



HAL
open science

Développement de fonctionnalités avancées pour Game of Words

Maryam Sadat Nejat

► **To cite this version:**

Maryam Sadat Nejat. Développement de fonctionnalités avancées pour Game of Words. Sciences de l'Homme et Société. 2015. dumas-01236326

HAL Id: dumas-01236326

<https://dumas.ccsd.cnrs.fr/dumas-01236326>

Submitted on 17 Dec 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Développement de fonctionnalités avancées pour Game of Words

Maryam Sadat Nejat

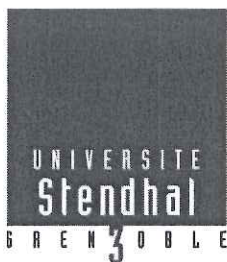
UFR LLASIC

Mémoire de Master 2 professionnel – 20 crédits

Mention : Sciences du Langage – Spécialité : Industries de la langue

Sous la direction de Mathieu Loiseau et Virginie Zampa

Année 2014/2015



Déclaration anti-plagiat
Document à scanner après signature
et à intégrer au mémoire électronique


DECLARATION

1. Ce travail est le fruit d'un travail personnel et constitue un document original.
2. Je sais que prétendre être l'auteur d'un travail écrit par une autre personne est une pratique sévèrement sanctionnée par la loi.
3. Personne d'autre que moi n'a le droit de faire valoir ce travail, en totalité ou en partie, comme le sien.
4. Les propos repris mot à mot à d'autres auteurs figurent entre guillemets (citations).
5. Les écrits sur lesquels je m'appuie dans ce mémoire sont systématiquement référencés selon un système de renvoi bibliographique clair et précis.

NOM : NEJAT

PRENOM : MARYAM SADAT

DATE : 01/09/2015

SIGNATURE : 

Remerciements

Je tiens tout d'abord à remercier Virginie Zampa, directrice du mémoire, pour son suivi à la rédaction de ce mémoire, ses relectures et ses conseils efficaces.

Je remercie également Mathieu Loiseau, coordinateur du lot Gamer et le maître de stage, pour les sprints, ses explications et ses conseils judicieux tout au long du stage.

Toute ma reconnaissance à David Blachon pour sa patience et son soutien précieux.

Sommaire

Introduction.....	8
Partie 1 Présentation du contexte, des problématiques et des méthodes.....	10
Chapitre 1 – Présentation du contexte du stage.....	11
1.1 Structure d'accueil.....	11
1.1.1 Le projet Innovalangues.....	11
1.1.2 Les lots de travail du projet.....	12
1.2. Missions et problématiques.....	13
1.3 Le jeu et sa place dans l'apprentissage.....	13
1.3.1 Un bref historique du jeu et de son rapport à l'apprentissage.....	13
1.3.2 Les apports du jeu dans l'apprentissage.....	14
1.4. Bilan du contexte du stage.....	15
Chapitre 2 – Présentation et état du jeu Game of Wrods.....	16
2.1. Description du jeu.....	16
2.1.1. Présentation du Tabou traditionnel.....	16
2.1.2. Jeux numériques du Tabou.....	16
2.1.3. Game of Words : plus qu'une version numérique du Tabou.....	18
2.1.4. Objectifs du jeu, public visé et accessibilité.....	19
2.2. État du jeu existant.....	20
2.3. Description technique du jeu.....	23
2.3.1. Structure du jeu.....	23
2.3.2. Programmation orientée objet et schéma d'architecture logicielle MVC.....	24
2.3.3. La base de données du jeu.....	25
Chapitre 3 – Méthodologie et objectifs du stage.....	27
3.1. Méthodes agiles.....	27
3.1.1. De la méthode classique aux méthodes agiles.....	27
3.1.2. La méthode Scrum.....	28
3.1.3. La méthode XP.....	29
3.2. La mise en place des backlogs.....	30
3.2.1. Définition des backlogs et user stories.....	30
3.2.2. Backlog sprint.....	31
3.2.3. Le backlog produit du stage.....	31
3.3. Objectifs du développement et outils.....	35
3.3.1. Objectifs du développement.....	35
3.3.2. Outils pour la réalisation.....	35
Partie 2 Solutions proposées pour Game of Words.....	36
Chapitre 4 – Évaluation des niveaux et gestion des langues.....	37
4.1. Modèle de la gestion des langues.....	37
4.1.1. L'usage des langues dans la version initiale.....	37
4.1.2. Le modèle de gestion de langue.....	37
4.2. Réalisation de la gestion de langues.....	40
4.2.1. Le modèle dans la base de données.....	40
4.2.2. L'interaction de l'utilisateur.....	42
4.3. Modèle d'évaluation des points.....	43
4.3.1. Rétribution des actions effectuées.....	44
4.3.2. Estimation des points.....	45

4.3.3. Évaluation globale.....	46
4.4. Mise en place du système de gestion des scores.....	46
4.4.1. Les tables de scores dans la base de données.....	46
4.4.2. Amélioration de la visualisation du score.....	48
4.4.3. Limites et perspectives.....	49
Chapitre 5 – Refonte de l'interface : une interface orientée vers l'utilisateur.....	50
5.1. Le système de notifications.....	50
5.1.1. Les avantages d'un système de notifications.....	50
5.1.2. Notre proposition de système de notifications.....	51
5.1.3. Réalisation du système.....	53
5.2. L'amélioration des parties du devin, du druide et de l'oracle.....	55
5.2.1. Nouvelle visibilité du chronomètre.....	55
5.2.2. L'interface de l'oracle plus progressive.....	57
5.3. Bilan.....	60
Chapitre 6 – Perspectives de travail.....	61
Conclusion.....	64
Bibliographie.....	66
Sitographie.....	67
Annexes.....	68

Table des figures

Figure 1: Illustration du jeu traditionnel du Taboo.....	17
Figure 2: Séquence des tours de jeu d'une partie de Game of Words.....	19
Figure 3: Page de connexion de Game of Words.....	20
Figure 4: Page de connexion adaptée à la largeur de la fenêtre.....	21
Figure 5: Edition du profil.....	21
Figure 6: Arbitrage du Druide.....	22
Figure 7: Partie du Devin.....	22
Figure 8: Table des scores.....	22
Figure 9: Partie de l'oracle.....	22
Figure 10: Illustration des différentes fonctionnalités et de leur organisation.....	24
Figure 11: Illustration du cycle en cascade des méthodes classiques.....	27
Figure 12: Illustration du cycle de développement de la méthode Scrum.....	29
Figure 13: Illustration de la méthode XP.....	30
Figure 14: Illustration de la gestion des langues.....	38
Figure 15: Schéma des niveaux de langue du CECR.....	39
Figure 16: Extrait de la grille de description de la compétence d'écoute pour les niveaux A1 à B1 du CECR.....	40
Figure 17: Extrait de la table des langues parlées dans la base de données.....	41
Figure 18: Extrait de la table des profils qui met en évidence les langues de l'interface et du jeu...	41
Figure 19: Nouvelle page de profil.....	42
Figure 20: Capture d'écran du menu déroulant de la langue de jeu pour l'oracle.....	43
Figure 21: Capture d'écran du menu déroulant de la langue de jeu pour le devin.....	43
Figure 22: Illustration des différents cas d'attribution des points.....	44
Figure 23: Illustration des tables de points et coefficients dans la base de données.....	46
Figure 24: Illustration des phases dans le calcul du score.....	47
Figure 25: Illustration du cas du calcul de score de l'oracle.....	47
Figure 26: Captures d'écran de la page des scores : en haut le classement global puis les classements par langue.....	48
Figure 27: Composition du système de notifications.....	51
Figure 28: Illustration de la table de notifications dans la base de données.....	52
Figure 29: Schéma de fonctionnement de l'affichage des notifications.....	54
Figure 30: Captures d'écran du menu déroulant des notifications et du marqueur de rappel.....	54
Figure 31: Illustration d'une notification d'événement.....	55
Figure 32: Schéma de fonctionnement du chronomètre.....	56
Figure 33: Affichage du chronomètre et illustration de la fonction de "pression".....	57
Figure 34: Message affiché à la place du chronomètre quand le temps est écoulé.....	57
Figure 35: Capture de l'interface d'enregistrement dans la précédente version.....	58
Figure 36: Capture d'écran de l'affichage des instructions dans la nouvelle version.....	59
Figure 37: Capture d'écran du nouvel affichage des cartes.....	60
Figure 38: Avancement de l'envoi du fichier.....	60

Liste des tables

Table 1: Extrait du backlog produit.....	35
Table 2: Coefficients et nombres de points suivant le niveau de jeu, la maîtrise de la langue et la difficulté de la carte.....	45
Table 3: Table des coefficients suivant le rang du score.....	46

Introduction

Le projet Innovalangues a pour objectifs de rechercher des méthodes innovantes pour l'enseignement des langues. Parmi les directions de recherche, le lot GAMER vise à évaluer la place du jeu dans l'apprentissage et à sensibiliser les enseignants de langues à leur usage. Des prototypes de jeux sont développés au sein de ce lot, dont Game of Words¹, une version numérique du Tabou.

Le Tabou appartient aux jeux de défi qui incitent le joueur à faire deviner un mot en le définissant sans avoir recours à une liste de mots interdits. Il est fondé sur l'art de la paraphrase et de l'association, et une grande partie du plaisir qu'il procure tient à l'imprévu et à la complicité de pensée qu'il exige. Ce jeu sollicite la créativité et favorise le développement des stratégies de contournement pour définir le mot à deviner. L'un des objectifs de la didactique des langues est d'amener l'apprenant à mobiliser ses connaissances linguistiques dans un contexte déterminé. C'est pour cela que le tabou peut être intéressant à utiliser afin de perfectionner l'expression orale dans l'environnement scolaire. En ce sens, la plateforme Game of Words représente un outil pertinent dans des cours de langue.

Un autre objectif du lot GAMER vise la réflexion sur des méthodes innovantes d'enseignement, dont l'approche intercompréhensive. Selon Doyé, « l'intercompréhension est une forme de communication dans laquelle chaque personne s'exprime dans sa propre langue et comprend celle de l'autre » (Doyé, 2005). D'un point de vue des compétences linguistiques, l'intercompréhension s'appuie sur l'exploitation des ressemblances et transparences entre les langues. Pour comprendre un texte dans une langue cible, les locuteurs du même groupe linguistique disposent déjà de nombreux éléments pour l'apprentissage au niveau lexical, phonologique, morphologique et syntaxique auxquels s'ajoutent les indices discursifs, textuels ou culturels liés à une communauté linguistico-culturelle déterminée.

Grâce à la possibilité de jouer des parties dans plusieurs langues, Game of Words pourrait être un outil intéressant pour les enseignants adeptes de la pratique intercompréhensive ou les chercheurs désireux d'étudier cette méthode. Game of Words s'affiche ainsi comme un outil aux différents usages : à la fois jeu pour l'apprentissage des langues suivant des méthodes classiques pour améliorer l'expression et la compréhension orale ; et plateforme pour la mise en place de l'intercompréhension des langues.

1 Le jeu est accessible à l'adresse suivante : <http://gamer.innovalangues.net/gameofwords-dev/GOWTEST/Projet-GOW1.1-cleanUp4FF/test/GoWBETA/>

La réalisation du jeu Game of Words est toujours en cours de développement. Ce rapport de stage décrit le travail effectué depuis la première version du prototype. L'organisation du travail a reposé sur la définition de nombreuses fonctionnalités, portant sur des thèmes spécifiques. Le premier concerne justement la gestion de plusieurs langues pour un joueur ainsi que l'intégration du niveau dans ces langues. Le second thème vise à mettre en place une stratégie de gestion des scores qui attribue des points suivant les actions des joueurs. Enfin, le troisième thème de développement concerne l'interface, que nous avons rendue plus ergonomique et claire.

Le travail du stage a aussi porté sur un autre jeu du lot : Magic Word². Il s'agit d'une version numérique du jeu Boggle, le célèbre jeu où l'on doit trouver le plus de mots à partir des 16 lettres de la grille. Comme pour Game of Words, le jeu a une vocation d'enseignement. Les grilles de lettres proposées aux joueurs l'incitent à retrouver le plus de mots cachés et favorisent l'apprentissage du vocabulaire. L'une de nos tâches a été la création de ressources morpho-syntaxiques uniformes dans différentes langues (français, anglais, espagnol, italien et allemand). La disponibilité de ces ressources numériques servira de base au développement d'autres fonctionnalités. Par exemple, elle permettra de générer automatiquement des contraintes pour la conception de grilles de mots.

Pendant le stage, j'ai été associée à un étudiant de ma promotion, David Graceffa. Nous avons travaillé en équipe sur Game of Words et Magic Word, mais nous avons pris chacun la responsabilité d'un jeu. Pour ma part, il s'agit de Game of Words. C'est pourquoi le travail présenté dans ce rapport se concentre sur les problématiques, conceptions, stratégies, méthodes et réalisations de ce jeu. La première itération de développement de Game of words a été effectuée par deux étudiantes de master IDL de l'université Stendhal : *Chloé Cimpello* et *Justine Reverdy*. Nous, David et moi, avons réalisé la deuxième itération.

La présentation du travail effectué dans ce rapport suit le plan suivant. La première partie évoque le contexte du stage et, notamment, revient en détails sur les projets Innovalangues et GAMER ; les missions et problématiques ; et propose une réflexion du jeu dans l'apprentissage. La deuxième section décrit le jeu Game of Words tel qu'il était au début du stage avec, notamment, une description technique de son fonctionnement. La troisième section aborde les méthodes de travail et de gestion de projet suivies au cours du jeu ainsi que le cahier des charges (ou backlog produit) qui résume les fonctionnalités à développer. La quatrième section présente les solutions pour les problématiques de score et de langues en présentant les solutions de manière théorique, puis leurs réalisation dans le jeu. La cinquième section est consacrée aux modifications apportées à l'interface. Enfin, la sixième section aborde les perspectives de travail.

2 Voir capture d'écran en annexe 1. Le jeu est accessible à l'adresse suivante : <http://gamer.innovalangues.net/magicword/V2/Magic-Word-game-master/>

Partie 1

Présentation du contexte, des problématiques et des méthodes

Chapitre 1 – Présentation du contexte du stage

Dans cette première partie, nous présentons les éléments qui déterminent le contexte du stage. Dans un premier temps, la structure d'accueil est décrite, ainsi que le projet Innovalangues, au sein duquel le stage est déroulé. Par la suite, nous présentons les missions et problématiques et la partie se termine par une courte réflexion sur l'intérêt du jeu dans l'apprentissage, thème du travail réalisé pendant ce stage.

1.1 Structure d'accueil



1.1.1 Le projet Innovalangues

Le projet Innovalangues (Innovation et transformation des pratiques de l'enseignement/apprentissage des langues dans l'enseignement supérieur) vise à améliorer des pratiques de l'enseignement des langues. Sur le plan de la formation, le projet poursuit un double objectif : la constitution et la mise à disposition de modèles de formation initiale et continue, ainsi que le soutien au déploiement de ces dispositifs hybrides ou totalement à distance. Pour cela, une formation en autonomie ou tutorée est mise en place sous la forme d'un écosystème numérique en ligne doté d'un espace communautaire pour le partage, de ressources mobilisables et de dispositifs de formation des langues. Ce projet, financé dans le cadre d'un appel à projet IDEFI (initiatives d'excellence en formations innovantes) de 2012 à 2018 est dirigé par Monica Masperi, directrice du LANSAD.

Le projet Innovalangues est porté par la structure LANSAD (Langues pour les Spécialistes d'Autres Disciplines) de l'Université Stendhal. Cette structure offre, une grande variété de formations en langues étrangères. Les enseignements sont organisés en groupes à l'issue d'un test de positionnement en conformité avec les niveaux définis dans le cadre du CECRL³. Ce centre propose également un apprentissage en autonomie en utilisant des outils multimédias en fonction des objectifs pédagogiques de chaque formation.

Afin de dynamiser et d'étendre la démarche de recherche en apprentissage des langues, Innovalangues construit des partenariats avec des institutions et des établissements à l'échelle internationale. Le projet est composé de différents lots, que nous allons présenter dans le point suivant.

3 Cadre européen commun de référence pour les langues

1.1.2 Les lots de travail du projet

Suivant l'article de Masperi et Quintin (2014), le premier lot est représenté par l'acronyme *SELF* (Système d'Évaluation en Langues à visée Formative). Ce lot vise à proposer des modules d'évaluation diagnostique à visée formative. L'équipe a mené une réflexion sur les questions de « positionnement » et de « certifications » pour faire avancer les recherches sur l'évaluation des compétences en langues.

Le second lot est identifié par l'acronyme *THEMPPO* (*Thématique Prosodie & Production Orale*). Il a comme objectif le développement des compétences prosodiques et de production orale en langue étrangère. Il se situe dans le cadre d'une recherche et développement d'un parcours d'apprentissage personnalisé (PAP) et son but est de mener une action de rénovation et innovation pour améliorer les compétences linguistiques des étudiants afin qu'ils puissent atteindre un niveau B2 certifié du CECRL.

L'équipe du lot PARCOURS se concentre sur l'environnement numérique personnalisé d'apprentissage (ENPA) qui accueillera les réalisations des autres lots. L'équipe développe également des outils de communication et de collaboration qui seront intégrés à l'ENPA. A terme, l'ensemble des outils devrait faciliter l'échange et le partage entre enseignants, tuteurs et apprenants.

Le lot COCA (Compétence Orale : Conception et Assistance) a pour objectif de proposer une solution pour les difficultés liées à la compréhension orale chez des apprenants. Il vise à concevoir et réaliser un éditeur de son en élaborant un référentiel de compétences pour la compréhension orale et une typologie des activités.

Pour terminer, nous présentons le lot GAMER (*Gaming Application for Multilingual Educational Resources*), dans lequel s'inscrit ce stage. Il vise deux objectifs liés au jeu et à l'apprentissage. L'objectif du lot est de créer des jeux pour sensibiliser les enseignants à l'usage des jeux dans les cours de langues et mener une réflexion sur la place du jeu dans une plateforme d'apprentissage. Dans ce cadre, GAMER travaille sur différents chantiers : jeux en ligne, Check Your Smile (en partenariat avec Toulouse), Game2learn (en partenariat avec grenoble INP). Le chantier dans lequel s'inscrit ce stage concerne les jeux en ligne et plus particulièrement Magic Word, jeu inspiré du Boggle et Game of Word, jeu inspiré de Tabou.

Pour compléter la présentation du contexte du stage, nous décrivons dans la suite les missions et problématiques au cours du stage.

1.2. Missions et problématiques

Ce rapport est consacré au développement de nouvelles fonctionnalités sur Game of Words. Le développement du jeu est géré par l'outil de versionnage Git et le service d'hébergement de code GitHub ; et grâce à une méthode agile de gestion de projet, que nous définirons dans la suite. A ce titre, un *backlog produit*⁴ résume l'ensemble des fonctionnalités à développer. Il est alimenté par les requêtes des différents partenaires (client du produit et équipe de développement essentiellement).

