



# Hidden Words – Projet Final

Dossier de conception technique

Version 1.0

**Auteur**

Axel Mesnard  
Développeur Python



## TABLE DES MATIERES

<b>1 - Versions .....</b>	<b>3</b>
<b>2 - Introduction .....</b>	<b>4</b>
2.1 - Objet du document .....	4
2.2 - Références.....	4
<b>3 - Le domaine fonctionnel .....</b>	<b>5</b>
3.1 - Référentiel.....	5
<b>4 - Architecture Technique.....</b>	<b>6</b>
4.1 - Application Web.....	6
4.1.1 - AWS S3.....	6
4.1.2 - AWS DynamoDB.....	7
4.1.3 - AWS Lambda et API Getway .....	7
<b>5 - Glossaire.....</b>	<b>8</b>



# 1 - VERSIONS

Auteur	Date	Description	Version
Axel Mesnard	14/01/2022	Création et finalisation du document	1.0



## 2 - INTRODUCTION

### 2.1 - Objet du document

Le présent document constitue le dossier de conception technique de l'application Hidden Words.

#### Contexte :

Faisant partie d'un groupe qui est fan des mots fléchés, mots cachés/mêlés, etc, je me suis rendu compte qu'il pouvait être intéressant de créer un générateur de mots cachés/mêlés afin qu'ils n'aient plus à en acheter.

#### Objectif du document :

Définir le domaine fonctionnel et concevoir l'architecture technique de la solution.

### 2.2 - Références

Pour de plus amples informations, se référer également aux éléments suivants :

1. **DCF** : Dossier de conception fonctionnelle.
2. **DE** : Dossier d'exploitation.

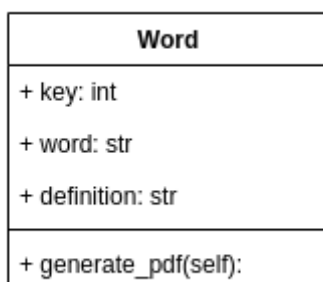


## 3 - LE DOMAINE FONCTIONNEL

### 3.1 - Référentiel

#### Diagramme UML de classes

*(Cliquez sur l'icône de pièce-jointe en haut à droite pour l'agrandir)*



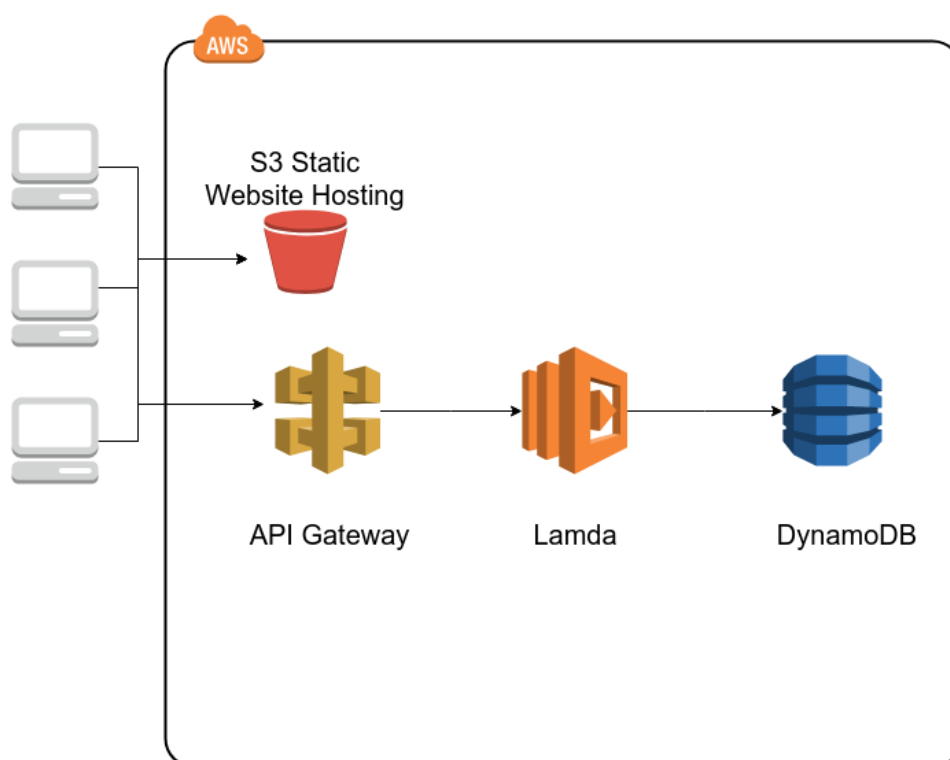
Structure du système Hidden Words modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets. Ici l'application est simple, seulement une classe est nécessaire au bon fonctionnement de l'application, à savoir les mots qui sont ajoutés à la grille. La key permet de faire une query des mots afin que la génération du PDF soit plus rapide qu'un simple scan. Les définitions étaient importantes pour la première version de l'application (mots croisés), ils ne sont plus utilisés pour la version finale mais j'ai préféré les garder en vue d'une future amélioration de l'application.



## 4 - ARCHITECTURE TECHNIQUE

### 4.1 - Application Web

Diagramme d'architecture AWS



#### 4.1.1 - AWS S3

Comme l'application consiste uniquement à une seule page web où l'utilisateur peut générer un fichier PDF, j'ai décidé d'héberger mon site web sur AWS S3 qui permet de stocker ses fichiers statiques et de mettre en ligne son site web facilement. A noter que j'ai deux "buckets" (espace de stockage) différents, un public pour les fichiers du site, et un autre privé qui héberge les PDF générés. Un lien sécurisé pré-signé est généré pour télécharger le PDF.



#### 4.1.2 - AWS DynamoDB

Pour stocker les informations des mots et de leurs définitions, nous utilisons DynamoDB qui est un système de base données non relationnel. Il est très utile dans un environnement dit de "serverless" car on peut très rapidement faire une query de la database. Le pricing est lié à l'écriture et la lecture effectuée. Nous ne payons pas lorsque l'application n'est pas utilisée. On peut évidemment améliorer sa rapidité en augmentant les taux d'écritures et de lectures alloués à une table.

#### 4.1.3 - AWS Lambda et API Gateway

Concernant les opérations sur le serveur, nous allons utiliser AWS Lambda. Lambda permet de ne pas mettre en place une instance EC2, il va directement exécuter le code qu'on lui demande et il se débrouille tout seul pour le scaling. Si mon application a des pics d'utilisation, Lambda va automatiquement allouer plus de ressources pour répondre à la demande. Afin que les fonctions Lambda s'exécutent, nous allons utiliser API Gateway afin de générer une URL de requête de notre fonction Lambda.



## 5 - GLOSSAIRE

<b>AWS Lambda</b>	AWS Lambda est un service de calcul d'événement sans serveur qui vous permet d'exécuter du code pour presque tout type d'application ou de service de backend, sans vous soucier de l'allocation ou de la gestion des serveurs.
<b>AWS S3</b>	Amazon Simple Storage Service est une solution de stockage sur Internet. Amazon S3 offre une interface simple de services web qui vous permet de stocker et de récupérer n'importe quelle quantité de données, à tout moment, de n'importe où sur Internet.