



## **Première Analyse/*First Analysis***

---

**Projet/*Project name* : DicMath**

**Equipe/*Team Number* : 101**

**Partenaire/*Partner* : Microsoft, Philippe Troitin**

---

## **1 Objectif du projet (attendu final) / Objectives and final delivery**

L'objectif de notre projet est de concevoir un logiciel permettant à des jeunes aveugles de dicter des équations mathématiques et de recevoir des retours vocaux pour pouvoir résoudre l'équation.

Ce logiciel fonctionnera sur ordinateur. Toutes les commandes peuvent être exécutées par saisie au clavier ou vocalement, ce qui évite l'utilisation de la souris.

Les principaux utilisateurs du logiciel seront des jeunes aveugles (niveau bac max). Le logiciel ne fournira donc que les opérations mathématiques de base, notamment l'addition, la soustraction, la multiplication et la division, la puissance et les fonctions.

Notre logiciel met en œuvre à la fois l'entrée et la sortie des équations : l'utilisateur dicte l'équation et le logiciel retourne vocalement l'équation comprise. Compte tenu de l'incertitude de l'équation d'entrée, le logiciel émet un signal sonore lorsque l'équation est terminée, répétant éventuellement l'équation et émettant respectivement un signal sonore différent lorsqu'une erreur est reportée. La fonction de navigation permet à l'utilisateur de localiser et de modifier l'équation existante.

Compte tenu des ressources disponibles en bibliothèque, nous mettons en œuvre la dictée dans un maximum de langues possibles, l'anglais et le français compris.

## 2 Décomposition en tâches et sous tâches / *task and subtask decomposition*

On peut séparer ce projet en deux parties importantes qui serviront dans la répartition des tâches :

### 2.1 UNE PARTIE FRONTEND, ELLE-MEME SEPARÉE EN PLUSIEURS SOUS-TÂCHE :

- La création d'une maquette visuelle qui vise à être la plus ergonomique possible pour des étudiants aveugles.
- La mise en place de l'interface via cette maquette qui puisse intégrer et utiliser le code du backend.

### 2.2 UNE PARTIE EN BACKEND, QUI REGROUPE ELLE-MEME PLUSIEURS SOUS-PARTIES :

- La mise en place de la structure de donnée pour le stockage des équations
- La création/amélioration d'un module d'Intelligence Artificielle de speech-to-text qui est suffisamment certain pour permettre à un malvoyant d'utiliser le logiciel en toute confiance
- La mise en place du mode de navigation à travers les équations avec diverses assistances (Raccourcis, retours vocaux...).
- Traduction multilingue de l'application

### 2.3 LE RASSEMBLEMENT DE CES DEUX PARTIES POUR AVOIR UNE APPLICATION FONCTIONNELLE ET COMPLÈTE

### 2.4 LA RÉDACTION D'UN MODE D'EMPLOI POUR LES POTENTIELS UTILISATEURS, OU PERSONNES QUI SOUHAITENT REPRENDRE LE SUJET.

### 3 Proposition de planning / *Planning proposal*

10–23 oct.

- Appropriation de la mission
- Schéma technique
- Contact avec l'ancienne équipe PI2 qui ont travaillé sur le même projet
- Chercher sur le choix des librairies, des logiciels, etc.

24–30 oct.

- Retour à P.Trotin qui est le responsable de Microsoft de nos idées
- Rencontre avec les jeunes aveugles
- Confrontation de nos idées avec leurs besoins
- Adaptation de nos solutions et décisions quant à la solution finale

1 nov. – 18 déc.

- Implémentation de la solution
- Codage, test, validation

18 déc. - 1 janv.

- Partie code finie, dernières retouches

2–23 janv.

- Dernières corrections
- Réalisation des vidéos et posters
- Complémentation le GitHub
- Ajout de la documentation/didacticiel

Nous respectons ce calendrier et nous pourrions travailler en avance si le projet se déroule bien. Bien que nous ne nous rencontrions pas pendant les vacances, nous confirmerons les résultats des travaux par télé-conférence pour nous garantir que rien ne bloque.

Des changements mineurs seront probablement inévitables, mais notre objectif sera toujours de respecter les délais importants.

## 4 Risk Management

Ce projet est un projet assez complet qui retrace de A à Z les étapes requises pour faire une application. Il est donc possible de retrouver des points de difficultés qui peuvent ralentir l'avancée du projet.

Afin de les détecter, prenons les différentes features dans l'ordre antéchronologiques de leur implémentation et voyons comment palier à leur manque en cas de problème rencontré.

### 4.1 La navigation

Si certaines bibliothèques permettent une meilleure compréhension de certains domaines, il n'y a aucune garantie de pouvoir détecter avec exactitude les différents mots requis pour une navigation confortable.

Il faudra alors revoir le fonctionnement de l'interface pour établir plus de raccourcis clavier, afin de garder l'application utilisable par la majorité des malvoyants.

### 4.2 La traduction et représentation fidèle en LaTeX

Nous utilisons aujourd'hui des bibliothèques qui utilisent du Speech-to-Text afin de nous faciliter la tâche lorsque nous voulons générer des Mathématiques en LaTeX.

Malheureusement, ces librairies sont presque exclusivement en Anglais, ce qui rajoute une difficulté quant à la traduction directe de la voix d'un étudiant en texte anglais pour l'utiliser comme entrée, tout en gardant la signification mathématique exacte de ce qu'il a voulu exprimer.

Dans le cas où nous ne trouvons pas de solution, il faudra revoir le reste de l'application pour que celle-ci fonctionne en Anglais, et rajouter les supports dans d'autres langues comme des features annexes lorsque le projet sera déjà dans sa phase finale.

### 4.3 Le stockage des données pour la navigation

S'il est possible d'enregistrer et d'afficher des équations assez fidèlement à l'entrée vocale, il est plus compliqué de savoir comment facilement y accéder une fois l'enregistrement fait.

L'idée est donc de simplifier le problème au maximum en utilisant une séparation « en bloc » de chaque équation, elle-même placée dans une structure de donnée qui est compatible avec les notions d'équations 'précédentes et suivantes'.

Si au final, il s'avère que ces solutions sont trop compliquées à implémenter, il faudra se reposer sur de la manipulation de liste et d'index.

### 4.4 Facilitation du vocabulaire à employer

Enfin, en tant qu'étudiants sans handicap, nous n'avons que l'expérience de logiciels comme Siri et Cortana pour donner des commandes vocales efficaces, mais nous n'avons en aucun cas une vision sur le fonctionnement de la résolution d'équations

mathématiques pour les étudiants malvoyants, il est alors compliqué de se mettre dans leur peau et de comprendre leur besoin sans référence.

Pour pallier ce problème, il est planifié de prendre rendez-vous avec l'INJA (institut national des jeunes aveugles) pour mieux comprendre leur besoin, et savoir précisément quelles commandes vocales figureront en priorité dans notre application.