sous-objectifs comme ici nous a permis de voir de vraies avancées lorsque nous finissons un sous-objectif

Diapo 9 : le projet étant en équipe, plutôt nombreuse, créer des sous équipes nous a semblé indispensable afin d'avancer plusieurs objectifs en même temps mais surtout pour structurer l'équipe afin que tout le monde trouve sa place et qu'il n'y ait pas d'embrouille dans la résolution du projet. Ainsi 3 équipes ont été créées, 2 pour la "période 1" du projet et 2 pour la "période 2". Une équipe égal un langage -> dire les équipes. Ces équipes ont été faites en fonction du niveau et des envies de chacun afin de garder la motivation sur le projet.

Diapo 10 : présenter le Gantt, il a été réalisé en fin de projet car nous avons utilisé la méthode AGILE, c'est-à-dire que nous n'avons pas de planification sur la totalité du projet mais nous avançons semaines par semaines avec des échanges avec le client s(notre tuteur). Chaque semaine, nous cherchons des améliorations sur ce qui avait été fait la semaine précédente. Et chaque semaine, nous planifions le travail à faire lors de la semaine

suivante, tout en gardant en tête les délais que nous étions fixés en début de projet afin de contrôler l'avancé du projet. Description du Gantt

Diapo 11 : partie python : expliquer les objectifs en python

Diapo 12 : multiplication du problème pour mieux le résoudre : nous ne nous sommes pas lancés dans la résolution de la version finale dès la première semaine mais nous avons séparé cette version en sous-version plus simple à résoudre. Pour cela création de type de contrepèterie, et chaque type aura un algorithme associé qui va trouver les contrepèteries

Diapo 13-16 : type : lire et expliquer + exemples. Dire en quoi la solution change

Diapo 17 : faire correspondre les algos avec les types

Diapo 18 : présentation du 1er algo, reprise du type et de l'objectif de celui-ci. Explication de l'algo en français. Dire le dictionnaire utilisé

Diapo 19 : Présenter résultats

Diapo 20 : Test de performance, dire que c'est important pour nous. Nous avons cherché à optimiser notre algo -> 2 solutions : le code ou la structure. On a essayé le code mais rien de concluant pour cette version et pire à l'avenir. Donc sur la structure

Diapo 21 : Lire + expliquer pourquoi ça peut nous faire gagner du temps. Expliquer l'algo

Diapo 22 : Commenter résultats et conclure sur la structure : utilisation des SET

Diapo 23 : Reprise algo : explication du type et objectif

Diapo 24 : Expliquer résultats

Diapo 25 : Explication de l'utilisation des dictionnaires, + dire lequel on utilisait jusque-là. Poser problème de sons. Expliquer qu'un son n'est pas égal à une lettre et qu'il y a plusieurs combinaisons. Exemple : c ch cu ce.

Diapo 26 : Nous avons fait face aux problèmes de la langue française donc utilisation provisoire d'un dictionnaire de sons anglais.

Diapo 27 : volonté d'avoir un dictionnaire en français, solution scraper : (récupérer le contenu d'un site), en l'occurrence wiktionary afin de récupérer le son. Nous avons donc récupéré le contenu de plus de 300000 mots avec leurs sons que nous avons insérés dans un fichier CSV (Expliquer le fonctionnement plus en détail)

Diapo 28 : explications algo + type + dictionnaires. La phrase n'est plus vue comme une suite de mots mais comme une suite de sons.

Diapo 29 : Présenter le résultat, dire que c'est un résultat d'une version améliorée de cet algo, avec l'utilisation de voisin, permettant de trouver des contrepèteries plus complexes. Montrer la ligne avec les sons

Diapo 30 : Avis critique vis-à-vis des résultats. Présentation du scrapping de types. Expliquer ensuite la complexité pour appliquer des règles telles que celles de Noam Chomsky. Mais dire la règle simple que nous avons utilisée.

Diapo 31 :

Diapo 32 : Afin de réaliser un site WEB et pour que ce dernier soit utilisable côté client, nous avons dû transformer le code que nous avions en Python en Javascript. Pour cela nous avons ensuite réécrit la logique de ces algorithmes en JavaScript, pour son efficacité côté client.

Diapo 33 : Nous avons rapidement créé un site vitrine, sans interaction avec l'utilisateur, afin d'avoir un visuel sur la structure de notre site et qu'il soit validé rapidement car le projet ayant un but pédagogique, le site doit être ergonomique et facilement lisible.

Diapo 34 : Sur cette diapo, on retrouve une capture d'écran de la première page du site. On a donc en haut de page Qu'est ce qu'une contrepèterie. Et ensuite dans la page, on retrouve des définitions et des exemples tels que : Je vous laisse le choix dans la date. Dont je vous laisse chercher la réponse.