La première tâche du stage a été de compléter le *backlog produit*. En effet, suite aux tests réalisés sur la version initiale du jeu, nous avons découvert des bogues divers et variés qu'il fallait corriger. Par la suite, nous avons défini un ensemble de tâches à réaliser, à partir des requêtes du *backlog produit* et suivant l'importance des besoins des clients, afin de développer des fonctionnalités complètes. Trois directions de développement du jeu sont apparues, qui constituent les 3 axes principaux d'implantation effectués :

- l'absence d'un système de gestion des scores : Game of Words est avant tout un jeu avec une portée éducative, la présence d'un système de scores est donc justifiée ;
- la gestion limitée des langues de jeu : la représentation de la langue dans le jeu se limite à une langue de jeu qui est aussi la langue de l'interface et qui doit être choisie parmi le français ou l'anglais ; il nous paraît nécessaire que le jeu propose d'autres langues ;
- l'interface soignée mais basique du jeu ; en particulier, l'utilisateur a peu de moyens d'interaction et d'information : après qu'il a créé une carte ou un enregistrement, il n'a aucune idée du devenir de ses productions ; la gestion du profil est très sommaire ; il nous paraît nécessaire de rendre l'interface plus dynamique et d'offrir plus de contrôle à l'utilisateur.

Pour compléter la présentation du contexte du stage, la section suivante aborde le contexte thématique en proposant une brève réflexion sur le lien entre le jeu et l'apprentissage.

1.3 Le jeu et sa place dans l'apprentissage

1.3.1 Un bref historique du jeu et de son rapport à l'apprentissage

Historiquement, le jeu avait mauvaise réputation, comme le résume Racha Hallal dans son

4 Dans la gestion de projet informatique, le backlog produit contient, sous forme de liste, les choses que le client veut que l'équipe réalise.

mémoire de Master (Hallal, 2014). Parce qu'il est pratiqué par des enfants, le jeu est doté de valeurs qui leur sont associées, des êtres fragiles et avec peu d'expérience. Le jeu est perçu comme un moyen de récréation et s'oppose à l'idée du travail. Puis le jeu connaît un regain d'intérêt durant la Renaissance où il commence à être vu comme une ruse pédagogique pour faire travailler les enfants.

A la fin du XIX^e siècle, le courant de l'éducation nouvelle, fondée en partie sur l'Émile de Rousseau, prône l'apprentissage par l'expérience personnelle et la participation active des individus dans l'apprentissage. Ainsi, le jeu commence à trouver sa place dans l'apprentissage. Plus récemment, au début des années 90, les pratiques d'enseignement des langues étrangères ont vu apparaître une nouvelle approche dite actionnelle qui considère l'apprenant d'une langue comme un acteur social et la communication dans une langue comme un moyen d'agir. L'approche consiste à mettre ce dernier en situation réaliste face à un problème pour qu'il développe non seulement des compétences langagières mais aussi communicatives. Dans son livre, Evelyne Rosen (2009) considère que la pédagogie ne se fait plus par une simulation imaginaire dans des cours de langues mais par l'emploi de la langue comme si nous étions dans une obligation de communiquer dans des circonstances réelles.

1.3.2 Les apports du jeu dans l'apprentissage

De nombreuses définitions du jeu existent, plus ou moins communes. Cependant, celle qui nous intéresse particulièrement ici a été élaborée par Caillois (1958). Celui-ci décrit le jeu comme une activité libre, séparée, incertaine, improductive, réglée et fictive. Par ses caractères séparé et fictif, le jeu crée une réalité distincte, limitée dans le temps et l'espace qui est propice à l'application de la méthode actionnelle. En outre, l'objectif du jeu peut être identifié au problème auquel l'apprenant est confronté dans l'emploi de cette méthode. Le jeu pousse l'apprenant à agir vers la réalisation de cet objectif. Ainsi, le jeu selon Caillois semble particulièrement bien adapté à l'approche actionnelle.

Cependant, il faut également se demander si le jeu est un bon outil pour l'apprentissage. D'après Sauv   et al (2007), le jeu « cr  e des conditions favorables    l'apprentissage [et] motive l'apprenant ». Or la motivation est probablement le facteur le plus important pour l'apprentissage. La motivation est notamment renforc  e par le fait que le jeu « d  formalise l'apprentissage scolaire » selon Gilles Broug  re (2005), en d  pla  ant la focalisation de l'apprenant vers l'objectif du jeu et la strat  gie.

Un autre avantage offert par le jeu est celui d'autoriser l'erreur grâce à la réalité fictive et à l'issue incertaine du jeu, relevées par Caillois dans sa définition. Or si l'erreur est acceptée, l'apprenant est plus enclin à s'investir et essayer, ce qui favorise l'apprentissage.

Enfin, le jeu favorise la communication. Cet aspect est formalisé par le système de classification de jeux ESAR⁵ qui propose 6 critères. Plus particulièrement, l'un d'eux définit la caractéristique sociale du jeu suivant que l'activité est individuelle, associative, compétitive ou coopérative. Le système propose également un critère langagier qui définit les compétences employées (production ou compréhension, à l'oral ou à l'écrit). L'un des mécanismes du jeu qui favorise l'apprentissage est la constante rétroaction qui résulte de l'interaction de l'apprenant avec les autres joueurs. Cette rétroaction, combinée à la motivation créée par le jeu, pousse l'apprenant à s'engager et à réessayer.

Ainsi, le jeu semble aujourd'hui parfaitement s'intégrer à l'approche actionnelle employée pour l'apprentissage des langues. Il ne souffre plus des valeurs auxquelles il était historiquement attaché. Au contraire, il fait l'objet de nombreux travaux pour tenter de le caractériser et de justifier sa place dans l'enseignement. Concernant l'apprentissage des langues, plusieurs de ses vertus ont été relevées telle que le maintien de la motivation chez l'apprenant, l'incitation à la communication et la dédramatisation de l'erreur.

1.4. Bilan du contexte du stage

Dans cette première section, nous avons présenté le thème du stage : le jeu et l'apprentissage. Nous avons présenté quelques idées afin d'illustrer ce propos et montrer que le jeu a sa place dans l'apprentissage, notamment d'une langue. Nous avons également introduit la structure d'accueil et le projet à l'origine du besoin auquel répond le stage. Ainsi, Innovalangues est un projet d'envergure qui dispose d'un financement important et qui s'étend sur une période longue. En son sein, plusieurs thématiques de travail, regroupées en lots, sont représentées. Le lot Gamer mène une réflexion sur la place du jeu dans l'apprentissage et vise à sensibiliser les enseignants à l'usage du jeu dans les cours de langue. Plusieurs projets informatiques ont été lancés dont Game of Words et Magic Word, deux jeux éducatifs. Ce rapport se concentre sur le développement de 3 fonctionnalités du jeu Game of Words : la gestion des scores, la gestion des langues et l'amélioration de l'interface pour la rendre plus interactive. La section suivante décrit précisément le jeu Game of Words, au moment du début du stage.

5 D'après le site Ludothèque ô fil du jeu, ESAR, mis au point en 1980 par Denise Garon, est aujourd'hui l'un des systèmes de référence pour la classification des activités ludiques. Ce système les classe en 4 grandes catégories : Les jeux d'Exercice, Les jeux Symboliques, Les jeux d'Assemblage et Les jeux de Règles.

Chapitre 2 – Présentation et état du jeu Game of Words

Cette partie est dédiée à la présentation du jeu Game of Words dans sa version disponible au début du stage. La première section introduit le jeu en présentant le Tabou, qui est à l'origine du jeu. Puis, nous étudierons quelques versions numériques du Tabou avant d'introduire la solution retenue dans Game of Words. La section se termine par la présentation d'informations factuelles sur le jeu, notamment autour de l'accessibilité de celui-ci. La section suivante présente l'état du jeu au début du stage. Enfin, la troisième section entreprend une description technique du jeu. L'organisation du code y est décrite ainsi que les fonctionnalités, au moyen de concepts tels que la programmation orientée objet ou le schéma Modèle-Vue-Contrôleur. Les concepts énoncés seront présentés par la suite. La section se termine par la description de la base de données, principale ressource du système du jeu.

2.1. Description du jeu

Cette première section introduit Game of Words, des principes du jeu jusqu'à la solution choisie pour la version numérique et ses objectifs.

2.1.1. Présentation du Tabou traditionnel

Le Tabou est un jeu de société créé en 1990 par Brian Hersch et Bernard Bougrèle et édité par la société Hasbro. Le jeu se joue en deux équipes, avec au moins 4 joueurs. Un tour de jeu consiste, pour un joueur d'une équipe, à faire deviner le plus de mots aux autres joueurs de son équipe. Pour cela, le joueur lit le mot secret sur une carte et prend connaissance des 5 mots interdits. Il doit alors élaborer une définition sans prononcer ces mots, ni des mots dérivés du mot secret et des mots tabous. Pendant ce temps, l'équipe adverse vérifie le respect du temps imparti au moyen d'un sablier et surveille qu'aucun mot interdit n'est prononcé. La figure 1 illustre une boîte du jeu traditionnel. Dans la suite, nous appellerons *devin*, le joueur qui devine le mot ; *oracle* le joueur qui fait deviner ; et *druide* le joueur qui vérifie le respect du temps et des mots.

2.1.2. Jeux numériques du Tabou

Le Tabou se décline aujourd'hui en différentes versions numériques, accessibles depuis un ordinateur ou même un smartphone. Il existe de nombreuses versions en ligne, que nos deux prédécesseurs Chloé Cimpello et Justine Reverdy avaient résumées dans leur rapport de projet professionnel de Master 2 (Cimpello et Reverdy, 2015). Ainsi, on peut trouver le *Taboo Game*, le *Play Taboo* ou encore le *Online Taboo*. Tous les 3 sont accessibles depuis un site web et permettent

ainsi d'être joués depuis n'importe quel terminal relié à Internet. Le *Taboo Game* propose de jouer avec des cartes générées automatiquement ou de créer ses propres cartes. Cependant, il arrive que certains mots générés n'existent pas. En outre, il manque un système de gestion qui permettrait de gérer des parties entières. Le *Play Taboo* repose sur une base de cartes. Il embarque un chronomètre pour les tours de jeu et propose un système de points, ce qui le rendent attrayant. Cependant, cette version ne propose pas d'arbitrage des définitions proposées. Enfin, *Online Taboo* est un jeu collaboratif où les joueurs sont invités à jouer une partie ou créer une carte. En outre, le jeu embarque un système de gestion du score et se décline en plusieurs langues, grâce aux contributions des joueurs. La qualité et la quantité des cartes dans les différentes langues est très inégale : on trouve parfois beaucoup de fautes d'orthographe et le français n'est illustré que par 17 cartes.



Figure 1: Illustration du jeu traditionnel du Taboo ; source : https://upload.wikimedia.org/wikipedia/commons/1/19/Taboo_02.jpg

Des versions sur smartphone existent également, sous forme d'applications. On notera le *Taboo!*, décliné également pour le système Windows, qui repose sur une base de données de plus de 70 000 mots. L'application est asynchrone et intègre un système de gestion de scores. Cependant, le jeu ne peut être joué qu'en anglais.

L'étude précédente met en avant certaines caractéristiques intéressantes des versions numériques mais permet surtout de soulever un certains nombres de problème par rapport à ce que nous souhaitons :

- Nous avons ainsi relevé le problème de la langue de jeu, souvent l'anglais (tous les jeux cités à l'exception du *Online Taboo* ne se jouent qu'en anglais) qui limite l'accessibilité du jeu ; or nous voulons que le jeu puisse être réalisable dans le plus grand nombre des langues enseignées au LANSAD.
- De même, la gestion de la base de cartes est un vrai problème. Nous avons besoin d'une grande base, composée de cartes de qualités. A ce sujet, le jeu *Taboo!* est remarquable par sa grandeur, avec sa base de 70000 cartes. Lorsque la base est faible, une solution, employée notamment par les jeux *Online Taboo* et *Taboo Game*, consiste à faire créer les cartes par les joueurs. Bien que théoriquement séduisante, cette solution requiert une vérification importante afin de proposer des cartes de qualité, exemptes de fautes d'orthographe. Une autre solution consiste à générer des cartes automatiquement par un système. Dans le cas du *Taboo Game*, le système de génération automatique est très limité puisque certaines cartes proposées contiennent parfois des mots inexistantes ;
- Le système de gestion de jeu doit permettre le bon déroulement d'un tour de jeu avec notamment le décompte du temps (comme pour le *Play Taboo*) et la vérification des mots interdits. En outre, un système de gestion de points (*Taboo!*) est un avantage pour donner plus d'importance à la partie.
- Enfin, le jeu numérique doit permettre de jouer même lorsque l'on est seul. La solution asynchrone du *Taboo!* est une piste intéressante.

2.1.3. *Game of Words : plus qu'une version numérique du Tabou*

La solution choisie pour *Game of Words* repose sur l'asynchronisme des parties. Cela consiste à découper la partie en séquences successives indépendantes qui peuvent être jouées par différents utilisateurs quand ils le pourront mais en respectant l'ordre des séquences. La présentation des différentes phases du jeu comme des modes de jeu indépendants donne l'illusion à l'utilisateur de jouer de manière synchrone.

Cette approche offre plus de liberté en permettant à un joueur isolé de jouer n'importe quel rôle. En outre, un même enregistrement pourra être réutilisé par plusieurs joueurs dans des parties distinctes, ce qui permet d'augmenter le nombre de parties possibles. En outre, la combinaison de l'asynchronisme et de l'enregistrement des différentes étapes d'une partie (enregistrement, statut de la validation, propositions du devin) permet la résolution des litiges plus facilement. Par exemple, en cas d'invalidation d'un enregistrement par un druide, l'oracle pourra contester et demander à faire écouter l'enregistrement. A l'inverse, la réalisation synchrone requiert un contrôle strict des

diffusions et échanges entre les différents joueurs : les joueurs d'une même partie doivent pouvoir interagir simultanément pour jouer.

Dans le cas de Game of Words, l'asynchronisme est possible grâce à la séparation des trois rôles du devin, de l'oracle et du druide présentés plus haut. Ces rôles sont habituellement simultanés, cependant, il est possible de les découper suivant une séquence, c'est l'idée de la partie asynchrone.

Le démarrage d'un tour de jeu se fait par la définition du mot à trouver par l'oracle. Cette définition est soumise à l'appréciation du druide, qui la valide ou la sanctionne. Dans le cas favorable, la définition est soumise au devin, qui tente de trouver le mot secret. Cette séquence des 3 étapes est résumée dans la figure 2.

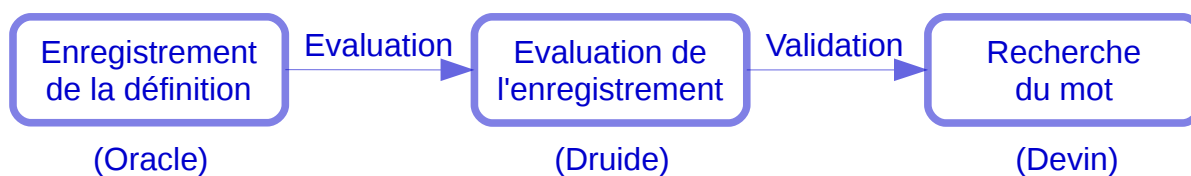


Figure 2: Séquence des tours de jeu d'une partie de Game of Words

2.1.4. Objectifs du jeu, public visé et accessibilité

Dans un premier temps, le public de ce jeu sera constitué des utilisateurs de la plateforme Innovalangues, principalement des étudiants et leurs professeurs. La mise en place des trois rôles est à la fois un avantage pour l'accroissement du jeu mais aussi un atout pour l'apprentissage. La méthode la plus naturelle de génération de cartes passe par les utilisateurs et c'est ce que permet l'action de création de carte par le druide dans le jeu. Par ailleurs, chaque rôle incite l'apprenant à mettre en œuvre des compétences langagières spécifiques. La génération d'un enregistrement sollicite la production orale. La validation d'un enregistrement stimule la compréhension orale. Enfin, l'action de deviner le mot favorise la compréhension orale et l'expression écrite. En outre, le jeu requiert et enrichit le niveau de vocabulaire de l'apprenant. L'utilisation du jeu est complémentaire à d'autres activités d'apprentissage d'une langue.

Concernant la disponibilité du jeu, il sera intégré à la plateforme Innovalangues. Il sera donc accessible depuis n'importe quel appareil connecté à Internet et doté d'un navigateur. Les seules contraintes sont d'avoir un appareil muni d'un microphone et d'un système d'écoute (haut-parleurs intégrés ou prise de sortie pour brancher un casque ou des enceintes) puisqu'il est nécessaire d'enregistrer et écouter des définitions.

2.2. État du jeu existant

La version présentée dans cette partie est la première réalisée par nos deux prédécesseurs Chloé Cimpello et Justine Reverdy, deux étudiantes de ma promotion. La structure du code décrite a été réalisée et la base de données est à l'image de celle présentée. Les fonctionnalités du jeu sont celles illustrées par la figure précédente. Game of Words est accessible en français et en anglais.

Un module de génération automatique de carte a été créé et ajouté aux parties de jeu de l'oracle. La génération est réalisée grâce aux ressources du site jeuxdemots.org/rezo.php et à un *crawler*⁶. A partir d'une requête soumise, le site propose une liste de mots associés, classés par importance de la relation sémantique : les premiers mots de la liste ont un lien plus fort avec le mot de la requête que les derniers mots. Ensuite, une sélection est opérée sur ces mots associés pour choisir les mots tabous.

L'interface est soignée et tire profit de l'environnement Bootstrap⁷ pour la génération d'interface simple et dynamiques. Nous présentons plusieurs captures d'écran prises à partir de la version existante au début du stage. Les figures 3 et 4 illustrent l'adaptation de l'affichage à la taille de la fenêtre : la barre de menu est remplacée par un bouton dans la figure 4.



Figure 3: Page de connexion de Game of Words

6 Un *crawler* est un robot d'indexation conçu pour explorer un ensemble de documents et collecter des ressources. Pour plus de détails, voir fr.wikipedia.org/wiki/Robot_d%27indexation

7 Pour plus de détails, voir www.getbootstrap.com

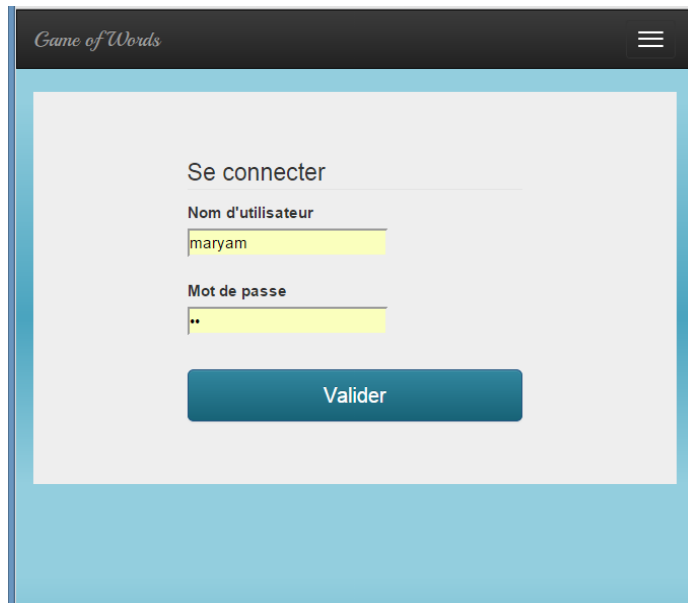


Figure 4: Page de connexion adaptée à la largeur de la fenêtre

La figure 5 illustre la page d'édition du profil et montre en particulier le peu d'informations et réglages possibles dans cette page. On y retrouve les informations de base du compte : le nom d'utilisateur, l'adresse e-mail et le mot de passe ainsi que la langue de jeu.

