

RAPPORT DE PROJET

Proposé à :

Sonntag Benoit

Par :

Ashkan MOTAMEDIFAR

Mehdi JAFARI ZADEH

Hiver 2023

TABLE DES MATIERES

Introduction	4
Architecture et Modélisation	4
Algorithmes	4
Génération de Grille (Generation)	4
Affichage de la Grille (GrilleGUI) :	5
Vérification de la Grille.....	5
Choix de Modélisation et d'Implémentation.....	5
Utilisation du programme.....	5
Sur le menu principal	5
Nouvelle Partie	5
Charger une grille depuis un fichier	5
Charger une partie sauvegardée	6
Tableau des scores.....	6
Quitter.....	6
En jeu.....	6
Barre de navigation.....	6
Temps écoulé.....	6
Sauvegarder :	6
Vérifier.....	6
Quitter la partie	6
Conclusion.....	6

Introduction

Ce rapport analyse le projet Java de création d'une application pour charger, afficher, et générer des grilles de mots croisés. Le projet est structuré en plusieurs packages qui séparent clairement les responsabilités entre l'interface utilisateur (GUI), la logique du jeu (Game), et les utilitaires (Utilitaire). Chaque classe joue un rôle spécifique dans la réalisation des objectifs du projet.

Architecture et Modélisation

- Package GUI : Contient les classes pour l'interface graphique. Menu sert comme point d'entrée pour l'utilisateur, GrilleGUI affiche la grille, et CasePanel représente chaque case dans la grille. Cette séparation permet une gestion claire de l'interface utilisateur.
- Package GrilleMots : Comprend les classes Grille, Case, et Mot. Grille représente la grille de mots croisés, Case une case individuelle, et Mot un mot dans la grille. Cette conception reflète la structure réelle d'une grille de mots croisés.
- Package Game : Inclut Generation pour la génération de la grille, Score pour la gestion des scores, et Temps pour le suivi du temps. Generation est particulièrement important car il implémente l'algorithme de génération de grille.
- Package Utilitaire : Fournit des fonctions utilitaires comme la sélection de fichiers et la limitation de la taille des entrées.

Algorithmes

Génération de Grille (Generation)

- Extraction des mots : Lit un fichier de mots et définitions, créant une liste de Mot.
- Placement des mots : Trie les mots par longueur et les place dans la grille. Utilise un algorithme qui tente de placer chaque mot de manière aléatoire, tout en vérifiant les contraintes de taille et les intersections possibles avec des mots déjà placés.

Affichage de la Grille (GrilleGUI) :

Génère un affichage graphique de la grille. Chaque Case est représentée par un CasePanel, permettant des interactions utilisateur comme la saisie de lettres.

Vérification de la Grille

Après que l'utilisateur a rempli la grille, la vérification compare la grille remplie avec la grille de référence pour déterminer si la grille est correctement remplie.

Choix de Modélisation et d'Implémentation

- Modularité : La séparation en package distincts améliore la maintenabilité et la compréhension du code. Chaque classe a une responsabilité unique, ce qui suit le principe de responsabilité unique (SRP).
- Orienté Objet : Le projet utilise une approche orientée objet, ce qui facilite la représentation des concepts du monde réel (comme les grilles, les cases, et les mots) et favorise la réutilisation du code.
- Flexibilité : La capacité de charger des grilles à partir de fichiers et de sauvegarder des états de jeu offre une grande flexibilité à l'utilisateur.

Utilisation du programme

Sur le menu principal

Nouvelle Partie

Permet de sélectionner un fichier. gc2wds qui permet de générer une grille de mots croisés. Sélectionne le nombre de lignes et le nombre de colonnes.

Charger une grille depuis un fichier

Sélectionne un fichier. gcwds et affiche la grille.

Charger une partie sauvegardée

Sélectionne un fichier.cwds qui est l'état d'une partie antérieure et charge cette partie.

Tableau des scores

Affiche le tableau des scores organisé par type de grille.

Quitter

Quitter le jeu.

En jeu

Barre de navigation

Temps écoulé

Indique le temps de jeu.

Sauvegarder :

Permet de sauvegarder l'état courant du jeu dans un fichier .cwds.

Vérifier

Permet de vérifier l'état du jeu. Cadre rouge si une lettre est au mauvais endroit et cadre vert si une lettre est au bon endroit. Si toutes les cases sont correctement remplies, la partie est considérée comme terminée. Si on a un nouveau meilleur temps, le pseudo du joueur est demandé, sinon, il est enregistré au nom "Anonymous". On affiche le tableau des scores.

Quitter la partie

Revient au menu principal.

Conclusion

Le projet de générateur et afficheur de grilles de mots croisés est un exemple bien structuré d'une application Java utilisant les principes de la programmation orientée objet. L'accent mis sur la modularité et la clarté dans la conception permet une maintenance et une évolution aisées du projet. Les algorithmes

de génération et de vérification de grille sont des points clés de l'application, offrant une expérience utilisateur interactive et engageante.