

# Conception Orientée Objet

---

## Guess The Word - Compte Rendu TP1

## 1 Ce qui a été réalisé

### 1.1 Création du jeu guess the word

L'utilisateur cherche à deviner un mot en proposant des lettres, il a le droit à x erreurs.  
Tout a été réalisé comme indiqué dans le sujet.

### 1.2 Gestion de la difficulté

Menu permettant de choisir la difficulté  
(Facile :10 vies, Moyen :7 vies, Difficile : 5 vies)

### 1.3 Implémentation d'un parser simple lisant une liste de mots

Chaque ligne correspond à un mot, on ne prend en compte que les majuscules dans cette version.

### 1.4 Affichage d'un pendu en fonction du nombre de vies restantes

Dessiné en ASCII progressivement, à chaque fois que le joueur perd une vie.

## 2 Les difficultés rencontrées

### 2.1 Difficulté pour créer plusieurs écrans.

Comment switcher entre plusieurs JPanel ?

J'ai eu quelques problèmes sur le début (Comment accéder au JPanel en dehors de celui-ci ?)

Il suffisait simplement de stocker chaque JPanel dans la même classe et de les appeler.

Alternative : Tous les JPanels dans la même classe.

Le JFrame final contient 3 JPanel

-EcranJeu : jeu du pendu

-EcranDifficile : début de partie, choix de difficulté

-EcranGameOver : fin de partie

On crée un seul ActionListener dans le JFrame (Il y en avait au début dans chaque JPanel, problème pour changer d'écran).

On associe alors chaque bouton à l'action Listener de la JFrame et en fonction de chaque texte du bouton (utilisé comme identifiant), on y associe l'action correspondante.

### 2.2 Liste des boutons

« facile », « moyen », « difficile » : défini le nombre de vie et lance l'écran de jeu.

« A », « Z », « ? » : Teste si la lettre appartient au mot et remplace les \* correspondantes. Sinon on enlève une vie et on affiche le pendu.

« Abandon » : Passe sur l'écran de fin de partie.

« Rejouer », « Quitter » : Relance une partie ou quitte la partie.