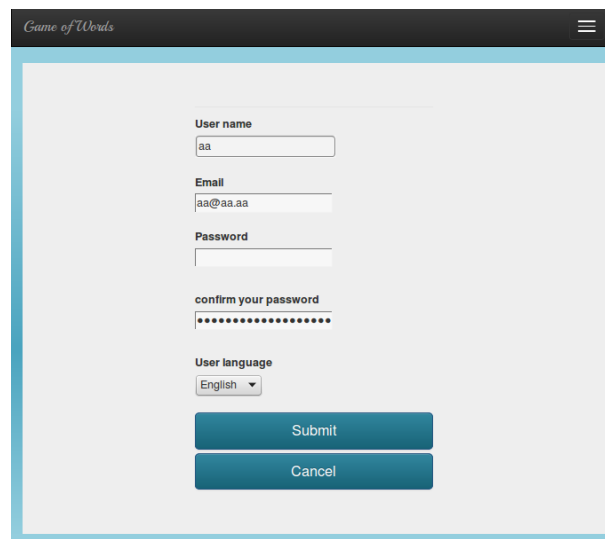


Figure 5: Edition du profil

Les figures 6, 7 et 9 illustrent les trois types de parties possibles suivant les rôles du druide, du devin et de l'oracle. Comme on peut le constater, les différentes interfaces sont basiques. Les informations y sont présentées de manière linéaire, sans mise en valeur des éléments importants et sans indication sur la manière de jouer. Enfin, la figure 8 affiche la table des scores avec un simple tableau des scores suivant les rôles.

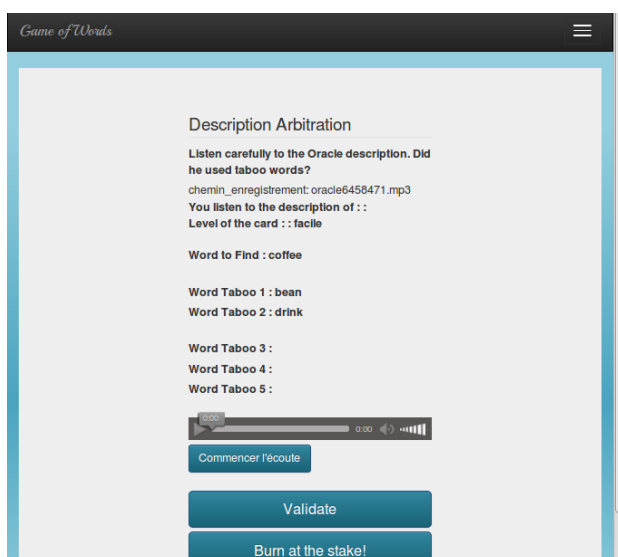


Figure 6: Arbitrage du Druides

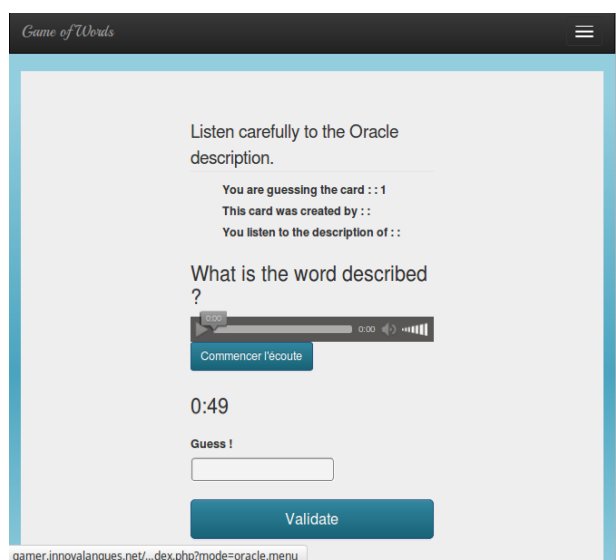


Figure 7: Partie du Devin



Figure 9: Partie de l'oracle

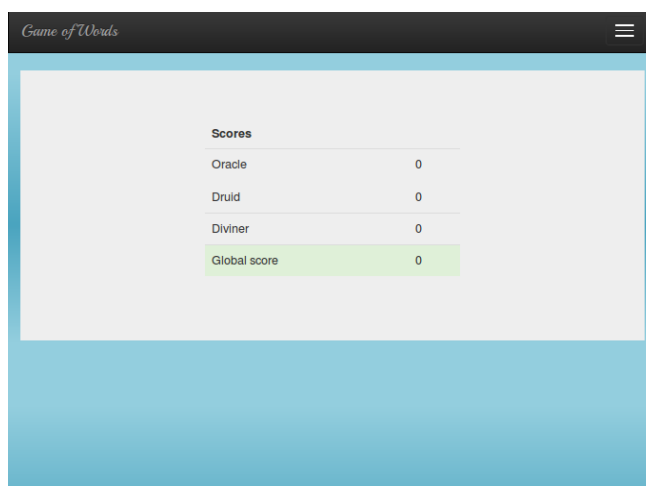


Figure 8: Table des scores

2.3. Description technique du jeu

Dans cette section, nous présentons la structure du code du jeu et la base de données associée au système. Afin de mieux comprendre la structure du code, nous décrivons brièvement les concepts de la programmation orientée objet (POO) et du patron d'architecture logicielle Modèle-Vue-Contrôleur (MVC) après la structure du jeu.

2.3.1. Structure du jeu

La structure du jeu repose sur la combinaison de la programmation orientée objet et du schéma MVC. D'abord, les principales fonctionnalités ont été identifiées et séparées. Celles-ci se composent des actions suivantes :

- inscription d'un nouveau joueur,
- connexion avec un profil existant,
- édition d'un profil existant,
- les différents tours de jeu de l'oracle, du devin et du druide.

Concernant les tours de jeu, la première version permet de jouer en tant qu'oracle pour enregistrer la définition d'une carte créée automatiquement, sélectionnée aléatoirement ou sélectionnée à partir de son identifiant. Le druide peut créer une carte ou arbitrer un enregistrement. Enfin, la tâche du devin consiste à trouver le mot secret et se solde par un succès ou un échec. Chacune des fonctionnalités ci-dessus est associée à une vue particulière. En outre, chaque fonctionnalité est également rempli par un contrôleur. Ainsi, lorsque l'utilisateur choisit une fonctionnalité du jeu, le système crée une instance du contrôleur associé qui à son tour va générer la vue de la fonctionnalité.

Le passage d'une fonctionnalité à une autre est réalisé par le fichier *index.html*, qui est le « point de passage » lorsqu'une fonctionnalité se termine. Cette page affiche la vue d'accueil du site lorsqu'on est connecté et initialise le contrôleur adéquat en fonction du choix de l'utilisateur.

Un modèle, représenté par la classe *user*, contient les informations du joueur en cours. Dans cette classe, on retrouve les informations sauvegardées dans la base : nom, prénom, e-mail, langue, pseudo et identifiant associé au profil. Le schéma de la figure 10 illustre l'organisation des fonctionnalités.

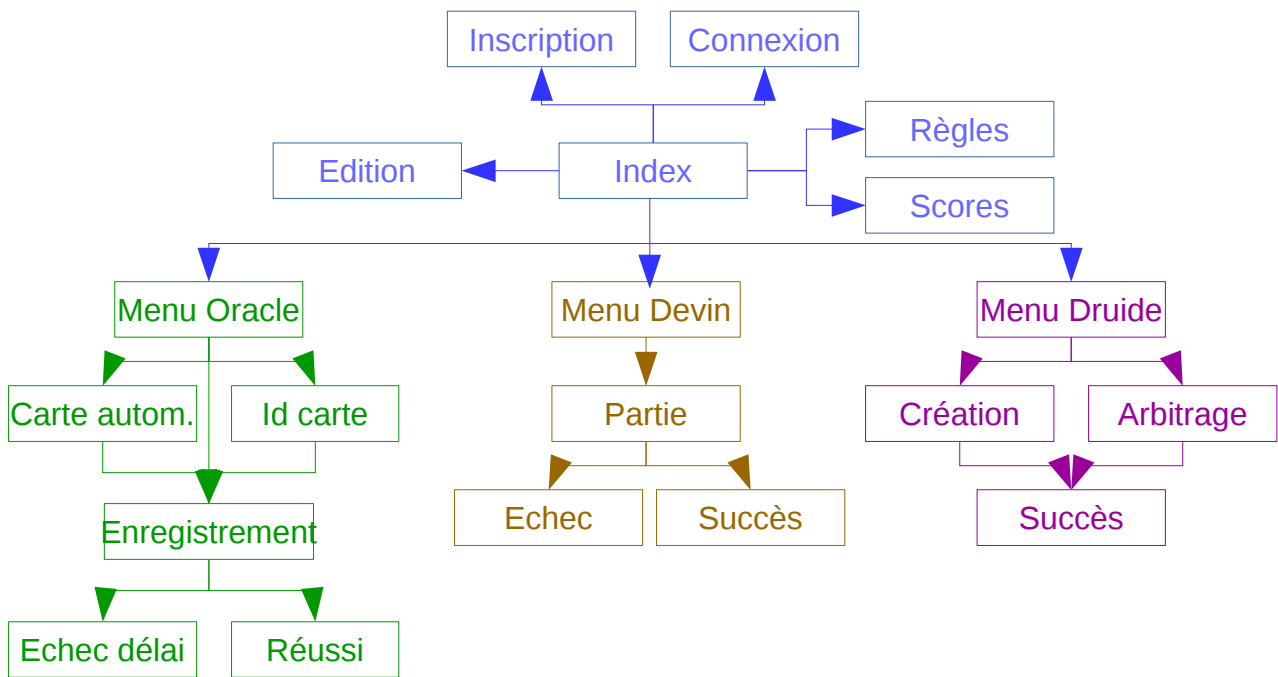


Figure 10: Illustration des différentes fonctionnalités et de leur organisation

2.3.2. Programmation orientée objet et schéma d'architecture logicielle MVC

POO⁸ – La programmation orientée objet est un paradigme de programmation informatique élaboré dans les années 1960. Il consiste à organiser le code suivant des blocs qui représentent des concepts. Une telle organisation existe en programmation procédurale, en définissant une structure et des fonctions qui la prendront en paramètre. La POO va plus loin avec les principes mis en place qui donnent une certaine cohérence à l'objet et ses fonctions.

Un objet peut être assimilé à une telle structure composée de variables et complétée par un ensemble de fonctions qui définissent les actions possibles de l'objet. Les variables qui composent l'objet sont les attributs (ou champs) et les fonctions sont aussi appelées les méthodes de l'objet.

Le principe d'encapsulation de la POO définit la classe, comme la structure globale (les attributs et les méthodes). L'*objet* est une instance de cette classe. Ainsi, la classe peut être assimilée à un « moule » qui servirait à créer des objets aux attributs et propriétés identiques. L'encapsulation est à l'origine du cloisonnement de l'objet, qui sépare son contexte de celui qui l'entoure. Autrement dit, l'encapsulation permet de protéger l'accès à certains attributs ou méthodes depuis l'extérieur.

⁸ Pour plus d'informations consulter le site à l'adresse: <http://tdinfo.phelma.grenoble-inp.fr/2Apo0/>

Le second principe de la POO est l'héritage. La POO permet une mise en commun de fonctionnalités par la hiérarchisation de celles-ci. A l'origine, il y a une classe unique qui définit des attributs et méthodes communes à l'ensemble des classes. Cette classe est ensuite dérivée pour créer des classes *enfants* qui se spécialisent avec des attributs et méthodes particulières et qui peuvent, à leur tour, être à l'origine de classes *enfants*.

MVC⁹ – Le patron d'architecture logicielle (*design pattern* en anglais) représente un ensemble de conventions en développement informatique qui consiste à séparer les différentes structures et fonctionnalités suivant leur rôle. On distingue trois rôles différents :

- le modèle représente les données ;
- la vue présente les informations à l'utilisateur et permet l'interaction ;
- le contrôleur prend les décisions, répond aux événements (par exemple ceux générés par l'interaction de l'utilisateur) et assure la synchronisation des éléments (en particulier entre la vue et le modèle).

Le fonctionnement habituel commence par une requête réalisée par l'utilisateur, soumise *via* la vue et interceptée par le contrôleur. Celui-ci l'analyse et entreprend les actions nécessaires. Vraisemblablement, la vue devra être mise à jour. Pour cela, certaines données sont nécessaires et le contrôleur les récupère *via* le modèle. Une fois les données récupérées et traitées le cas échéant, le contrôleur les transmet à la vue pour l'affichage.

2.3.3. La base de données du jeu

Le modèle de l'utilisateur est une représentation temporaire de certaines informations par le système. Toutes les informations ne sont pas chargées pendant l'utilisation, mais toutes sont accessibles dans la base de données. Celle-ci est composée de plusieurs tables qui regroupent les informations d'un même concept. Au total, il y en a six dans la base :

- *user* : les informations des utilisateurs y sont renseignées ;
- *carte* : les descriptions des cartes y sont enregistrées (mot à deviner, mots tabous, niveau de difficulté) ainsi que des informations pratiques comme l'identifiant du joueur qui l'a créée, la langue des mots où la date de création ;
- *enregistrement* : les fichiers sonores contenant les définitions d'une carte sont répertoriés dans cette table ; le chemin de l'enregistrement, la carte à laquelle il est associé ou l'identifiant du joueur qui l'a créé sont accessibles dans cette table ;

9 Pour plus d'informations, consulter le site à l'adresse: <http://wiki.innovalangues.net/index.php/MVC>

- *partie* : cette table représente les tours de jeu des joueurs qui tentent de deviner les mots ; les informations les plus importantes sont l'identifiant du joueur, celui de l'enregistrement qu'il le statut de la partie est indiqué (réussi, non-réussi ou en cours de jeu) ;
- *score* : la sauvegarde des différents scores des joueurs est effectuée dans cette table ; pour chacun, on trouve les scores des trois rôles et le score global ;
- *arbitrage* : le résultat de la validation d'un enregistrement est enregistré dans cette table ; on y trouve l'identifiant de l'enregistrement, le statut associé (validé ou non), l'identifiant du joueur qui l'a évalué et la date de l'évaluation ;

La communication du système avec la base de données se fait au moyen de requêtes SQL. Les principales requêtes utilisées sont les suivantes :

- **SELECT** : pour récupérer les informations depuis une table ; il est possible de spécifier les champs souhaités et de filtrer les lignes suivant une condition (mot-clé WHERE)
- **INSERT** : cette requête crée une nouvelle entrée dans la table ; il est possible de spécifier les valeurs de certains champs ;
- **UPDATE** : cette requête met une ou plusieurs entrée pré-existantes avec les valeurs des champs spécifiées ; il est possible de mettre à jour toutes les entrées d'une table ou seulement certaines grâce au filtre appliqué avec le mot-clé WHERE ;

La description de la base de données termine la seconde partie qui a présenté le jeu Game of Words, dans son ensemble, au moment du début du stage. La section suivante présente la méthode de gestion de projet appliquée à ce stage et le cahier des charges précis pour le développement.

Chapitre 3 – Méthodologie et objectifs du stage

La troisième partie de ce rapport décrit la méthode de gestion de projet appliquée pendant ce stage et rapporte le cahier des charges du développement des fonctionnalités, présenté sous forme d'un *backlog produit*. En ce sens, cette partie est une transition dans le rapport entre l'état existant du jeu, décrit dans la partie précédente, et les nouvelles fonctionnalités, décrites dans les parties suivantes. Dans la mesure où nous avons participé à l'élaboration du *backlog produit*, celui-ci constitue le premier résultat obtenu pendant le stage. Avant de le présenter, une première section est consacrée à la description des méthodes de gestion de projet dites agiles. La description est illustrée par deux exemples, les méthodes XP et Scrum. La dernière a été employée pendant le stage.

3.1. Méthodes agiles¹⁰

3.1.1. De la méthode classique aux méthodes agiles

Un bon chef de projet doit maîtriser les techniques de gestion de projet, de management d'équipe et avoir un bon relationnel lors des échanges avec le client. Son objectif est de pouvoir mener son projet à terme en ayant recours aux méthodes de gestion de projet.

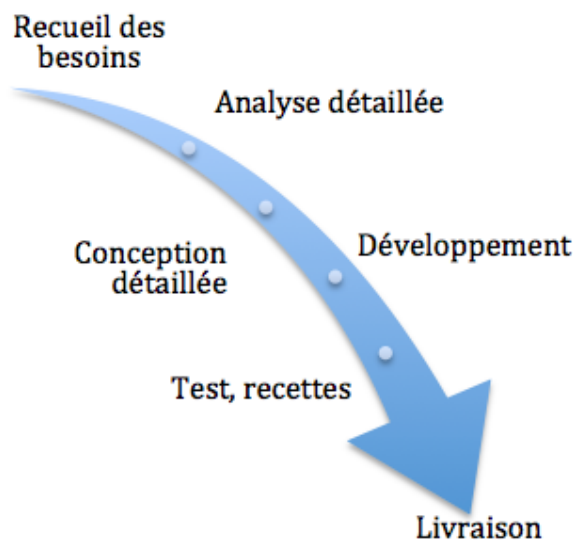


Figure 11: Illustration du cycle en cascade des méthodes classiques (source <http://www.access-dev.com/access-dev/la-gestion-de-projet-methodes-classiques-vs-methodes-agiles/>)

Les méthodes classiques de gestion de projet (dites prédictives) sont fondées sur des cycles en cascade (voir figure 11). Ces cycles se composent de plusieurs étapes du recueil des besoins des clients à la livraison en passant par des phases d'analyse, développement et test. La figure ci-

¹⁰ Pour plus d'informations, consulter le site à l'adresse:
http://wiki.innovalangues.net/index.php/Méthode_de_développement_agile

dessous illustre la cascade des étapes. Dans cette méthode le passage d'une phase à l'autre est réalisé après la validation de l'étape. La gestion des imprévus n'est pas facile car les phases précédemment validées doivent alors être remises en cause. Ainsi, les risques d'erreur ou d'échec augmentent à mesure que le projet avance. Afin d'anticiper au mieux les imprévus, il incombe au chef de projet de préparer au mieux les différentes phases, en rédigeant des documents précis par exemple, tel que le cahier des charges. Un des inconvénients de cette méthode est la faible participation du client dans la phase de développement. Après la rédaction des documents et la planification des étapes, ce dernier n'a plus de rôle jouer dans la réalisation, avec comme risque la présence de défauts dans la livraison que le client aurait pu détecter avant s'il avait été impliqué.

