

# Projet S10

Application prononciation

**Contributeurs :**

Chabre Manon

Coscia Daniel

De Araujo Bastien

Denis Guillaume

Alacali Kadir Uraz

Dans le cadre de notre projet du dernier semestre de nos études nous avons eu l'opportunité de réaliser un projet de fin d'étude sur une application de prononciation dans le domaine médical.

Nous sommes donc un groupe de 5 où se trouve Manon, Daniel, Bastien, Guillaume et Kadir.

01 RAPPEL  
DU SUJET

02 ARCHITECTURE  
TECHNIQUE

03 RÉALISATION  
TECHNIQUE

04 GESTION DE  
PROJET

05 OUTILS  
UTILISÉS

06 MÉTRIQUE  
LOGICIEL

07 CONCLUSION



Dans un premier temps nous allons vous rappeler le contexte du sujet puis après parler de l'architecture technique, des réalisations que nous avons pu produire. Après nous allons vous expliquer notre gestion de projet, comment nous avons pu dans le temps imparti fournir une application correct et utilisable par les utilisateurs. Ensuite, nous parlerons des outils utilisés et nous finirons par les métriques logiciels.

---

## Rappel du sujet

- Application mobile
- Pouvoir permettre aux enfants atteints de trouble du langage de pouvoir apprendre à la maison
- Supervision par le spécialiste
- Travail avec une orthophoniste du CHU de Grenoble



Ce projet avait pour objectif de créer une application mobile pour une utilisation sur tablette dans le but d'aider les enfants en situation de trouble de langage fort. Pour ce faire, nous allons utiliser une approche via des mini-jeux pour que les enfants puissent travailler à la maison de manière autonome. L'application aura pour but de lui faire travailler la prononciation via de la gestuelle (en utilisant la méthode BOREL-MAISONNY).

Nous avons eu la chance de se faire guider par une spécialiste (orthophoniste) Estelle Gillet-Perret qui nous a permis de construire un cahier des charges précis. Nous la remercions aussi pour l'aide au niveau du design des images BOREL et des vidéos des sons.

## Architecture technique



### Architecture

- Utilisation sur tablette et/ou smartphone



## Technologies



- Moteur de jeu Godot Engine
- Utilisation d'API de google

Concernant l'architecture et les technologies utilisés c'est plutôt simple nous avons seulement une application sur mobile, elle n'utilise pour l'instant aucun serveur distant mais possède quand même une base de données local en ".json". Dans l'absolu il serait très intéressant de pouvoir utiliser un serveur (SQL ou autre) pour pouvoir stocker plus de contenu et permettre une collection des données qui pourront être ensuite utilisé par des chercheurs du métier.

Concernant les technologies nous utilisons un moteur de jeux open source brésilien qui ressemble à Unity, je me permet de vous envoyer le lien vers le site très bien documenté (<https://godotengine.org/>)

Godot Engine permet de faire des jeux 2D et 3D avec un langage propriétaire pour faire des scripts : GDScript. Ce langage ressemble beaucoup à du Python. Il nous a permis de réaliser des mini-jeux simples et ludiques pour que les enfants puissent apprendre facilement.

Nous utilisons aussi une API de google pour pouvoir écouter et enregistrer les mots prononcés par l'enfant. Celle-ci est sensiblement la même que celle de nos téléphones lors d'enregistrements vocaux. Dans un premier temps nous avons voulu prendre une autre API open-source mais aucune n'était aussi fiable que celle de google pour utiliser sur tablette. Nous avons même écrit un rapport de nos choix concernant l'enregistrement vocal (<https://github.com/WriteInGesturesProject/docs/blob/master/Analysis/STT%20%26%20TTS.md>)



---

## Réalisation Technique / DEMO



- Côté utilisateur "ENFANT" :

3 jeux à disposition : Jeux de l'oie, Écoute et choisis et Memory

Possibilité d'entraînement aux mots choisis par le spécialiste

Pouvoir s'entraîner sur la prononciation des sons avec une vidéo expliquant l'image Borel correspondante

Peut choisir son nom et un "avatar" qui lui correspondra

- Côté utilisateur "ORTHOPHONISTE" :

Possibilité de créer des exercices via la structure syllabique ou le nombre de syllabe.