Diapo 35 : Ici, on peut voir les onglets de la page, dans l'onglet classification on trouve quatre pages où l'on a rangé les contrepèteries par type. La contrepèterie classique est une permutation de consonnes. La seconde contrepèterie porte sur la permutation de voyelles ou encore de syllabes. Dans la contrepèterie irrégulière, on range dans cette dernière catégorie, toutes les contrepèteries qui n'appartiennent pas aux deux précédentes. C'est à dire déplacement d'une ou plusieurs lettres, ou d'une permutation de lettres à l'intérieur d'un mot.

Diapo 36: Sur cette diapo, on voit une capture d'écran de la contrepèterie classique, on a la définition et ensuite quelques exemples. Le projet ayant un but pédagogique, il nous semblait important de définir les différents types qui existent

Diapo 37 : l'aide à la contrepèterie est un objectif ajouté en cours de projet. Cette aide consiste à donner une liste de mots pouvant être complémentaire au entré

Diapo 38 : On peut voir une capture de la page d'aide à la contrepèterie avec un bouton pour charger le dictionnaire et une zone de texte où l'on rentre un mot. Quand on appuie sur le bouton "lancer la recherche" une liste de mots apparaît.

Diapo 39 : Ici, on fait le test avec un mot qui est sachons. Le site affiche donc tous les mots par qui peuvent être trouvé en changeant un son.

Diapo 40 : La génération de contrepèteries suit le même principe que la suggestion de mots mais cette fois pour toute une phrase. On commence par charger le dictionnaire et ensuite quand l'utilisateur rentre un mot l'algorithme renvoie les contrepèteries possibles s'il y en a.

Diapo 41 : On peut voir sur cette diapo une capture de la page de génération de contrepèteries. Comme la page d'aide à la contrepèterie il y a un bouton pour charger le dictionnaire et une zone de texte pour entrer la phrase.

Diapo 42 : Dans les deux pages que nous avons présentés juste avant, nous avons besoin de charger un dictionnaire pour que les algorithmes fonctionnent. Pour cela nous avons utilisé PapaParse. Pour nous il lis un fichier csv, qui est le dictionnaire, ligne par ligne pour le charger dans la page.

Diapo 43 : Nous allons maintenant passer au jeu. Le but est simple, on affiche une phrase à l'écran et le but est de trouver les deux lettres ou sons à échanger pour trouver la contrepèterie. On commence avec une minute et à chaque contrepèterie trouvée on ajoute 10 secondes au chronomètre. Le jeu se termine quand le chronomètre arrive à 0 et on affiche le score du joueur.

Diapo 44 : Il s'agit de la page Jeu où se trouve le jeu de contrepèterie. Il suffit de cliquer sur commencer la partie pour commencer à jouer.

Diapo 45 : Voici un exemple de contrepèterie présente dans le jeu. Ainsi en ayant cliqué sur la bonne paire de bouts de mots ou lettres, qui sont en bleus, la réponse s'affiche en vert et on peut cliquer sur contrepèterie suivante pour continuer le jeu et chercher la réponse de la prochaine contrepèterie.

Diapo 46 : Sur cette diapo, on constate que lorsque le timer est écoulé, le score du jeu apparaît indiquant le nombre de contrepèteries trouvées avec le nombre total de contrepèteries disponibles.

Diapo 47 : On peut changer la langue du site entre anglais et français.

Diapo 48 : Il s'agit de la page principale du site en anglais à laquelle on peut accéder en appuyant sur l'icône en anglais en haut à droite.

Diapo 49 : En conclusion, nous sommes satisfaits de notre, nous avons obtenu des résultats en Python et le site est toujours en finition. L'objectif de génération et vérification est globalement réussi, nous pouvons retrouver n'importe quelle contrepèterie, cependant nous ne gérons pas (encore) la grammaire ce qui nous donne beaucoup de faux résultats, c'est-à-dire une phrase sans aucun sens. Concernant le site WEB, nous sommes satisfait de son design, nous pensons qu'il est ergonomique et simple à utiliser. Nous y donnons les informations que nous avons jugées essentielles. Le système d'aide à la contrepèterie fonctionne. Notre jeu fonctionne avec une réserve de contrepèteries suffisantes.

Plus globalement, ce projet nous a permis de travailler dans une équipe nombreuse et d'apprendre à gérer un projet du début à la fin. Nous allons maintenant tous faire plus attention aux phrases utilisées et en utiliser certaines contrepèteries.

Enfin, nous pensons que le projet est une réussite mais pas une finalité. En effet, plusieurs améliorations sont possibles : gérer la grammaire dans le vérificateur et le générateur de contrepèteries. Plusieurs modes de jeu sont ajoutables pour permettre à chaque personne de trouver celui qui lui convient.