À cause des inconvénients abordés, des experts en génie logiciel ont mis au point un manifeste¹¹, au début des années 2000, qui définit 4 valeurs pour la gestion de projet et sur lesquelles s'appuient les méthodes agiles :

- les individus et leurs interactions plus que les processus et les outils,
- des logiciels opérationnels plus qu'une documentation exhaustive,
- la collaboration avec les clients plus que la négociation contractuelle,
- l'adaptation au changement plus que le suivi d'un plan.

Les méthodes agiles permettent la mise en place d'un développement itératif, qui consiste à découper le projet en étapes. Ces étapes, définies avec le client, indiquent les objectifs à atteindre à la fin de l'étape. Les objectifs sont des fonctionnalités à développer, suivant l'interprétation du deuxième principe du manifeste. La validation de chaque étape est réalisée avec le client.

La segmentation du développement en phases intermédiaires et la planification à l'échelle d'une phase a de nombreux avantages. Elle réduit l'effort d'anticipation du chef de projet. Le client est régulièrement impliqué dans le développement et peut faire évoluer sa demande. Les imprévus sont détectés plus tôt et les changements autorisés.

3.1.2. La méthode Scrum

De nombreuses méthodes agiles existent. Dans la suite, nous présentons la méthode Scrum et la méthode XP. La méthode Scrum (mêlée en anglais) propose un développement dynamique et continu. Ainsi, le cahier des charges initial est peu fourni et les fonctionnalités sont définies au fur et à mesure. La méthode nécessite trois rôles :

11 Manifeste consulté sur le site du Club Agile, accessible à l'adresse: <http://www.agilemanifesto.org/iso/fr/>

- le *product owner* : le client ou son représentant,
- le *scrum master* : le garant de l'application de la méthode,
- l'équipe de développement.

L'unité de la méthode est le *sprint* qui représente une itération du développement. Les fonctionnalités du *sprint* sont définies au cours d'une réunion de planification avec les 3 parties et détaillées dans le *backlog sprint*. Au cours d'un *sprint*, des réunions quotidiennes (*daily scrum*) ont lieu au sein de l'équipe de développement afin de faire le point sur l'avancement et les difficultés rencontrées. A l'issue d'une période de *sprint*, une réunion se tient avec toutes les parties afin de faire le point et de définir la période suivante. La figure 12 ci-dessous illustre la méthode Scrum.



Figure 12: Illustration du cycle de développement de la méthode Scrum

3.1.3. La méthode XP

Alors que la méthode Scrum est assez générale, la méthode XP (*eXtreme Programming* en anglais) est plus adaptée pour le développement logiciel. Outre les principes généraux des méthodes agiles, XP revendique des principes communs du développement qu'elle transforme en pratiques à adopter. Par exemple, la revue de code doit être faite en permanence, par un binôme ; les tests sont systématiquement mis en œuvre ; les besoins évoluent vite, donc les cycles doivent être courts.

Comme pour la méthode Scrum, le client est très impliqué. La méthode XP incite celui-ci à être présent pendant toute la durée du projet. Son rôle est important : grâce aux cycles courts, il peut être régulièrement informé et faire évoluer sa demande ; il lui est également demandé de créer des scénarios pour les fonctionnalités à développer. La méthode se veut aussi humaine. Le rythme de travail doit rester soutenable. Les membres de l'équipe de développement fonctionnent en pair. La

communication au sein de l'équipe et avec le client est l'un des piliers. Le courage est d'ailleurs évoqué comme l'une des cinq valeurs de la méthode car il en faut parfois pour prendre des décisions difficiles. La figure 13 ci-dessous illustre les différentes étapes de la méthode XP .

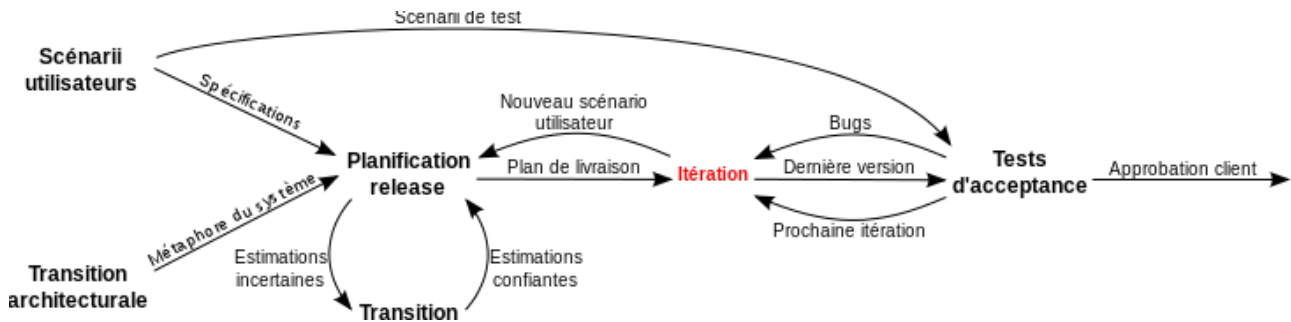


Figure 13: Illustration de la méthode XP
(source : https://fr.wikipedia.org/wiki/Extreme_programming#/media/File:Extreme_programming.svg)

3.2. La mise en place des backlogs

3.2.1. Définition des backlogs et user stories

Le *backlog*, qui peut être traduit par carnet, est un document de conception et de suivi de projet qui contient les fonctionnalités à développer. Il est défini et utilisé dans les méthodes agiles, en particulier *Scrum*. Le *backlog produit* peut être assimilé à un cahier des charges évolutif car il contient l'ensemble des fonctionnalités à développer.

La représentation des fonctionnalités au sein du backlog est régie par des conventions. Une fonctionnalité est exprimée sous forme de *user story* (ou récit d'utilisation en français) qui illustre le besoin à partir d'un exemple simple et précis. En particulier, le récit doit indiquer ce que réalisera la fonctionnalité, qui s'en servira et, de manière optionnelle, la raison de cette fonctionnalité.

Dans le cas de Gamer, un modèle précis de backlog a été défini. Chaque fonctionnalité est représentée par les informations suivantes :

- un identifiant dans le backlog ;
- un type qui indique une catégorie de la fonctionnalité ; les catégories utilisées sont: fonction, contenu/configuration, recherche d'information, bug ;
- un module, qui représente un concept global par exemple, l'authentification d'un système est un module ;
- le groupement, qui est un ensemble de fonctionnalités au sein d'un module ;

- le nom de la fonctionnalité pour la définir dans les communications ;
- la description de la fonctionnalité : c'est la *user story* ;
- la priorité de la fonctionnalité ;
- l'estimation de la difficulté, représentée en nombre de jours ;
- un champ de « démonstration » qui indique la procédure à suivre pour la validation à la fin du sprint ;
- un champ de commentaire ;
- le nom de l'initiateur de la fonctionnalité ;
- l'échéance ;
- le statut (ok ou en cours) ;

3.2.2. Backlog sprint

Le *backlog produit* est à l'origine des *backlogs sprint* qui déterminent les fonctionnalités à réaliser pendant un *sprint* de la méthode. Les fonctionnalités sont choisies en fonction des priorités, échéances, ressources disponibles et l'estimation de la complexité.

3.2.3. Le backlog produit du stage

Nous présentons dans la table 1 un extrait du backlog produit qui contient l'ensemble des modifications développées pendant le stage. Nous avons choisi une représentation concise des fonctionnalités, c'est pourquoi seuls les modules, groupements et descriptions sont affichés. De plus, dans un souhait de clarté, nous avons passé en surbrillance les fonctionnalités directement liées aux problématiques. Ainsi, les détails de la gestion des langues sont en orange, les modifications liées à la gestion des scores en cyan et celles qui concernent l'interface en magenta.

Beaucoup de tâches ne sont pas en rapport avec les problématiques évoquées précédemment, ce sont les tâches qui ne sont pas colorées dans le tableau ci-après. Cependant, il paraît important de les mentionner car elles ont été remplies et, ainsi, illustrent la quantité de travail effectuée. C'est le cas de nombreuses corrections techniques qui portent sur les cas particuliers des fonctionnalités que l'on met en place. Par exemple, dans le tableau du backlog produit, nous rapportons deux modifications à apporter au comportement de la validation d'une réponse par le devin. Le premier concerne la saisie d'une mauvaise réponse, qui doit être acceptée et correctement gérée. L'entrée du backlog intitulée *DevinSpolié* s'y réfère et indique que, pour l'instant, si le joueur entre une mauvaise réponse et la valide, alors toutes les réponses saisies par la suite seront considérées

comme fausses. Un autre cas, moins rationnel, porte sur l'appui sur le bouton de validation répété plusieurs fois qui déclenche l'acceptation de la réponse. D'autres problèmes portent sur la communication entre l'affichage et la base de données. Notamment, il faut prévoir les cas où des cartes ou enregistrements sont indisponibles (car la base de cartes d'une langue vient d'être créée ou parce que le joueur a épuisé la réserve en jouant trop de parties).

MODULE	GROUPEMENT	DESCRIPTION
Débuggage	Test et débogage	Juste une étape par laquelle il faudra passer de test
Son	Mono	Utiliser des enregistrements Mono pour sauver de la BP et de l'espace disque
Traces et TAL	Niveau d'un joueur	En tant que joueur, j'ai une évaluation de mon niveau dans le jeu qui me satisfasse.
Traces et TAL	Estimation de la difficulté d'une carte	En tant que joueur, j'ai une indication de la difficulté de la carte qui m'est proposée
Interface	Lancement Oracle	En tant qu'Oracle, l'enregistrement, le chronomètre et l'affichage de la carte ne sont déclenchés QUE lorsque j'ai autorisé le micro
Interface	Lancement Devin	En tant que Devin, je n'ai pas de contrôles pour la lecture qui est déclenchée automatiquement, avec la minuterie
Compatibilité	Enregistrement firefox	Faire fonctionner l'enregistrement sur firefox
Débuggage	SuperDevin	Sur la page de jeu, lorsque l'utilisateur clique plusieurs fois sur le bouton "valider" (même si le champ réponse est vide ou erroné) la réponse est considérée comme correcte, et le joueur est gratifié de 10 points.
Débuggage	DevinSpolié	Toujours dans la page de jeu (Devin) si l'utilisateur entre une mauvaise réponse et valide. Même s'il modifie sa réponse, et entre celle attendue, ça ne fonctionnera pas...
Débuggage	Bûcher	Pour le druide, lorsqu'il met un enregistrement "au bûcher", l'enregistrement est toujours valide, et n'est pas effacé. De même, on peut tomber plusieurs fois sur le même enregistrement, même s'il a été mis "au bûcher".
Débuggage	Profil	Dans l'onglet profil, si l'utilisateur se trompe lors du changement de son mot de passe ou s'il choisi un champ vide pour la langue, une erreur apparaît 'home/tab/public_html/gow10/views/edit.form.html on line 10.
Généricité	Messages	Tous les messages affichés dans l'application sont stockés dans des fichiers dédiés afin qu'il soit aisé d'ajouter une langue
Généricité	Scoring	Toute la gestion des nombres de points est faite dans un même module ou à travers un unique fichier de constantes, afin de pouvoir modifier la stratégie facilement
Debuggage	Upload et chrono	Attention le chrono doit couper l'enregistrement, pas l'upload. Là il est possible de finir sa description en 30 secondes, de lancer l'upload du

		fichier, que le chrono se termine pendant que la barre de téléchargement s'affiche pour finalement que l'on ait perdu...
Interface	Clarté Oracle	En tant qu'utilisateur, j'ai un indicateur du fait que l'enregistrement est en cours
Debuggage	Erreur lorsque le niveau n'est pas récupéré	Le niveau n'est pas récupéré en javascript. Erreur 404.
Interface	Oracle	Améliorer l'interface pour l'oracle pour que ce soit plus agréable et ludique. (découper l'interface en trois parties distinctes)
Debuggage	Enregistrement non présent sur le serveur	Lorsque l'enregistrement d'une carte n'est pas trouvé sur le serveur, proposer une autre carte avec enregistrement.
Généricité	Multi-lingue	En tant que joueur, je choisis une langue pour mon interface (qui devient la langue de jeu par défaut), mais à tout moment je peux changer la langue de jeu
Gestion des scores	Scoring par langue	En tant que joueur, je peux afficher mes points de niveau langue par langue
Interface	Notifications	Créer un système de notifications
Débuggage	BD vide	En tant qu'Oracle, si j'essaie de jouer sur une appli sans carte, c'est tout désordonné...
Gestion des langues	Profil	Dans le profil, avoir la possibilité de choisir une langue maternelle et une ou des langue(s) d'apprentissage et puis, le joueur peut supprimer des langues ou en ajouter d'autres .
Gestion des langues	Rôle	Dans chaque partie, on peut choisir la langue à jouer (par défaut la 1ère langue apprise choisie lors de l'inscription). Le joueur peut jouer en tant qu'Oracle et Druide uniquement dans les langues choisies dans son profil, néanmoins, il pourra en tant que Devin jouer dans n'importe quelle langue.
Interface	Langue de l'interface	Avoir une liste déroulante des langues respectant la norme iso 639-1 dans la bar de navigation de la plateforme. De plus, pour les langues les plus fréquemment apprises il y aura un petit drapeau qui facilitera la lecture.
Gestion des scores	Table dans la BDD	Cette dernière retrace toutes les actions réalisées par les joueurs. A partir de ces actions, nous pourrons alors définir un système de notation plus précis.
Système de calcul de score	Calcul du score	En tant que joueur mon score est mise à jour chaque fois que je joue comme Druide, Devin. Si un autre Druide ou Devin joue avec un de mes enregistrements, je gagne des points entant qu'

		Oracle.
Gestion des scores	Calcul de score	Le calcul de score est réalisé à partir d'un indicateur de niveau de difficulté de la carte et d'un indicateur de niveau de langue de jeu de joueur.
Gestion des scores	Tables des scores	En tant que joueur, je peux aller voir mon classement et les 3 meilleurs pour chaque langue et globalement

Table 1: Extrait du backlog produit

3.3. Objectifs du développement et outils

3.3.1. Objectifs du développement

Pour résumer, outre les corrections d'erreurs et autres bogues, le *backlog produit* présente des fonctionnalités à développer suivant les 3 problématiques de gestion de scores, gestion des langues et amélioration de l'interface. La problématique liée à la gestion des langues requiert principalement le choix de pouvoir jouer dans plusieurs langues et de dissocier le rôles des langues. Ainsi, il devrait y avoir une langue pour l'interface et une ou plusieurs langues pour le jeu. Le joueur doit également pouvoir changer les langues renseignées quand il le souhaite. La structure mise en place doit faciliter l'ajout d'une nouvelle langue pour l'affichage des messages de l'interface.

La problématique liée à la gestion des scores requiert la prise en compte du niveau de difficulté de la carte et du niveau de maîtrise de la langue par le joueur. En outre, un système d'évaluation du score basé sur l'expérience acquise au fil du temps et estimée par les actions réalisées dans le jeu est évoqué. Le traitement du score requiert également plusieurs éléments d'affichage pour l'utilisateur, qui seront abordés avec le traitement de l'interface.

Enfin, l'amélioration de l'interface porte essentiellement sur l'information de l'utilisateur. Le système de notifications et les indicateurs d'enregistrement ou de déclenchement de partie en sont des exemples. Les fonctionnalités des deux autres problématiques évoquent également le besoin d'interagir avec l'utilisateur. Ainsi, le joueur devrait pouvoir changer les langues de jeu, afficher son score, la difficulté d'une carte ou le classement global des joueurs.

3.3.2. Outils pour la réalisation

A l'image du travail réalisé par nos prédécesseurs, les travaux de développement présentés dans ce rapport ont été effectués dans les langages du web que nous avons étudiés au cours de notre formation : PHP, HTML, CSS et Javascript. La bibliothèque de code WebRTC a été employée pour l'enregistrement et l'écoute de fichiers sonores directement depuis une page web. L'outil MySQL a servi à manipuler la base de données.

Partie 2

Solutions proposées pour Game of Words

Chapitre 4 – Évaluation des niveaux et gestion des langues

Cette partie aborde les solutions apportées aux problèmes de gestion des langues et des scores dans Game of Words. Dans un premier temps, nous introduisons les concepts du modèle de gestion de langues, avec notamment les différents status et rôles des langues dans le jeu et la prise en compte de niveau de jeu. Par la suite, nous évoquons la réalisation de ce modèle dans le jeu. La suite de la partie est dédiée au score et se décompose de la même manière. Le modèle de gestion de score est d'abord introduit, suivi de sa réalisation dans le jeu.

4.1. Modèle de la gestion des langues

4.1.1. L'usage des langues dans la version initiale

Comme nous l'avons présenté dans la partie 2, la version de Game of Words au début du stage limite le jeu au français et à l'anglais. Par ailleurs, le rôle de la langue est unique : la langue de jeu est la même que celle de l'interface.

Les fonctionnalités du *backlog produit* indiquent le besoin de choisir une langue pour l'interface et une autre langue pour le jeu. De plus, l'utilisateur devrait pouvoir renseigner plusieurs langues pour jouer et en changer comme il le souhaite. Chaque langue devrait être accompagnée d'un niveau de jeu. Dans un effort de généricité, les langues et niveaux devraient être définis suivant des catégories.

4.1.2. Le modèle de gestion de langue

Les rôles des langues – Afin de satisfaire les requêtes évoquées, nous introduisons un modèle qui distingue 3 rôles pour les langues :

- *les langues parlées* : ce sont les langues que l'utilisateur connaît suffisamment pour pouvoir jouer des parties ; ces langues sont choisies parmi un ensemble normalisé pour le jeu ;
- *la langue de jeu* : c'est la langue de jeu dans la partie courante ; elle a été sélectionnée parmi les langues parlées et peut être changée à n'importe quel moment ;
- *la langue de l'interface* : il s'agit d'une langue choisie parmi celles proposées pour afficher les messages de l'interface ; elle est indépendante des langues parlées et de la langue de jeu.

La figure 14 ci-après représente la gestion des langues proposées.