Créer des exercices via un choix des mots choisis par l'orthophoniste

Possibilité de voir les statistiques de l'enfant pour voir les mots qu'il ne réussit pas

L'enfant pourra donc voir le mot et/ou la gestuelle et/ou l'image du mot et pourra le prononcer. L'enfant aura donc plusieurs niveaux de difficulté (seul le mot ou le mot plus la gestuelle, ou le mot, la gestuelle et l'image. Tout ceci en fonction d'un pourcentage de réussite) ce qui aura pour but de stimuler l'enfant et le faire progresser chaque semaine. De plus pour atteindre l'aspect jeu, l'enfant pourra créer son propre utilisateur (avatar) et pourra être récompensé à chaque succès pendant l'exercice (pièces, étoiles, ...).

Concernant les exercices, il y en a deux types. Le premier sera une suite de mots choisis par le spécialiste ou il pourra jouer à 3 types de sous-exercices. C'est à dire que l'orthophoniste pourra écrire des mots en langage phonétique et cela créera un exercice à partir de ces mots. Deuxièmement, les autres exercices sont déjà préchargés dans l'application et l'enfant peut s'exercer dessus. Nous avons : les jours de la semaine, les chiffres et les couleurs.

De plus, nous avons une partie de l'application dédiée au descriptif de chaque geste du dictionnaire BOREL-MAISONNY. Dans chaque geste il y a l'image du geste, la phonétique associée, un petit descriptif du son et une vidéo de la phonétique. Cette section sera dédiée aux enfants en apprentissage des gestes et aux parents pour leur faire comprendre le but de l'application.

---

## Gestion de projet

### Application de la méthode agile (SCRUM)

- Chef de projet : Manon
- Scrum master : Guillaume
- Développeur : Bastien, Daniel, Kadir
- Daily meeting
- Division en plusieurs sprint
  - Sprint 1 : 3 Février → 17 Février 2020
  - Sprint 2 : 17 Février → 9 Mars 2020
  - Sprint 3 : 9 Mars → 20 Mars 2020
- Utilisation des issues et pull request de Github avec revue de code



Concernant notre gestion de projet nous avons choisi la méthode SCRUM comme utiliser lors du projet ECOM nous avons donc un chef de projet, un scrum master et des développeurs. De plus, nous avons réalisé 2 entretiens avec Estelle Gillet-Perret pour avoir les besoins client. Nous faisons aussi des daily meeting pour faire en sorte que tout le monde ai une vision globale du projet.

Nous avons divisé le projet en 3 sprints d'environ 2 semaines chacun. Le Sprint 1 est le sprint d'approche de la technologie et de mise en place du cahier des charges, maquettes et autres documents essentiels au bon déroulement d'un projet. Le sprint 2 est le gros du projet, c'est la où nous avons réalisé toutes les fonctionnalités importantes et indispensables. Le sprint 3 quant à lui a été dédié aux finitions, gestion des bugs et aux nombreux rapports demandés.

Concernant la qualité du code nous avons fait des revues de code lors des pull request sur GitHub. Nous développons chaque feature (ou bug) sur une branche distincte et nous réalisons des merges quand nous finissons une feature.

---

## Outils utilisés



- Git
- Méthode SCRUM
- Revue de code
- Banque image libre de droit
- API de Google



Concernant les outils et techniques utilisés nous avons comme la plupart du temps utilisé Git, nous avons pris le site GitHub pour notre part. De plus nous utilisons des revues de code en utilisant des pull request.

Nous avons aussi utilisé pour notre design des banque d'image libre de droit tel que pixabay ou freepic.

De plus nous utilisons comme expliqué avant, l'API de google qui nous permet de faire du "TextToSpeech" et du "SpeechToText". Cela sert à écouter un mot et de pouvoir s'enregistrer pour savoir si le mot dit est exact.