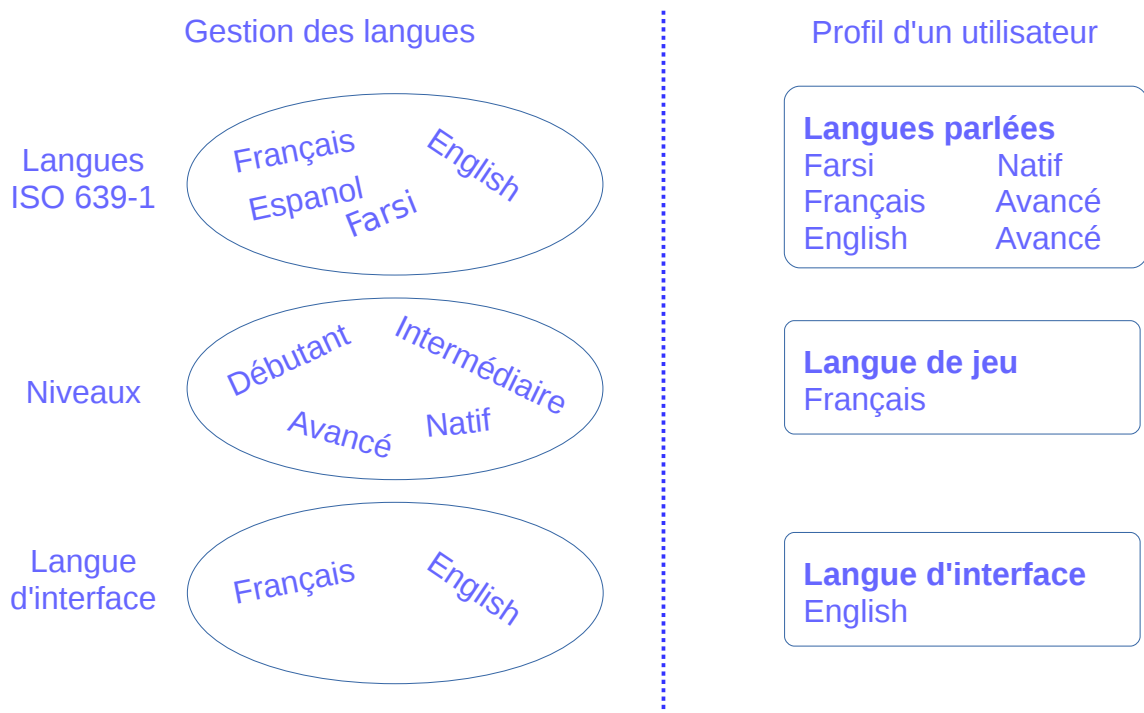


Figure 14: Illustration de la gestion des langues

Les langues proposées pour les parties du jeu proviennent de la liste des langues du standard ISO 639-1. Celui-ci est une partie de l'ensemble des noms de langues représentées par la norme ISO 639. Le standard ISO 639-1 répertorie 136 langues parmi les plus parlées au monde. Il a été publié pour la première fois en 1967 et la dernière révision est celle de 2002. Le standard associe un code de deux lettres à chacune des langues enregistrées. Au-delà d'une simple combinaison arbitraire, ce code essaie d'être représentatif du nom de la langue. Ainsi le code du français est *fr*, celui de l'anglais est *en* (pour *english*) et celui du persan est *fa* (pour *farsi* en persan). Cependant, le faible de nombres de combinaisons autorisé a mené à des réformes du standard pour intégrer de plus en plus de langues. D'autres standards ont été proposés depuis comme l'ISO 639-3 qui répertorie 7700 langues et leur associe un code de 3 lettres.

La langue de l'interface est choisie parmi un ensemble de langues réduit : il s'agit de celles dont la traduction des messages de l'interface est disponible. Pour l'instant, seuls le français et l'anglais sont disponibles.

La définition des niveaux de langue – Les langues parlées sont associées à un niveau de maîtrise. Outre la demande faite pour l'évaluation des scores, la définition du niveau est utile à la vocation d'apprentissage du jeu ; elle peut motiver l'apprenant et cibler le niveau du joueur par les enseignants.

Le cadre Européen Commun de Référence (CECR) est un document publié par Conseil de

l'Europe en 2001, qui tente d'identifier et de définir les facettes théoriques de l'apprentissage d'une langue en Europe. Le cadre est un guide de principe théorique et de références qui propose une base commune dans le domaine de la didactique de langues. Ce manuel met l'accent sur la nature de l'utilisation de la langue et sur celle de l'utilisateur ainsi que sur les répercussions sur l'enseignement et l'apprentissage.

Un des buts du Cadre de référence est d'aider les partenaires à décrire les niveaux de compétence exigés par les normes et les tests et les examens existants afin de faciliter la comparaison entre les différents systèmes de qualifications. C'est à cet effet que le CECR a conçu le Schéma descriptif et les Niveaux communs de référence (C.E. 2001). Dans les grilles de niveaux, il définit des seuils de maîtrise d'une langue étrangère en fonction de savoir-faire dans différents domaines de compétence (comprendre, parler, écrire).

Pour cela, le cadre a défini six niveaux : *introductif* ou de *découverte* ; *intermédiaire* ou de *survie*, *seuil* (ou *avancé*), *avancé* ou indépendant ; autonome ou de *compétence opérationnelle effective* ; et *maîtrise*. Ces six niveaux correspondent à des interprétations supérieures ou inférieures de la division classique en niveau élémentaire, indépendant et expérimenté. La figure 15 ci-dessous présente une division plus générale de ces niveaux.

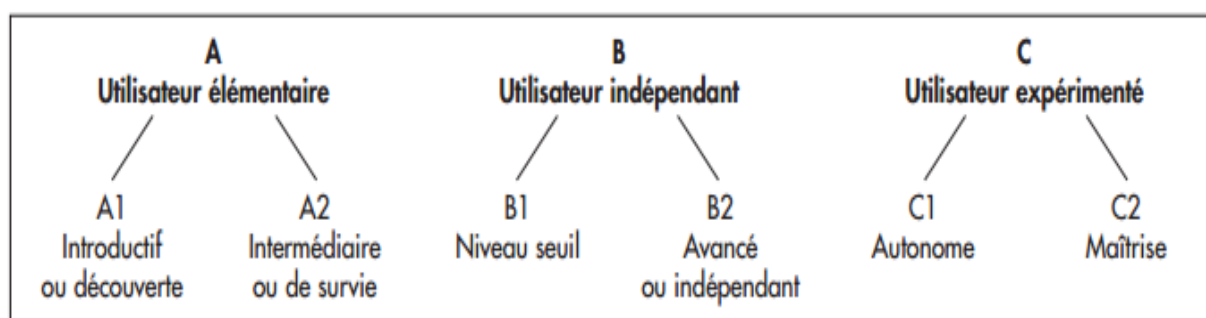


Figure 15: Schéma des niveaux de langue du CECR

Le CECR a aussi introduit une grille détaillée des descripteurs des niveaux pour l'auto-évaluation. Cette échelle présente trois compétences pour chacun des six niveaux de base. La première, qui concerne la compréhension, est divisée en deux parties : écouter et lire. La seconde est l'expression orale et se divise en deux : prendre part à une conversation et s'exprimer oralement en continu ; et la dernière compétence est l'expression écrite. La figure 16 présente la compétence d'expression orale (parler). De gauche à droite, les colonnes représentent les capacités des niveaux A1, A2 et B1.

P A R L E R	Prendre part à une conversation	Je peux communiquer, de façon simple, à condition que l'interlocuteur soit disposé à répéter ou à reformuler ses phrases plus lentement et à m'aider à formuler ce que j'essaie de dire. Je peux poser des questions simples sur des sujets familiers ou sur ce dont j'ai immédiatement besoin, ainsi que répondre à de telles questions.	Je peux communiquer lors de tâches simples et habituelles ne demandant qu'un échange d'informations simple et direct sur des sujets et des activités familiers. Je peux avoir des échanges très brefs même si, en règle générale, je ne comprends pas assez pour poursuivre une conversation.	Je peux faire face à la majorité des situations que l'on peut rencontrer au cours d'un voyage dans une région où la langue est parlée. Je peux prendre part sans préparation à une conversation sur des sujets familiers ou d'intérêt personnel ou qui concernent la vie quotidienne (par exemple famille, loisirs, travail, voyage et actualité).
	S'exprimer oralement en continu	Je peux utiliser des expressions et des phrases simples pour décrire mon lieu d'habitation et les gens que je connais.	Je peux utiliser une série de phrases ou d'expressions pour décrire en termes simples ma famille et d'autres gens, mes conditions de vie, ma formation et mon activité professionnelle actuelle ou récente.	Je peux m'exprimer de manière simple afin de raconter des expériences et des événements, mes rêves, mes espoirs ou mes buts. Je peux brièvement donner les raisons et explications de mes opinions ou projets. Je peux raconter une histoire ou l'intrigue d'un livre ou d'un film et exprimer mes réactions.

Figure 16: Extrait de la grille de description de la compétence d'écoute pour les niveaux A1 à B1 du CECR

Au vu de la précision des descriptions des compétences pour décrire les niveaux, nous remarquons qu'il est difficile de se positionner dans ces grilles pour chaque compétence surtout celle de l'expression orale. C'est pourquoi nous avons opté pour une description plus intuitive pour le jeu. Ainsi, nous avons fait le choix d'exprimer le niveau suivant quatre adjectifs : débutant, intermédiaire, avancé et natif. Nous pensons qu'il sera facile pour une personne de se situer au sein de cette échelle. En effet, les extrêmes sont facilement identifiables : une langue native ou une langue à peine parlée seront facilement reconnaissable respectivement par les adjectifs natif et débutant. Une personne qui n'est ni native ni débutante pourra estimer si son niveau est plus proche du débutant ou du natif et ainsi décider entre la catégorie intermédiaire ou avancée.

4.2. Réalisation de la gestion de langues

4.2.1. Le modèle dans la base de données

Comme nous l'avons vu dans la section dédiée à la description technique du jeu, la plupart des informations sont sauvegardées dans la base de données. C'est pourquoi, pour la gestion des langues, nous avons ajouté des tables à la base. C'est ainsi que les langues parlées par les joueurs sont entrées dans une table de la base de données. Une entrée correspond à un profil de joueur et contient son identifiant, les langues parlées et le niveau associé. Dans la mesure où le nombre de langues parlées est inconnu (bien que limité à quelques unes dans la plupart des cas), nous avons choisi de représenter l'ensemble des langues dans une seule colonne. Les langues sont entrées dans

la case correspondante et séparées au moyen d'un délimiteur (ici, il s'agit du caractère « ; »). Ce délimiteur, unique, permet aisément de séparer la chaîne de caractères qui contient les langues. Cette façon de faire a l'avantage de rendre quasi-illimité le nombre de langues parlées par l'utilisateur et réduit le nombre de colonnes en cas de visualisation de la table (en pratique, nous avons limité l'interface du profil pour l'ajout des langues à 9, donc une entrée ne contiendra jamais plus de 9 langues). La figure 17 est un extrait de la table des langues parlées dans la base de données. En rouge, sont représentés les champs des langues parlées et des niveaux (que nous décrivons ci-après). Également, l'identifiant du joueur est indiqué dans la première colonne.

id	userid	spoken_lang	niveau
te 2	69	Français;Anglais;	Natif;Débutant;
te 3	70	Français;Chinois;	Intermédiaire;Débutant;
te 4	71	Français;	Débutant;

Figure 17: Extrait de la table des langues parlées dans la base de données

De la même manière, les niveaux de langues sont renseignés dans une colonne adjacente. Chaque niveau est séparé par le même délimiteur. En outre, afin d'associer le niveau à la bonne langue, l'ordre des niveaux est identique à l'ordre des langues. Ainsi, le premier niveau est associé à la première langue, etc.

A partir des différentes langues parlées par un utilisateur, une langue particulière est employée pour le jeu. Le choix de cette langue est sauvegardée avec les informations du profil du joueur, dans une table dédiée. Cette table existait déjà dans la version initiale du jeu et a été complétée par plusieurs colonnes. La langue de jeu est renseignée dans la colonne intitulée *userlang_game*. La langue de l'interface est aussi mentionnée dans cette table, à la colonne *userlang*. La figure 18 illustre les colonnes mentionnées, dans une capture d'écran de la table des profils des utilisateurs.

userid	username	useremail	userpass	userlang	valkey	userlang_game	photo	userlvl
97	jojo	jojo@free.fr	7510d498f23f5815d3376ea7bad64e29	fr		de	profil/jojo/druide.jpg	medium
98	mma	mm@m.l	b3cd915d758008bd19d0f2428fbb354a	en		de	profil/unknow.jpg	hard
99	mp	mp@mp.mp	1f2dfa567dcf95833eddf7aec167fec7	fr		fr	profil/unknow.jpg	medium

Figure 18: Extrait de la table des profils qui met en évidence les langues de l'interface et du jeu

4.2.2. L'interaction de l'utilisateur

Après avoir présenté les modifications de la base de données pour y intégrer les nouvelles informations sur les langues du joueur, nous abordons maintenant l'affichage. Celui-ci permet au joueur de visualiser l'état actuel et d'interagir pour modifier un élément. Les modifications de langues sont réalisées *via* différentes interfaces : le profil et les modes de jeu. L'ensemble de ces interfaces existait dans la version initiale, nous les avons complétées. La figure 19 illustre la nouvelle interface du profil.

Les langues sont renseignées dans une table. Chaque entrée contient deux champs pour la langue et le niveau. Les deux champs sont des listes qui permettent de choisir parmi les langues du standard ISO 639-1 et les quatre niveaux présentés. Un bouton est associé à chaque entrée afin de sélectionner la langue de jeu. La langue de l'interface est renseignée par le biais d'un menu déroulant, plus bas dans l'interface. La validation déclenche les traitements du contrôleur qui récupère les noms des langues et niveaux et les met bout à bout dans deux chaînes de caractères. La table de la base de données est ensuite mise à jour.

The screenshot displays a user profile configuration page. At the top, there is a 'confirm your password' section with a masked password field. Below this, the 'Langues parlées' section is highlighted with a red border. It features a table with three columns: 'Langue', 'Niveau', and 'Langue de jeu'. The table contains three rows: 'Français' (Intermediate level) with a selected radio button, 'Anglais' (Beginner level) with an unselected radio button, and 'Allemand' (Beginner level) with an unselected radio button. Below the table are buttons for 'Add a language that you are studying' and 'Retirer langue'. Underneath, the 'Choisissez un niveau de jeu' section shows a dropdown menu set to 'Moyen'. The 'Interface language' section has a dropdown menu set to 'Anglais'. At the bottom, there are two large buttons: 'Submit' and 'Cancel'.

Langue	Niveau	Langue de jeu
Français	Intermédiaire	<input checked="" type="radio"/>
Anglais	Débutant	<input type="radio"/>
Allemand	Débutant	<input type="radio"/>

Figure 19: Nouvelle page de profil (partie basse)

Les trois modes de jeu permettent la modification de la langue de la partie. Cette modification est semblable à celle dans le profil et sera appliquée de manière permanente, jusqu'au prochain changement. La présentation au joueur se fait au moyen d'un menu déroulant, située sous la barre de navigation. Celle-ci propose des langues, basées sur les langues parlées par le joueur pour les modes de l'oracle et du druide et sur l'ensemble des langues du standard pour le devin. Les figures 21 et 20 illustrent le menu déroulant.

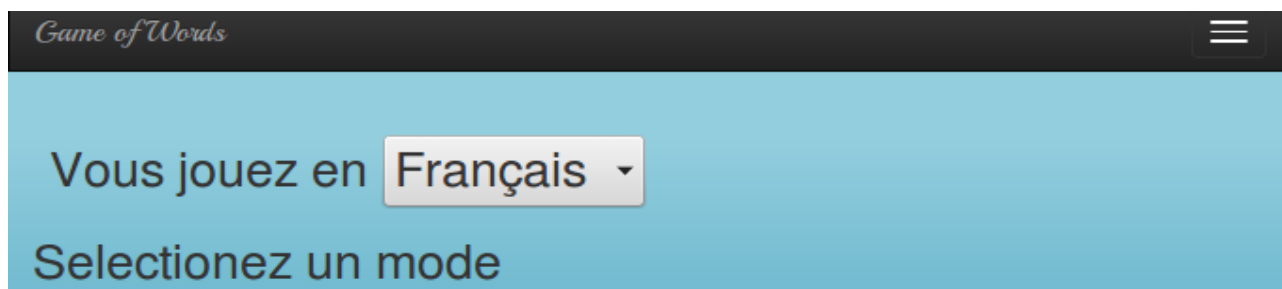


Figure 20: Capture d'écran du menu déroulant de la langue de jeu pour l'oracle

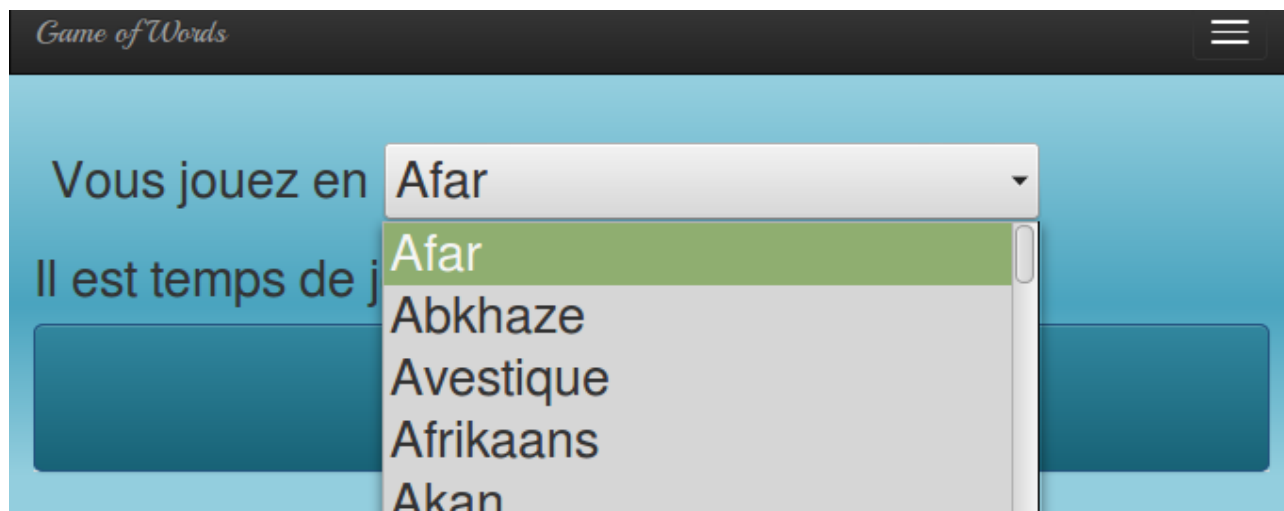


Figure 21: Capture d'écran du menu déroulant de la langue de jeu pour le devin

4.3. Modèle d'évaluation des points

À la manière de la description du modèle de langues dans les deux sections précédentes, nous présentons le modèle de gestion du score dans les deux sections suivantes. Cette section introduit les concepts tandis que la section 4.4 présentera les réalisations. Plus précisément, la stratégie de rétribution des actions est d'abord décrite avant d'aborder le calcul des points et l'évaluation globale.