## Métrique logiciel

Ligne de code, langages :

Temps ingénieur : 35 jours  
avec 7h par jour avec 5  
ingénieurs = 1225h

Language	files	blank	comment	code
JSON	219	1	0	14571
GDScript	32	594	269	2968
Java	19	125	811	1269
XML	65	305	142	932
HTML	1	3	13	185
Bourne Shell	1	21	22	129
Gradle	2	101	34	98
DOS Batch	1	23	2	59
SUM:	340	1173	1293	20211

Concernant les métriques logiciels nous avons pu grâce à l'outil cloc savoir le nombre de ligne de code de notre projet. Nous avons donc 20000 lignes environ. Les principaux langages sont le JSON et le GDScript. Nous utilisons JSON pour notre base de donnée locale et nous utilisons GDScript en langage principal de notre application.

De plus nous avons calculé 35 jours travaillé avec 5 ingénieurs soit un total de 1225h.

## Métrique logiciel

Répartition du travail :

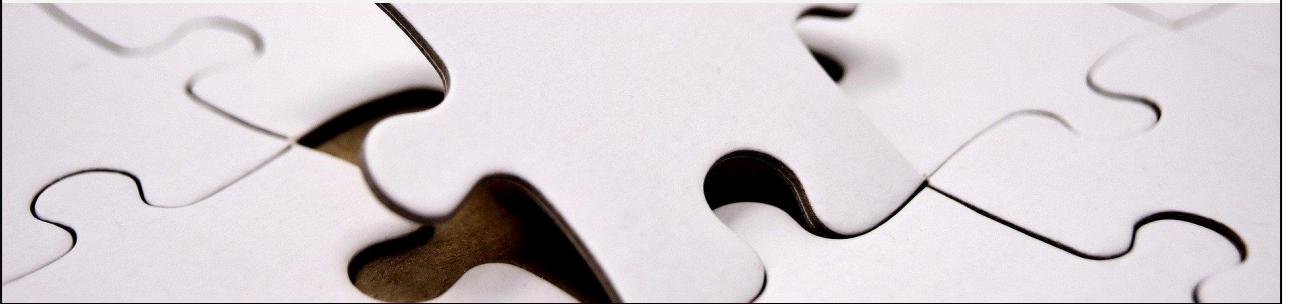
	Manon	Daniel	Bastien	Guillaume	Kadir Uraz	Total
Caractère ajoutés/ supprimés	24700/9600	3900/3100	5200/2100	12100/7400	5000/3600	50900/25800
Nombre de commit / pourcentage	94/32%	73/25%	48/16%	66/22%	9/3%	290

Concernant la métrique de travail nous vous avons affiché les caractères ajouté et supprimé ainsi que le nombre et pourcentage de commit en fonction des membres du groupe. Nous avons donc plus ou moins de commits et de caractères ajouté en fonction des fonctionnalités implémentés.

---

## Conclusion

- Travail qui pourra être amené à évoluer : projet Open-source
- Travail à but social qui pourra permettre à des enfants d'apprendre plus vite
- Nous avons découvert une technologie original : Godot Engine
- Nous avons fourni une application deliverable sur le PlayStore au minimum (Déploiement en cours, Acquisition des droit a l'image, licence....)



Pour conclure nous pouvons dire que :

- Nous avons réaliser un travail open-source qui pourra être continué par la suite, il est collaboratif. En effet les dessins des images borel ainsi que les vidéos ont été réalisés par l'orthophoniste.
- Ce travail a été motivant car il a un but social : aider les enfants dans leur trouble du langage.
- Nous avons abordé un technologie inhabituelle par rapport aux anciens langages que nous avons l'habitude de voir. Godot Engine était une autre approche de la programmation avec un peu de graphisme 2D.
- Nous avons fourni une application deliverable qui peut être utilisé aujourd'hui. Nous sommes également en train de voir avec l'orthophoniste pour déployer sur le PlayStore.