4.3.1. Rétribution des actions effectuées

Le modèle que nous décrivons rétribue les joueurs selon leurs actions dans les trois rôles. Nous décrivons les cas pour chaque rôle. Le cas le plus évident est celui du devin. La réussite du devin lui donne droit à un total de X points, dont le calcul est décrit dans la section suivante. A l'inverse, l'échec du devin ne lui retire pas de point.

Le second cas concerne le druide. La création d'une carte et l'arbitrage d'un enregistrement par le druide sont toujours récompensés par des points. Cela est justifié par la volonté d'enrichir la base de données.

Le troisième cas concerne l'oracle. Celui-ci est plus complexe que les deux précédents, car la rétribution est dépendante d'autres actions. La seule action de l'oracle est l'enregistrement, mais suivant les actions du druide et du devin, son score évoluera. La première étape de l'enregistrement est l'arbitrage du druide. Si ce dernier le valide, alors l'oracle gagnera des points ; à l'inverse, une invalidation par le druide retirera des points à l'oracle. La même logique est appliquée à la réussite du devin : le succès du devin rapporte des points à l'oracle tandis que son échec retire des points à l'oracle.

Cette stratégie se justifie par l'importance de l'enregistrement pour le devin. Une définition de mauvaise qualité (par son contenu ou son intelligibilité) pénalise le devin. Ainsi, la répercussion du résultat du devin sur le score de l'oracle incite les joueurs à réaliser des enregistrements de bonne qualité. La figure 22 ci-dessous décrit l'ensemble des cas présentés pour la rétribution des points.



Figure 22: Illustration des différents cas d'attribution des points

La dépendance du score de l'oracle aux actions des autres joueurs est une idée intéressante pour le renforcement de la qualité des productions. Cette stratégie devrait également être appliquée aux actions du druide. En effet, si celui-ci crée une carte contenant des fautes ou des mots irréels, le reste de la partie est faussé. Également, le druide peut commettre des erreurs pendant l'arbitrage d'un enregistrement. Pour ces raisons, son score devrait dépendre du résultat du devin.

4.3.2. Estimation des points

L'estimation des points suit le même calcul pour les trois rôles. Elle est faite en fonction du niveau de difficulté de la partie choisie par le joueur, du niveau de complexité de la carte et de la maîtrise de la langue jouée.

Le niveau de difficulté de la partie donne droit à un total de points, qui servira à la base du calcul. Trois niveaux sont acceptés – facile, moyen, difficile – qui donnent respectivement droit à 10, 20 et 40 points. La maîtrise de la langue est exprimée par un coefficient. Les quatre niveaux de maîtrise – débutant, intermédiaire, avancé et natif – sont associés aux coefficients 1,5 ; 1,2 ; 1 ; et 0,5. La complexité de la carte est également associée à un coefficient. Les niveaux facile, intermédiaire et difficile correspondent aux valeurs 0,5 ; 1 ; et 1,2. Ces valeurs sont résumées dans la table 2. Outre les coefficients, le niveau de difficulté du jeu conditionne le nombre de mots tabous et la durée du chronomètre allouée. Nous avons associé des valeurs à ces niveaux, qui sont présentées dans la figure 23.

Difficulté du jeu		Maîtrise de la langue		Difficulté de la carte	
Facile	10	Débutant	1,5	Facile	0,5
Moyen	20	Intermédiaire	1,2	Intermédiaire	1,0
Difficile	40	Avancé	1,0	Difficile	1,2
		Natif	0,5		

Table 2: Coefficients et nombres de points suivant le niveau de jeu, la maîtrise de la langue et la difficulté de la carte

Le calcul des points consiste à multiplier la base de points par les coefficients de maîtrise de la langue et de difficulté de la carte. Nous illustrons le calcul par un exemple. Considérons un devin qui joue une partie de difficulté moyenne en français. La difficulté de la carte est facile et sa maîtrise de la langue de jeu est intermédiaire. La base de points X est 20, due au niveau de difficulté moyen. Ce nombre est multiplié par le coefficient de maîtrise de la langue 1,2 et par la difficulté de la carte 0,5. Le score pour le devin est calculé par l'opération : $20 \times 1,2 \times 0,5 = 12$ points.

4.3.3. Évaluation globale

L'évaluation des résultats des joueurs est effectuée suivant les rôles et les langues. Ainsi, pour une langue donnée, les joueurs ont un score par rôle et un score global qui est la somme des scores de rôles. Par ailleurs, les joueurs ont un score pondéré, qui combine les scores globaux par langue en leur affectant un coefficient. Les valeurs sont 1 ; 1,25 ; et 1,5 (résumées dans la table 3). L'application des coefficients concerne les trois meilleures langues du joueur, ordonnées suivant de la meilleure à la plus faible. Ainsi, la langue au score le plus faible est avantagée et, inversement, l'effet de la « meilleure » langue est réduit. Cette méthode vise à encourager le jeu dans les langues qu'ils maîtrisent le moins.

Classement de la langue	
1er rang	1
2 nd rang	1,25
3 ^e rang	1,5

Table 3: Table des coefficients suivant le rang du score

4.4. Mise en place du système de gestion des scores

4.4.1. Les tables de scores dans la base de données

Les tables décrites dans la section précédente ont été ajoutées à la base de données. Les captures d'écran de la figure 23 illustrent leur composition.

userlvl	time	points	pointsSanction
easy	90	10	0
hard	45	40	0
medium	60	20	0

niveau_langue	coeff
Débutant	1.5
Intermédiaire	1.2
Avancé	1
Natif	0.5

Id	coeff
1	1
2	1.25
3	1.5

Figure 23: Illustration des tables de points et coefficients dans la base de données

De gauche à droite, on retrouve la table des valeurs de base pour le calcul des points, la table des coefficients de niveau et de langue et celle des coefficients de rang du score. La table qui sauvegarde les scores existait dans la version initiale et a gardé son utilité dans cette version.

Le calcul des points est réalisé grâce à une combinaison de requêtes SQL à la base de données. Il requiert essentiellement deux phases :

- identifier le profil et le rôle du joueur dont le score doit être modifié;

- effectuer le calcul et la mise à jour.

Les requêtes nécessaires pour la réalisation de ces opérations consistent à récupérer une information (score, identifiant, coefficient, etc) et mettre à jour une ou plusieurs valeurs dans les tables. Leur réalisation dans le langage SQL est respectivement *SELECT* et *UPDATE*.

Pour le calcul du score, les informations nécessaires sont le score courant du joueur dans le rôle, la base de points et les coefficients à appliquer. Le résultat du calcul est ajouté au score et mis à jour dans la table des scores. Le score global est également mis à jour. Cette étape est directement appliquée pour le druide et le devin, sans phase d'identification puisqu'elle est consécutive à leurs actions.

L'identification du joueur et du rôle est nécessaire lorsqu'il s'agit d'une modification *a posteriori*. En pratique, elle s'applique seulement à l'oracle, après l'arbitrage d'un enregistrement ou une partie d'un devin. L'identifiant de l'enregistrement permet de déterminer l'identifiant du joueur qui l'a créé. Par suite, on peut avoir accès au profil et au score du jour et donc avoir accès aux éléments nécessaires pour le calcul du score. Les figures 24 et 25 illustrent la phase de calcul du score et la séquence d'identification et de calcul pour l'oracle.

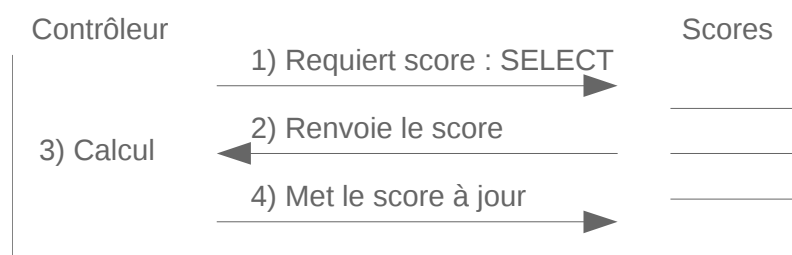


Figure 24: Illustration des phases dans le calcul du score

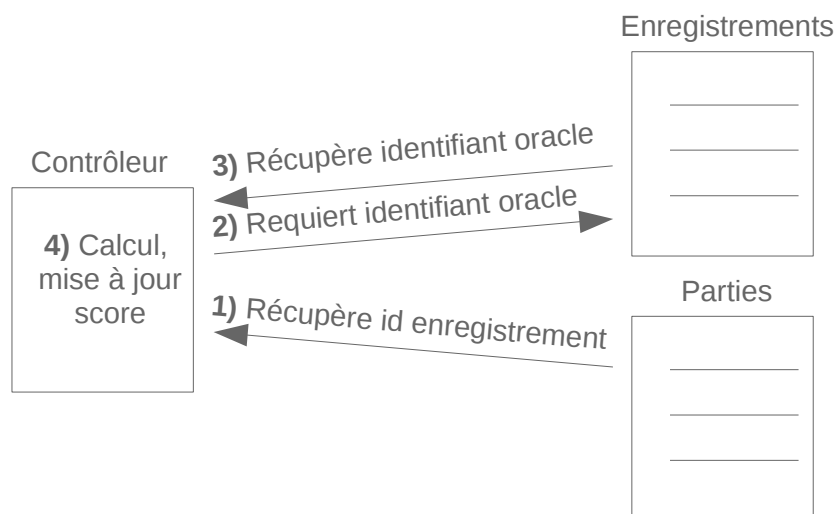


Figure 25: Illustration du cas du calcul de score de l'oracle

4.4.2. Amélioration de la visualisation du score

Suivant les requêtes établies dans le *backlog produit*, nous avons modifié la page d'affichage des scores. Plusieurs tables sont proposées au joueur, qui contiennent des classements suivant les langues et un classement global. La figure 26 illustre ces tableaux pour un joueur parlant allemand, français et ouzbek.

Classement	Joueur	Score Global	Nombre de langues
1	ee	1581.25	3
2	pi	1145.625	3
3	jp	1117.5	1
4	aa	537	3
5	mma	438.75	2

Scores Allemand					
	Joueur	Score_Oracle	Score_Druide	Score_Devin	Score Global
1	aa	1050	0	0	1050
2	jp	395	60	390	745
3	pi	0	10	230	240
4	mma	0	40	5	185
5	pop	0	0	100	100

Scores Français					
	Joueur	Score_Oracle	Score_Druide	Score_Devin	Score Global
1	mma	50	170	45	415
2	ee	30	360	0	390
3	nr	248	70	65	383
4	aa	0	40	288	328
5	pi	0	0	245	255

Scores Ouzbek					
	Joueur	Score_Oracle	Score_Druide	Score_Devin	Score Global
1	bobo	30	0	0	30
2	aa	0	20	0	20
3	ww	0	0	15	15

Figure 26: Captures d'écran de la page des scores : en haut le classement global puis les classements par langue

4.4.3. Limites et perspectives

Pour le moment, le calcul du score global d'une langue consiste en la somme des scores des rôles pour cette langue. Les perspectives concernent l'application du coefficient du rang de langue. Également, le coefficient de difficulté de la carte n'est pas appliqué pour le moment. Enfin, le calcul du score pour les actions *a posteriori* comme l'usage d'un enregistrement d'un oracle dans une partie présente une faiblesse. En effet, il tient compte des niveaux de difficulté du jeu et de maîtrise de la langue au moment du calcul et non au moment de la création de l'enregistrement. Si entre ces deux moments, l'oracle change de niveau de maîtrise de la langue, le nouveau coefficient sera appliqué et le nombre de points évalué sera incorrect. Cela peut représenter un moyen de triche pour un joueur qui créera les enregistrements au niveau facile et passera ensuite au niveau difficile pour augmenter le nombre de points reçus.

Chapitre 5 – Refonte de l'interface : une interface orientée vers l'utilisateur

Dans cette dernière partie, nous abordons le travail effectué sur l'interface. Celui-ci porte sur l'amélioration de l'information donnée à l'utilisateur. En particulier, les besoins exprimés dans le *backlog produit* expriment l'intérêt pour l'affichage d'informations complémentaires, notamment dans les tours de jeu du devin et de l'oracle. Un système de notifications est également souhaité. La section démarre par la réflexion sur un tel système et la réalisation qui a suivi.

5.1. Le système de notifications

Nous présentons dans cette section les travaux effectués sur les notifications. Après une brève description d'un tel système et des avantages pour le joueur, nous décrivons notre proposition pour l'intégration à Game of Words. En particulier, nous présentons le concept d'une notification, les objets qui composent le système et les actions à l'origine d'une notification. La réalisation illustre l'organisation du code pour ces fonctionnalités.

5.1.1. Les avantages d'un système de notifications

Présentation générale des notifications – Dans le contexte d'une interface utilisateur, une *notification* est un marqueur qui apparaît afin d'informer l'utilisateur. Celui-ci contient un message principal et peut contenir d'autres éléments utiles tels qu'un titre, une date ou un émetteur. La notification est générée par une action particulière, appelée aussi *événement*. De plus, les systèmes complets proposent généralement un *tableau* récapitulatif des dernières notifications reçues, dans un ordre chronologique inversé. L'usage des notifications est devenu très populaire dans les sites à l'image du réseau social Facebook.

L'intérêt d'un tel système est d'abord la diffusion d'une information de manière élégante et discrète. Plusieurs stratégies d'affichage sont adoptées suivant les objectifs visés, qui s'intègrent toutes à l'interface du site. La diffusion immédiate de l'information s'effectue au moyen d'un petit bandeau ou d'une fenêtre qui apparaît subitement et momentanément et qui ne contient qu'un résumé du message. Par la suite, la notification est « stockée » dans un tableau resté généralement invisible et accessible par un simple bouton de l'interface. L'utilisateur pourra accéder à l'information en « déroulant » le tableau et en sélectionnant la notification. En outre, un rappel indique aux utilisateurs la présence (et parfois le nombre) de notifications non lues. Celui-ci se manifeste par un symbole à côté du bouton d'accès au tableau récapitulatif. Grâce à ces différents moyens d'affichage, l'utilisateur peut être informé en temps réel d'une information particulière. Il

peut également décider d'approfondir la lecture d'une notification ultérieurement et bénéficie d'un rappel pour éviter d'oublier.

Application à Game of Words – Ce système présente plusieurs intérêts qui pourraient être exploités dans le jeu Game of Words. Nous dressons une liste des différents usages et fonctionnalités qui pourraient justifier sa réalisation :

- informer l'utilisateur de l'action d'un autre joueur sur l'un des enregistrements qu'il a produits ; en utilisant le tableau récapitulatif, l'utilisateur pourrait connaître cette information ;
- informer le joueur des actions effectuées pendant son absence *via* le tableau récapitulatif ;
- confirmer une information importante comme la mise à jour du score ; l'apparition d'une petite fenêtre pour indiquer le nouveau score du joueur serait très adaptée.

Nous définissons ces fonctionnalités comme les objectifs à réaliser pour la mise en place du système de notifications.

5.1.2. Notre proposition de système de notifications

Description générale – Nous proposons un système de notifications pour la réalisation des fonctionnalités précédentes. Celui-ci se compose des éléments suivants :

- une table dans la base de données qui sauvegarde toutes les notifications créées ; une notification est représentée par un modèle décrit dans la suite ;
- plusieurs affichages différents, adaptés aux différentes actions et décrits dans la suite ;
- un contrôleur, en charge de la mise à jour de la table et de l'affichage.

La figure 27 illustre la composition du système ainsi que les différentes interactions possibles entre les éléments. Les détails sont abordés dans les paragraphes suivants.

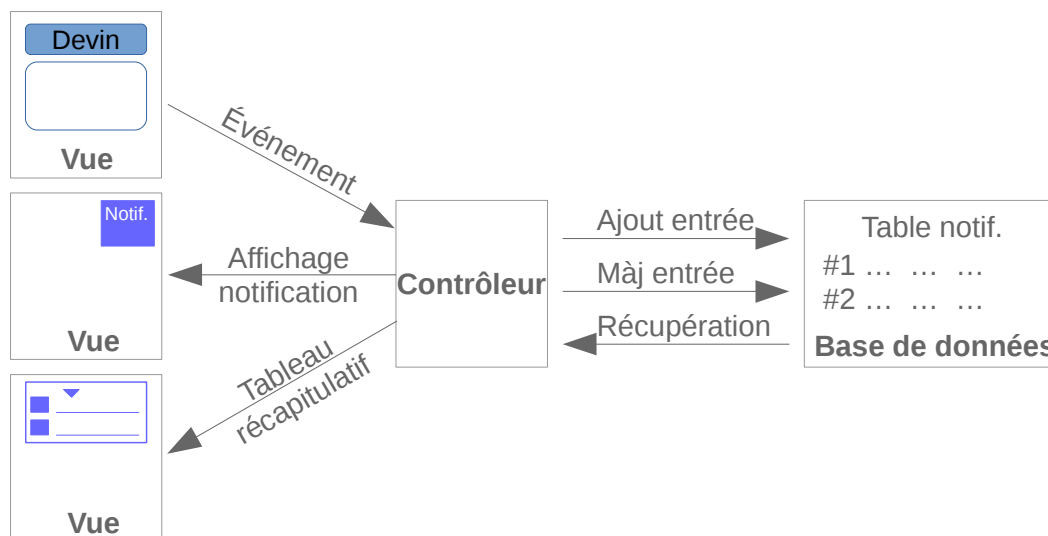


Figure 27: Composition du système de notifications

Description des éléments – Nous décrivons les principaux éléments qui composent ce modèle, en commençant par le modèle d'une *notification* et la table associée dans la base de données. Nous considérons une notification comme une information provenant d'un émetteur et délivrée à un destinataire avec un message particulier. Ainsi, nous la représentons par l'ensemble des éléments suivants :

- l'identifiant du joueur à qui elle est destinée ;
- l'identifiant de l'émetteur ;
- le message qu'elle contient ;
- la date ;
- le mode de jeu qui l'a créée ;
- l'indication de la lecture par le joueur ;
- un identifiant, pour la gestion interne des notifications ;

Les informations importantes sont le destinataire, le message et la date. L'émetteur concerne les parties où un joueur écoute l'enregistrement d'un autre joueur. Dans ce cas, le joueur ayant produit l'enregistrement recevra une notification dont l'émetteur sera le joueur qui l'a écouté. De plus, le mode de jeu précise l'action effectuée. Si l'on reprend l'exemple de l'enregistrement, il peut s'agir d'un druide qui arbitre ou d'un devin qui cherche le mot. L'indication de lecture indique si le joueur a pris connaissance de la notification. On retrouve ces différents champs dans la table de la base de données associée, illustrée par une capture d'écran dans la figure 28 :

userid	message	emetteur	id	state	game	time
128	Diviner : Vous jouez en Français	0	413	1	./profil/diviner	2015-07-14 18:01:44
2	test : +10 points	128	414	0		2015-07-14 18:01:54

Figure 28: Illustration de la table de notifications dans la base de données

L'affichage des notifications est maintenant décrit. Comme indiqué dans la section sur les avantages d'un tel système, nous employons deux méthodes pour l'affichage d'une notification. Le premier consiste à afficher une petite fenêtre temporaire pour y afficher les informations. Il s'agit de la vue du milieu sur le schéma de la figure 27. Ce moyen est réservé à certaines informations immédiates et jugées importantes, générées par des événements. Le second moyen consiste en un tableau récapitulatif, qui a déjà été évoqué de manière générale. Celui-ci contient l'ensemble des notifications du joueur et reste invisible au joueur la plupart du temps. Il est accessible par un bouton dans la barre de navigation. La vue du bas de la figure 27 l'illustre.

Enfin, l'affichage peut aussi déclencher des événements, en particulier à la fin d'un tour de jeu. Ces événements sont transmis au contrôleur qui les traite et met à jour la vue en conséquence.

Les événements sont décrits plus en détails dans la suite.

Le troisième élément qui compose le système est le *contrôleur*. Son rôle est de maintenir la cohérence entre l'affichage des notifications et leurs représentations réelles dans la base de données. Ainsi, il traite les demandes de création et d'effacement soumises par l'utilisateur *via* l'affichage en les appliquant à la base de données. Il possède une version à jour des notifications existantes en récupérant et filtrant les entrées enregistrées dans la table de la base de données, et peut ainsi maintenir l'affichage du tableau récapitulatif à jour.

Événements déclencheurs – Dans la liste suivante, nous indiquons les actions considérées qui sont à l'origine d'une notification (au sens du modèle, c'est-à-dire de la création d'une entrée dans la table de la base de données) :

- réalisation d'un tour de jeu qui crée une ressource (création d'une carte en tant que druide, enregistrement d'une définition en tant qu'oracle) ;
- réalisation d'un tour de jeu qui utilise une ressource existante (arbitrage d'un enregistrement, succès ou échec d'un devin suite à l'écoute de l'enregistrement produit) ;
- mise à jour d'un élément de mon profil (nom, prénom, pseudonyme, mot de passe, photo, langues parlées, langue de jeu, langue d'interface).

Il est également possible de transmettre des messages généraux sur le jeu par ce mode, par exemple si une mise à jour est effectuée ou un défaut constaté.

5.1.3. Réalisation du système

Nous décrivons la réalisation du système de notifications dans cette section. La table dans la base de données et le modèle d'une notification ont déjà été présentées. Nous nous concentrons sur le contrôleur et l'affichage de la vue.

Le contrôleur de notifications – Il s'agit d'une classe définie en php qui s'appelle *Notification*. Le principal champ de cette classe est un tableau qui contient l'ensemble des notifications reçues par le joueur. Ce tableau est construit lorsque le joueur s'est connecté avec ses identifiants. La classe est alors instanciée et une fonction récupère depuis le serveur toutes les entrées de notifications associées à l'identifiant du joueur et les met à disposition dans le tableau.

La classe dispose également d'une méthode pour ajouter une entrée à la base de données. Cette méthode est déclenchée par un mode de jeu qui fournit les informations nécessaires au contrôleur qui peut ajouter la notification à la table. Une autre fonction de la classe permet de marquer les notifications lues par l'utilisateur.

L'affichage des notifications – L'affichage des différentes vues des notifications met en scène plusieurs objets dont le contrôleur. Le code qui déclenche l'affichage est situé dans le fichier de *header.html*, de sorte que l'affichage est mis à jour à chaque actualisation de page. Le code du *header* vérifie d'abord que le joueur est connecté à son profil. Si c'est le cas, le contrôleur peut être créé et le tableau de notifications récupéré. Celui-ci est transmis à une fonction appelée *notifBox.js* et écrite en Javascript. Elle se charge de mettre à jour le code *html* de la barre de navigation pour y inclure le tableau, qui reste invisible à l'utilisateur tant que celui-ci n'a pas appuyé sur le bouton associé. L'apparition d'une nouvelle fenêtre de notification est géré une autre fonction Javascript appelée *notifOn.js*. L'ensemble du fonctionnement est illustré par le schéma de la figure 29.

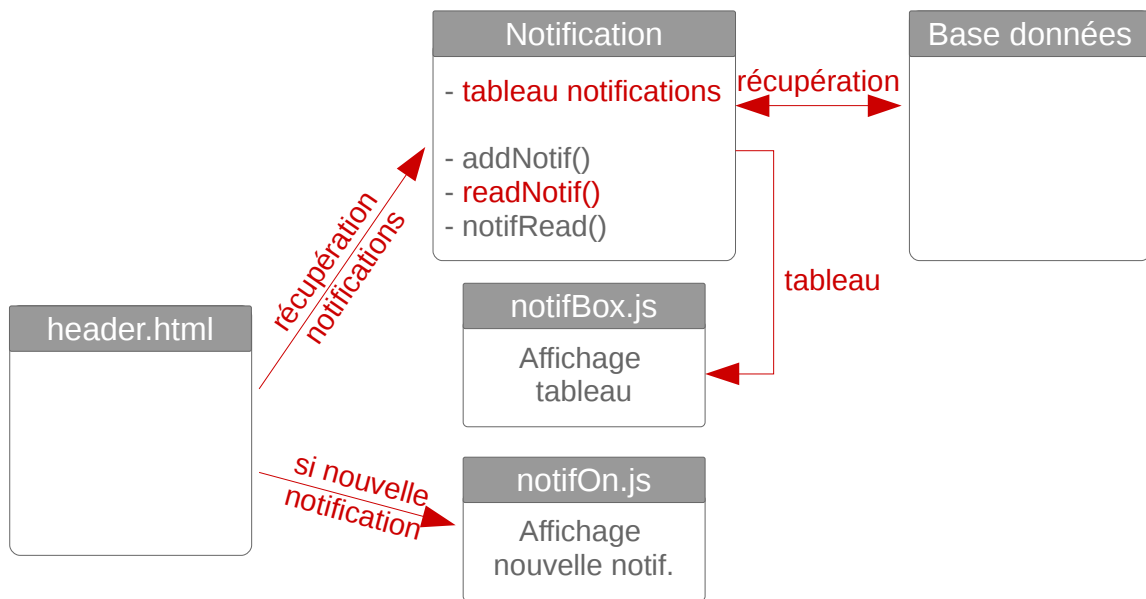


Figure 29: Schéma de fonctionnement de l'affichage des notifications

Nous illustrons le tableau de notification, le marqueur d'une notification non lue et l'affichage d'une fenêtre de notification dans les figures 30 et Error: Reference source not foundci-dessous.



Figure 30: Captures d'écran du menu déroulant des notifications et du marqueur de rappel



Figure 31: Illustration d'une notification d'événement

Limites du système actuel – Une grande partie des fonctionnalités prévues a été réalisée, mais certaines sont encore en cours de développement :

- les tours de jeu du joueur déclenchent l'affichage d'une notification immédiate mais n'apparaissent pas dans le tableau récapitulatif ;
- les mises à jour dans l'onglet du profil ne génèrent aucune notification ; à l'inverse, les changements de langues dans les modes de jeu sont à l'origine d'une mise à jour du tableau récapitulatif ;
- les événements d'autres joueurs sont directement rapportés dans le tableau récapitulatif, sans passer par l'affichage d'une notification immédiate.

5.2. L'amélioration des parties du devin, du druide et de l'oracle

Les tâches décrites dans cette section sont les autres modifications apportées en vue d'améliorer l'information donnée au joueur. Nous présentons d'abord le chronomètre du jeu dans les tours de l'oracle et du devin, qui a été repensé pour être mieux vu par les joueurs. Puis, nous présentons la nouvelle interface de l'oracle, censée être plus progressive.

5.2.1. Nouvelle visibilité du chronomètre

Suivant les principes de la POO, le chronomètre est représenté par une classe qui regroupe les principaux champs et méthodes nécessaires. Par ailleurs, suivant le modèle MVC, le chronomètre est contrôlé par d'autres fichiers pour les besoins de l'affichage. La séquence d'utilisation est la suivante :

- la page html de la vue qui gère le tour de jeu contient aussi du code php pour l'instanciation du chronomètre ;
- le chronomètre peut alors être démarré par sa fonction interne *start* ;
- la fonction *secondPassed* fait évoluer le décompte du chronomètre ;
- lorsque le temps est écoulé, la fonction *stop* est appelée pour terminer.

L'usage du chronomètre est complété par des fonctions utiles annexes, qui permettent la personnalisation de l'affichage.

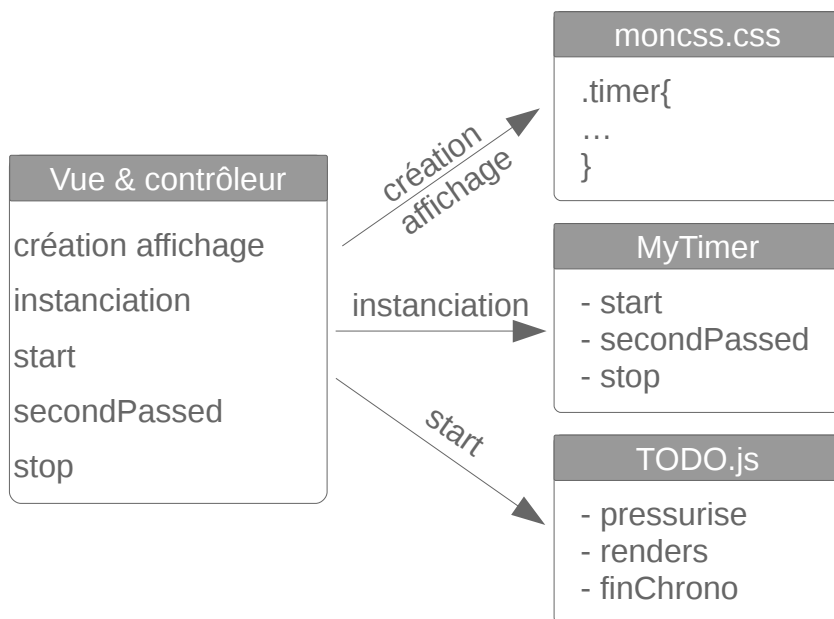


Figure 32: Schéma de fonctionnement du chronomètre

La particularité de ce chronomètre est la possibilité d'ajouter de la pression au joueur en attirant l'attention du joueur sur le temps restant. La réalisation rappelle le fonctionnement d'un voyant d'alerte : le temps restant clignote et change de couleur. Le temps d'activation de la pression est paramétrable ; nous avons choisi les 10 dernières secondes.

Le chronomètre est représenté par une classe en langage Javascript. Parmi les champs de cette classe, on retrouve la durée totale du chronomètre et la durée restante après le démarrage. Le pas de mise à jour du chronomètre est également défini dans les champs. Les 3 principales fonctions de cette classe s'appellent *start*, *secondPassed* et *stop*. La première démarre le chronomètre et appelle la fonction *secondPassed*. Cette seconde fonction calcule la durée restante et l'affiche. Elle est appelée régulièrement, suivant le pas de mise à jour, jusqu'à ce que le temps soit écoulé. Lorsque c'est le cas, la méthode *stop* est appelée.

Le modèle est complété par un ensemble de fonctions spécifiques. La fonction *renders* convertit le temps écoulé dans le bon format pour l'afficher ; en effet, le temps est mesuré en secondes qu'il faut transformer en minutes et secondes pour l'afficher. La fonction *pressurise* consiste simplement à changer la couleur de fond du temps affiché, régulièrement et aléatoirement, ce qui donne l'impression de clignotement dans plusieurs couleurs. Le pas de changement choisi par défaut est un quart de seconde. Ci-dessous, nous illustrons cette fonctionnalité par deux captures d'écran qui laissent imaginer l'effet.



Figure 33: Affichage du chronomètre et illustration de la fonction de “pression”

Enfin, la fonction *finChrono* est appelée lorsque le temps est écoulé et que la partie est finie. Elle modifie l'affichage du chronomètre et le remplace par un texte qui indique la fin du tour de jeu. La figure 34 montre ce qui est affiché à la fin d'une partie.

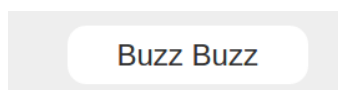


Figure 34: Message affiché à la place du chronomètre quand le temps est écoulé

5.2.2. L'interface de l'oracle plus progressive

Dans la version initiale du jeu, le tour de jeu de l'oracle était brusque et pouvait dérouter le joueur les premiers temps. En effet, le rôle de l'oracle est le moins intuitif des trois par son nom et, dans la version initiale, le joueur était directement amené sur la page de jeu, sans introduction à la tâche demandée. La figure 35 illustre la précédente interface. Les instructions étaient disséminées sur les différents éléments qui composent le jeu. Elles sont identifiées par le bloc rouge. La première se limite seulement à une phrase « Commence ta description ». D'autres sont résumées à un verbe, écrit sur les boutons pour indiquer les différentes possibilités (enregistrement, validation ou nouvel essai).

Par ailleurs, la représentation de la carte se limite à une liste de mots. Aucun des mots n'est mis en évidence, si bien que l'on peut les confondre.

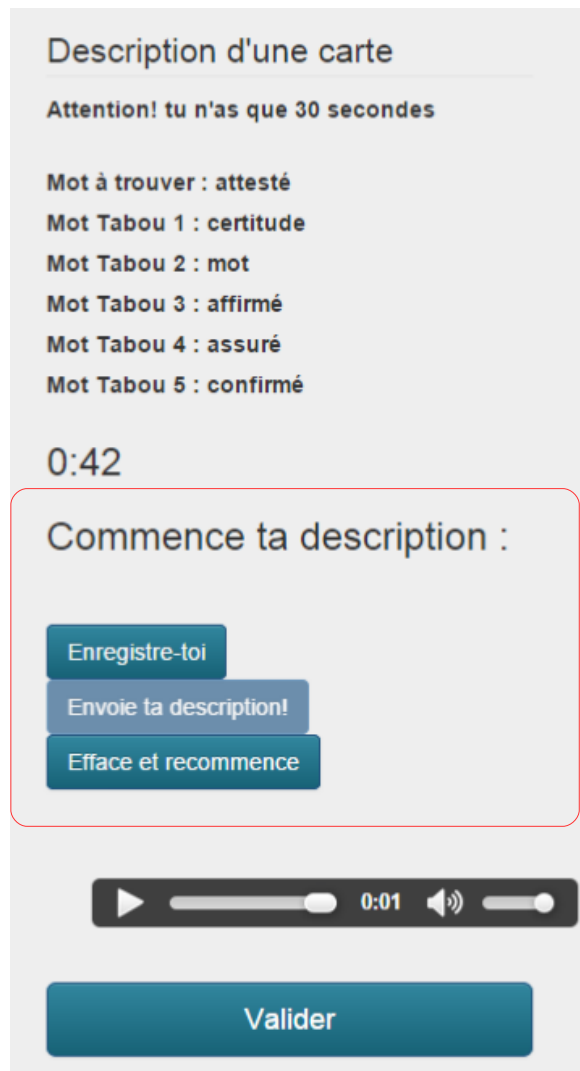


Figure 35: Capture de l'interface d'enregistrement dans la précédente version

Nous avons repensé l'interface afin de la rendre plus progressive. Ainsi, elle a été découpée en 3 parties, affichées les unes après les autres :

- l'explication des règles,
- l'affichage de la carte avec l'enregistrement
- l'envoi du fichier et la validation.

La première partie rappelle les instructions à suivre pour effectuer l'enregistrement. Comme on peut le voir sur la figure 36 qui représente une capture d'écran des instructions affichées, l'interface est épurée afin de concentrer l'attention du joueur sur les règles.

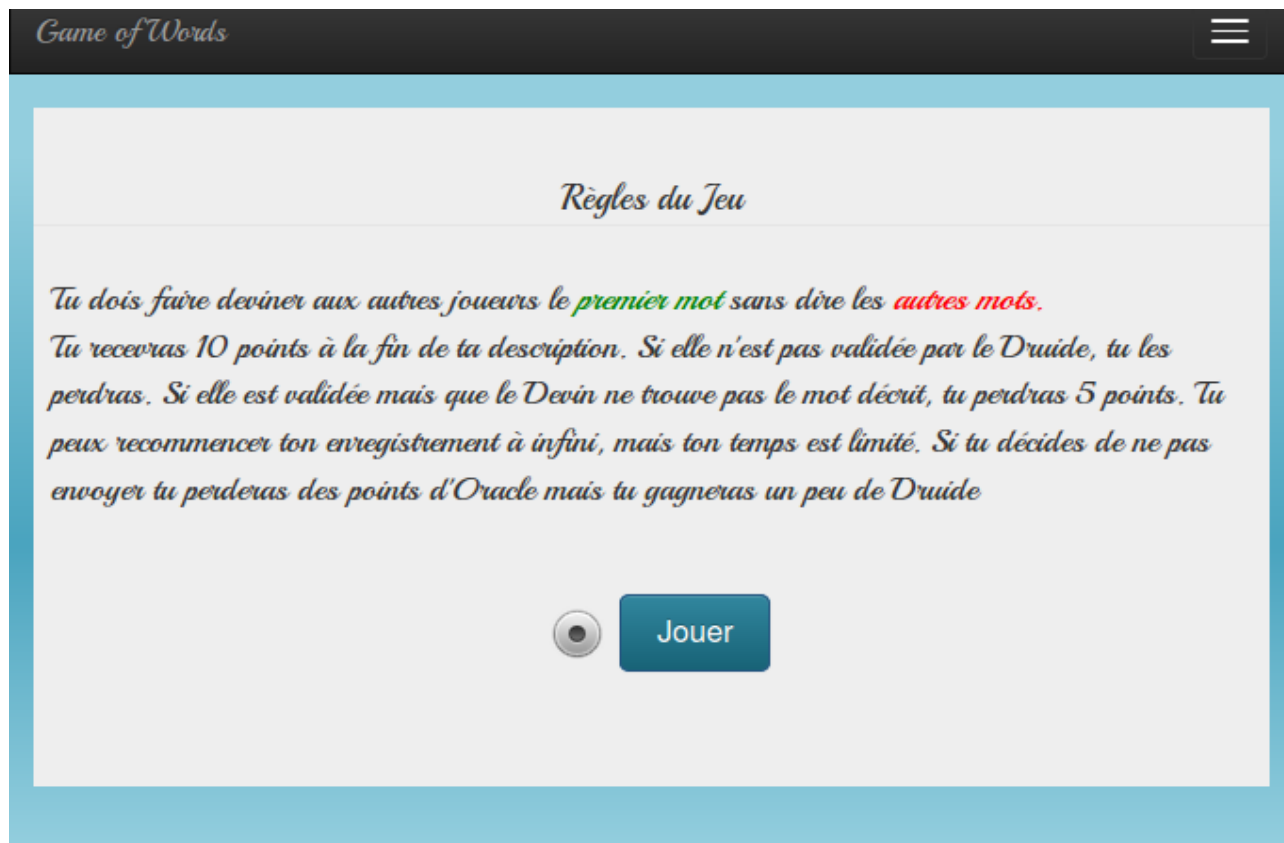


Figure 36: Capture d'écran de l'affichage des instructions dans la nouvelle version

La seconde partie consiste en l'enregistrement d'une définition. L'interface change pour afficher la carte à jouer (le mot secret et les mots tabous). Des indications et actions sur l'enregistrement sont également affichées, comme on peut le noter sur la figure 37. Ainsi, le chronomètre rappelle le temps restant ; le bouton rond et rouge clignote pour indiquer que l'enregistrement est en cours ; le bouton avec la flèche courbée désigne un recommencement de l'enregistrement ; le signe de validation vert permet d'arrêter l'enregistrement et de le valider. Le mot secret et les mots tabous sont mis en évidence afin que le joueur les identifie rapidement.



Figure 37: Capture d'écran du nouvel affichage des cartes

La troisième partie montre l'avancement de l'envoi du fichier (voir figure 38). Pendant ce temps, le joueur peut l'écouter et décider s'il souhaite le garder.



Figure 38: Avancement de l'envoi du fichier

5.3. Bilan

Pour conclure cette partie, nous avons présenté les travaux portant sur l'interface, axés autour de la transmission de l'information au joueur. À partir de la requête d'un système de notifications formulée dans le *backlog produit*, nous avons décrit les fonctionnalités intéressantes pour le jeu. Puis nous avons proposé une solution que nous avons réalisée. Les tours de jeu du devin, du druide et de l'oracle ont également été modifiés en insistant sur une interface plus claire. Le chronomètre est maintenant plus pertinent ; la carte des mots est plus expressive et les phases de jeu de l'oracle plus marquées.

Chapitre 6 – Perspectives de travail

De nombreuses perspectives de travail s'offrent aux prochains développeurs du jeu. Nous en proposons quelques unes dans cette partie.

Nous avons mis en place dans la version actuelle la possibilité d'ajouter le thème d'une carte. L'un des usages envisagés serait la mise en place de parties thématiques par les enseignants pour les étudiants. Cette fonctionnalité devrait permettre aux enseignants de se concentrer sur des champs lexicaux précis et de faire travailler les étudiants sur les différentes compétences d'expression et compréhension.

Les solutions pour la création automatique de cartes sont à explorer. La méthode consiste d'abord à déterminer le mot secret, puis à identifier des mots associés. La sélection du mot secret peut être effectuée naïvement à partir d'un mot tiré aléatoirement d'un corpus. La seconde étape d'identification des mots associés est plus complexe. La stratégie adoptée pour le français repose sur l'usage de l'outil Jeux de mots¹² qui classe les mots suivant différentes relations sémantiques au mot soumis. Par suite, il suffit de sélectionner un sous-ensemble parmi les mots les plus proches. Un tel outil n'existe pas pour d'autres langues, à notre connaissance. Cependant, il existe des ressources telles que Wordnet¹³ pour l'anglais, qui répertorie les mots dont le sens est proche et les associe à d'autres groupes de mots par des relations de sens. Il est possible d'explorer ces relations de manière automatique pour en extraire les mots les plus pertinents. C'est la solution envisagée pour la génération automatique de cartes en anglais.

L'estimation du niveau de carte est une tâche pour l'instant laissée aux druides qui créent les cartes. Cette estimation est subjective et peut donc varier suivant les joueurs en fonction de leur connaissance de la langue. Une évaluation automatique effectuée sur des critères définis permettrait d'homogénéiser les niveaux. De plus, elle pourrait être appliquée à la création automatique de cartes. La première stratégie pour l'estimation du niveau porte sur le nombre de mots tabous et la durée allouée pour deviner. Cette estimation peut être faite *a posteriori*. Cependant, le vrai problème concerne l'évaluation de la difficulté du mot et de la complexité ajoutée avec les mots interdits. Pour cela, il est possible de se référer à un niveau de vocabulaire des termes ; on peut également estimer la fréquence d'apparition du mot dans un corpus pour en juger la complexité en faisant l'hypothèse que plus un mot est fréquent, plus il est facile à deviner.

12 Pour plus d'informations consulter le site à l'adresse: <http://www.jeuxdemots.org/jdm-accueil.php>

13 Pour plus d'informations consulter le site à l'adresse: <https://wordnet.princeton.edu/>

Puisque le public visé est composé d'apprenants, il faut envisager que l'orthographe des mots ne soit pas maîtrisée. Or pour le moment, le traitement des réponses mal orthographiées est simplement rejeté. Nous envisageons un système qui comparerait le mot saisi par le devin à un dictionnaire pour vérifier l'orthographe. En cas de mot incorrect, le système devrait rechercher les mots les plus proches et les proposer au devin pour corriger sa réponse.

L'expérience du jeu pourrait être enrichie grâce à l'ajout de nombreuses fonctionnalités. La première consiste à introduire le concept d'expérience pour les joueurs. L'expérience doit être associée à des niveaux particuliers et des récompenses intermédiaires afin de stimuler le joueur. La gestion de l'expérience doit reposer sur un système de calcul de points et de règles d'attribution. Par exemple, si un joueur joue un nombre de parties suffisant dans les trois rôles, il pourrait être crédité d'un niveau. À l'inverse, si un joueur se limite à jouer seulement dans un ou deux rôles, il ne pourra passer au niveau supérieur. Les récompenses peuvent par exemple permettre l'accès à des fonctionnalités particulières comme des avatars pour le profil ou des bonus pour certains tours de jeu.

La combinaison des fonctionnalités précédentes d'estimation de la difficulté de la carte et d'estimation de l'expérience d'un joueur pourrait également permettre la suggestion de carte adaptée à un joueur.

Un autre moyen de stimuler les joueurs consisterait à permettre le défi entre joueurs. Par exemple, un joueur qui vient d'effectuer un tour du devin pourrait décider de défier un ami sur la même carte avec le même enregistrement. Le système des notifications pourrait être employé pour informer le joueur défié.

Le jeu peut être enrichi par la mise en place d'une communauté. Dans un premier temps, celle-ci pourrait permettre aux joueurs de trouver de l'aide sur le fonctionnement du jeu ou de formuler des commentaires. Une fonctionnalité plus avancée pourrait être de soumettre des réclamations sur le jeu. Par exemple, si un druide invalide un enregistrement, l'oracle devrait être en mesure de contester cette action. Dans ce cas, la communauté (ou au moins quelques uns de ses membres) pourraient être sollicitée pour statuer.

L'accès de la plateforme aux appareils mobiles tels que smartphones et tablettes pourrait inciter plus de joueurs à l'utiliser. Il est possible d'accéder à la plateforme depuis un mobile, mais l'enregistrement ne fonctionne pas correctement. L'une des perspectives consiste à adapter le code

pour corriger ce problème et ainsi assurer l'accès complet depuis des appareils mobiles.

Des imperfections de la nouvelle version sont également rapportées ci-dessous :

- le système de notifications est imparfait ; certains événements ne génèrent pas encore de notifications ; le nombre de points gagnés au cours d'un tour de jeu n'est pas toujours affiché ; le problème est dû à un changement de dernière minute dans le code qui ne met plus à jour la variable \$points initialement utilisée dans les modes de jeu (ex : *diviner.timeout.html*) ;
- le traitement des réponses du devin ne prend pas en compte les espaces ou les mots composés ; un traitement plus évolué assurerait une plus grande robustesse ;
- les langues de l'interface sont limitées au français et à l'anglais ; pourtant il est aisé d'ajouter d'autres langues grâce à la structure choisie : les différents messages de l'interface sont appelés par des variables regroupées dans un fichier par langue ;
- une autre erreur apparaît lors de l'affichage des scores :

Notice: Undefined index: Espagnol in /home/tab/public_html/GoW1.1/GOWTEST/Projet-GOW1.1-cleanUp4FF/test/GoWBETA/controllers/score.class.php on line 2

Conclusion

Au début du stage, Game of Words était un prototype. La conception avait été faite et l'architecture mise en place. Comme nous avons pu le voir, les fonctionnalités étaient bien définies, la base de données correctement élaborée, le code était bien organisé suivant des schémas et méthodes connues. L'interface proposait déjà un rendu de bonne qualité et les modes de jeu étaient accessibles. Le travail de conception et de réalisation de la structure de base était achevé.

C'est dans ce contexte que le stage s'inscrit. La version de prototype devait être testée pour évaluer la robustesse. De plus, l'ajout de fonctionnalités supplémentaires pouvait être abordé. En particulier, pour en faire un jeu d'apprentissage des langues, il était nécessaire d'améliorer la gestion de celles-ci. Suivant un cahier des charges précis, nous avons proposé un système qui permet aujourd'hui à un joueur de renseigner les langues qu'il parle et leur niveau de maîtrise, de choisir une langue de jeu parmi celles-ci et de déterminer une langue pour l'interface. L'interface a également été soignée pour lui permettre de visualiser et modifier ces informations en une seule fois, dans sa page de profil. En outre, le niveau de la langue permet non seulement l'évaluation par un enseignant mais il a également été transcrit en coefficient pour le calcul des points d'une partie.

À ce propos, la gestion du score et des points a représenté le deuxième axe de développement du jeu. Celle-ci se compose de règles d'attribution des points pour les différents cas d'utilisation ainsi que de formules qui calculent les points en tenant compte des concepts de difficulté du jeu, maîtrise de la langue et difficulté de la carte. La stratégie de calcul des points est une ébauche, mais elle modélise déjà partiellement l'influence d'une action d'un joueur sur un autre (par exemple, l'échec du devin face à un enregistrement cause la sanction de l'oracle qui a produit cet enregistrement). Un tableau des scores est également disponible pour les joueurs, afin de rendre le jeu plus attrayant par son aspect compétitif.

Enfin, comme on peut le deviner à travers les deux axes de développement précédents, l'amélioration de l'interface a constitué la troisième partie du travail sur le jeu. Bien que la base de l'interface initiale était bonne, elle présentait des limites par sa simplicité et son absence de fonctionnalités qui paraissent naturelles. L'une des tâches a consisté à mettre en place un système de notifications. Les bénéfices de ce système sont de divers ordres : distribution de l'information plus cadrée tout en étant élégante ; interface moderne. De plus, nous avons repensé certaines vues particulières, telle que l'ensemble du tour de jeu de l'oracle, qui est maintenant plus progressif et compréhensible. Nous avons également ajouté le concept de « pression » sur les tours de jeu, qui rend le jeu plus réaliste.

Au-delà des ajouts mentionnés, des vérifications et des tests ont été opérés sur différents navigateurs afin de rendre la plateforme plus robuste.

De nombreuses perspectives sont encore à envisager pour le jeu. Ainsi l'estimation de la difficulté d'une carte est laissé à son créateur, suivant une interprétation personnelle, mais devrait être automatisée, suivant des outils de traitement automatique de la langue. Dans le même sens, la création de cartes automatiques n'est possible qu'en français dans l'état actuel et devrait être élargie à d'autres langues. Cette fonctionnalité présente un fort intérêt pour les langues dont le groupe de joueurs est restreint. La présence d'une communauté de joueurs en interaction serait un atout pour ce jeu. Entraide, délibération sur des cas litigieux ou encore simple discussion sont des exemples d'application d'une telle communauté.

Sur le plan personnel, ce stage a été une très bonne expérience. Grâce au cadre de la méthode agile pour le développement et aux tâches de programmation, il m'a permis de développer mes compétences dans ce domaine et plus particulièrement, dans les langages du Web utilisés (Php, html et Javascript essentiellement). Également, la composition en binôme et l'affectation à deux jeux différents a nécessité une organisation certaine qui a été très enrichissante. Pour conclure, je retire une grande expérience de ce stage, sur le plan de la technique, de la gestion de projet et du travail en équipe.

Bibliographie

Brougère, G. (2005). *Jouer/Apprendre*. Paris : Economica/ Anthropos.

Caillois, R. (1958). *Les jeux et les hommes*. Paris : Gallimard (éd. revue et augmentée, 1967).

Cimpello, C., Reverdy, J. (2015). *Création d'un Serious Game de type Taboo| Time's Up destiné à l'apprentissage des langues*. Cahier des Charges : Traitement Automatique de la Langue Écrite et Parole. Grenoble : Université de Stendhal.

Conseil de l'Europe C.E. (2001). *Un cadre européen commun de référence pour les langues: apprendre, enseigner, évaluer*. Didier.

Doyé, P. (2005). *L'intercompréhension : Guide pour l'élaboration des politiques linguistiques éducatives en Europe- De la diversité linguistique à l'éducation plurilingue*. Division des Politiques linguistiques DGIV- Direction de l'éducation scolaire, extrascolaire et de l'enseignement supérieur Conseil de l'Europe. Strasbourg.

Hallal, R. (2014). *Game2Learn.Vers un système auteur pour la conception/réalisation de jeux tangibles en classe de langues*. Mémoire de master1 professionnel : Didactique des Langues et Ingénierie Pédagogique Multimédia. Grenoble : Université de Stendhal.

Maspero, M., Quintin, J.-J. (2014). L'innovation selon Innovalangues. *LEND lingua e nuova didactica*. N° 1/2014.

Sauvé, L., Renaud, L., Gauvin, M. (2007). Une analyse des écrits sur les impacts du jeu sur l'apprentissage. *Revue des sciences de l'éducation*, vol.33, n° 1, 89-107.

Rosen, E. (2009). La perspective actionnelle et l'approche par les tâches en classe de langue. *Le français dans le monde, Recherches et Applications*, n° 45.

Sitographie

Ludothèque ô fil du jeu. *Jeux et jouets - Ludothèque associative ofildujeu de Carbon-Blanc Gironde (33) près de Bordeaux* **[en ligne]**. [Consulté le 20.07.2015]. Disponible à l'adresse : <<http://www.ludothequeofildujeu.org/jeux-et-jouets/>>

Grenoble Inp-Phelma. *Phelma 2A SICOM et SEI: Programmation Orientée Objets* **[en ligne]**. [Consulté le 21.07.2015]. Disponible à l'adresse : <<http://tdinfo.phelma.grenoble-inp.fr/2Apoo/>>

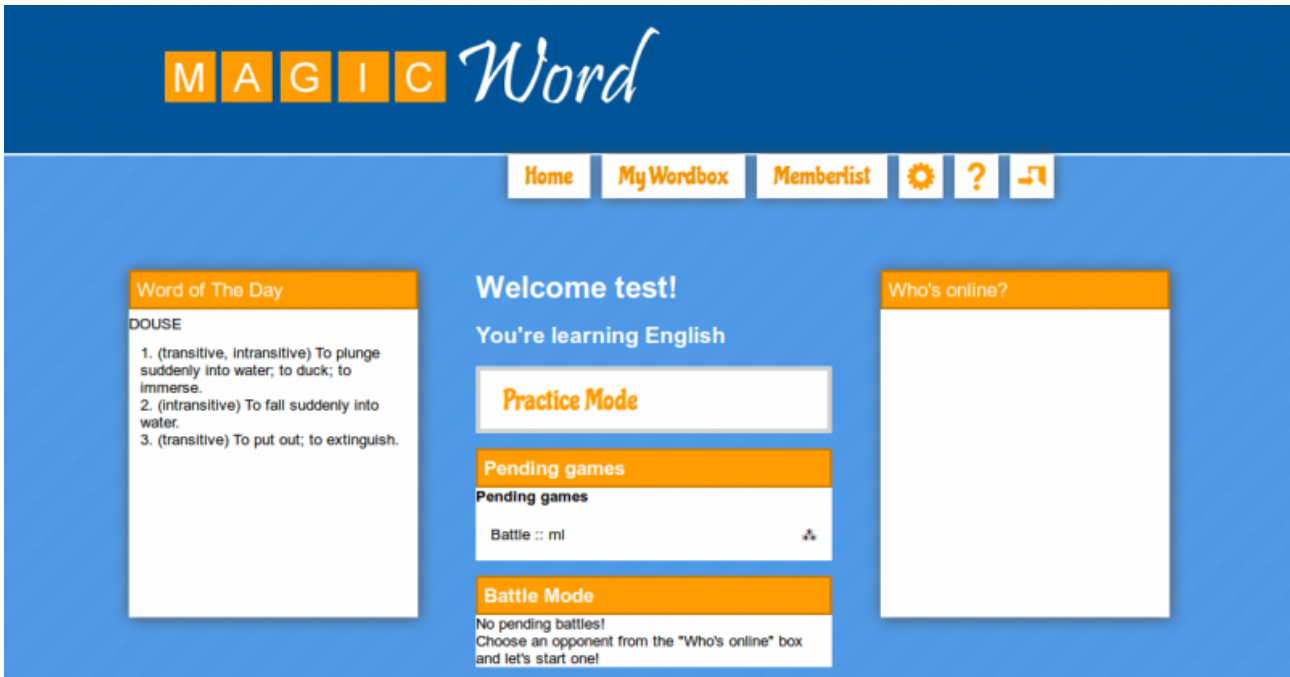
Innovalangues. *Wiki.innovalangues* **[en ligne]**. [Consulté le 05.06.2015]. Disponible à l'adresse : <<http://wiki.innovalangues.net/index.php/>>

Club Agile. *Manifeste pour le développement Agiles de logiciels* **[en ligne]**. [Consulté le 15.05.2015]. Disponible à l'adresse : <<http://www.agilemanifesto.org/iso/fr/>>

Annexes

Annexe 1

Ci-dessous deux captures d'écran du jeu Magic Word : la page d'accueil et l'illustration d'une partie.



MOTS-CLÉS : jeu du tabou, ingénierie pédagogique, développement web, apprentissage des langues, programmation orienté objet, MVC

RÉSUMÉ

Dans le cadre du projet Innovalangues (Université Stendhal-GrenobleIII), le lot GAMER a comme but à étudier et promouvoir la place du jeu dans l'enseignement des langues. Pour cela, il conçoit et développe un jeu du tabou qui s'appelle GAME OF WORDS. Ce jeu en ligne est le sujet de ce rapport du stage. La problématique se focalise sur l'amélioration du système du scoring, la gestion de langues et l'amélioration de l'interface de la première itération.

A partir d'un backlog du produit, qui résumait les fonctionnalités à développer nous avons défini les solutions à apporter et les avons réalisés. La gestion de langues porte sur la définition de plusieurs rôles pour les langues dans le jeu. La gestion du score porte sur la mise en place d'une stratégie d'attribution des points au différents rôles du joueur(Druide, Oracle, Devin). Enfin, nous avons amélioré l'interface en vue de la rendre plus ergonomique et plus claire pour l'utilisateur.

KEYWORDS : taboo game, educational engineering, web development, langages learning, object oriented programming, MVC

ABSTRACT

GAMER in Innovalangues Projet of Stendhal-GrenobleIII university has as a goal to study and promote the place of game in langages learning. For this reason, it designs and develops a taboo game called GAME OF WORDS . This online game is the subject of this report the internship. The problematic focuses on improving the system of scoring , managing languages and improving the interface of the first iteration.

From a product backlog, which summarized the features, we have defined to develop their solutions and have made them. The Languages management on the definition of roles for several languages in the game. The score management addresses the implementation of a strategy to different roles Player (Druid, Oracle, Devin). Finally, we have improved the interface to make it more ergonomic and clear for the user.